

Министерство просвещения Республики Казахстан  
Национальная академия образования имени И. Алтынсарина



**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ  
ПО ИЗУЧЕНИЮ КУРСА «РОБОТОТЕХНИКА»  
В 10-11 КЛАССАХ**

Астана  
2022

## Содержание

### Содержание

<b>1. Особенности преподавания курса "Робототехника" в 10-11 классах на основе международного опыта</b>	<b>3</b>
<b>2. Актуальные вопросы преподавания курса "Робототехника"</b>	<b>47</b>
<b>3. Методическая система преподавания курса "Робототехника"</b>	<b>73</b>
<b>4. Методические рекомендации по возможностям использования ИКТ ресурсов при изучении курса "Робототехника"</b>	<b>234</b>

# 1. Особенности преподавания курса "Робототехника" в 10-11 классах на основе международного опыта

## Модуль 1. Преподавание робототехники и методические пособия с опытом других стран для 10-11 классов

В Германии проходят курсы для девочек Roberta Initiative <https://www.roberta-home.de/en/>. Данные курсы ориентированы на школьников средней школы и учителей. Проводятся, как правило, после школьных занятий. Roberta продвигает цифровое образование с 2002 года и сотрудничает с Fraunhofer IAIS. (Рисунок 1.1.)



Рисунок 1.1. Курсы для девочек Roberta Initiative

Fraunhofer IAIS (*Институт интеллектуального анализа и информационных систем Фраунгофера IAIS*) разработал технологию облачного программирования (CPT) и визуального языка программирования NEPO, на котором можно легко программировать даже сложные системы. В основе CPT лежит современная веб-серверная технология, с помощью которой можно создавать программы NEPO непосредственно в веб-браузере (Рисунок 1.2.). Визуальный язык программирования NEPO и интегрированная поддержка через среду программирования обеспечивают быстрый и легкий доступ к программированию даже для новичков. Платформа и язык программирования уже много лет успешно используются в образовании.

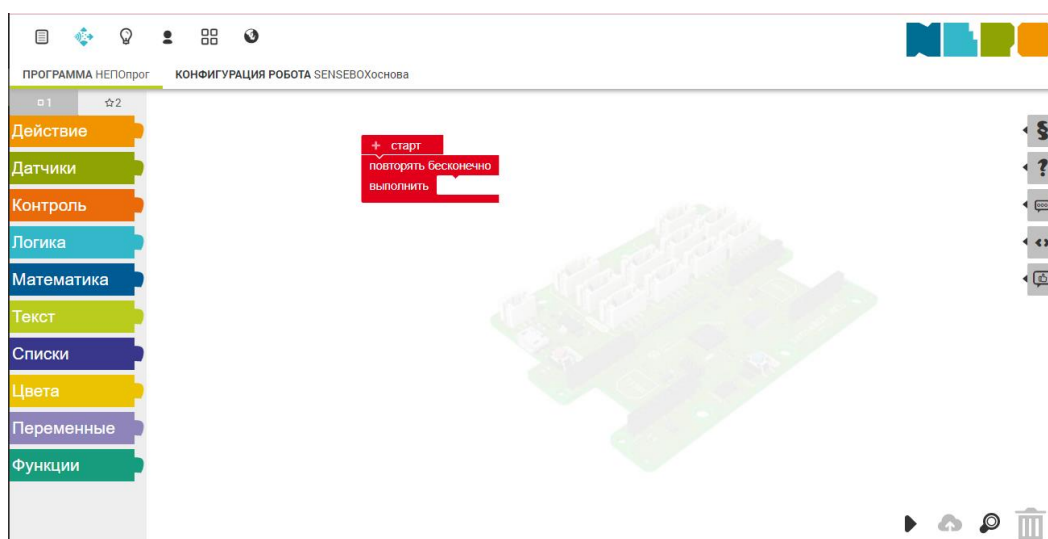


Рисунок 1.2. Создание программ с помощью перетаскивания блоков

Технология может использоваться независимо от операционной системы и может быть адаптирована для различных аппаратных систем. Несмотря на графическое представление, NEPO является всеобъемлющим языком программирования, который охватывает практически все аспекты традиционных языков программирования, таких как C, Java или Python.

В Open Roberta Lab можно запрограммировать множество ботов и досок (Рисунок 1.3.). Например, микроконтроллер для начинающих, управляющие роботы для продвинутых пользователей или робот-гуманоид для экспертов.

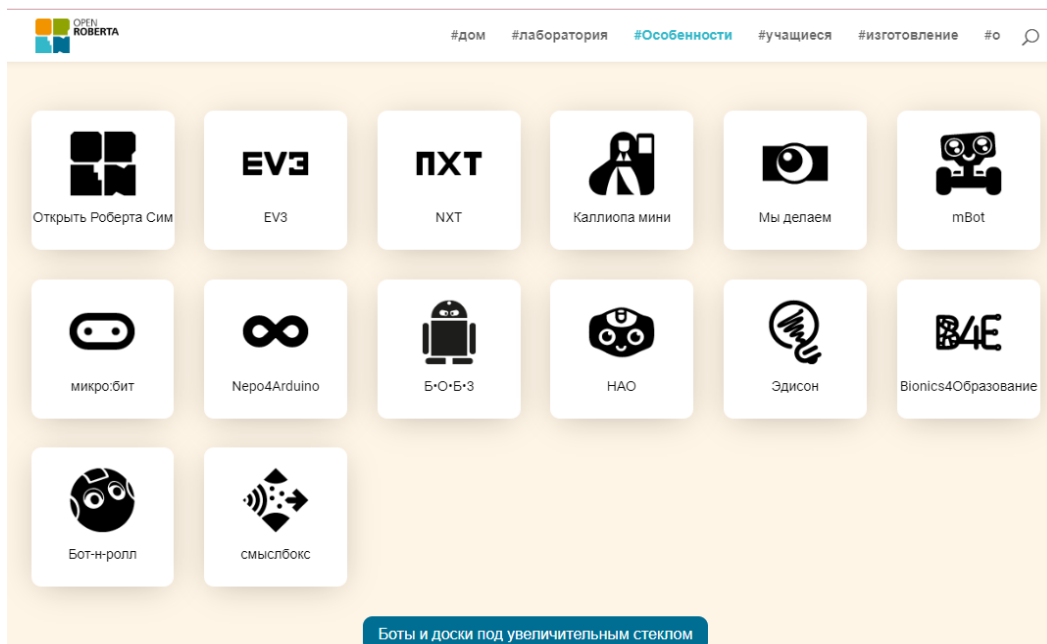


Рисунок 1.3. Боты и доски

Также интересен визуальный язык программирования Open Roberta NEPO, который доступен на бумаге (Рисунок 1.4.). Программирование без компьютера. Вырезаются блоки программы и на листе бумаги составляется код.



Рисунок 1.4. Программирование на бумаге



На YouTube канале «Программирование с Open Roberta feat. Даниэль Юнг и Софи Шарлотта» студентка факультета мехатроники Софи Шарлотта с преподавателем математики Даниэлем Юнгом демонстрируют программирование различных роботов и микроконтроллеров. (Рисунок 1.5.) [1]

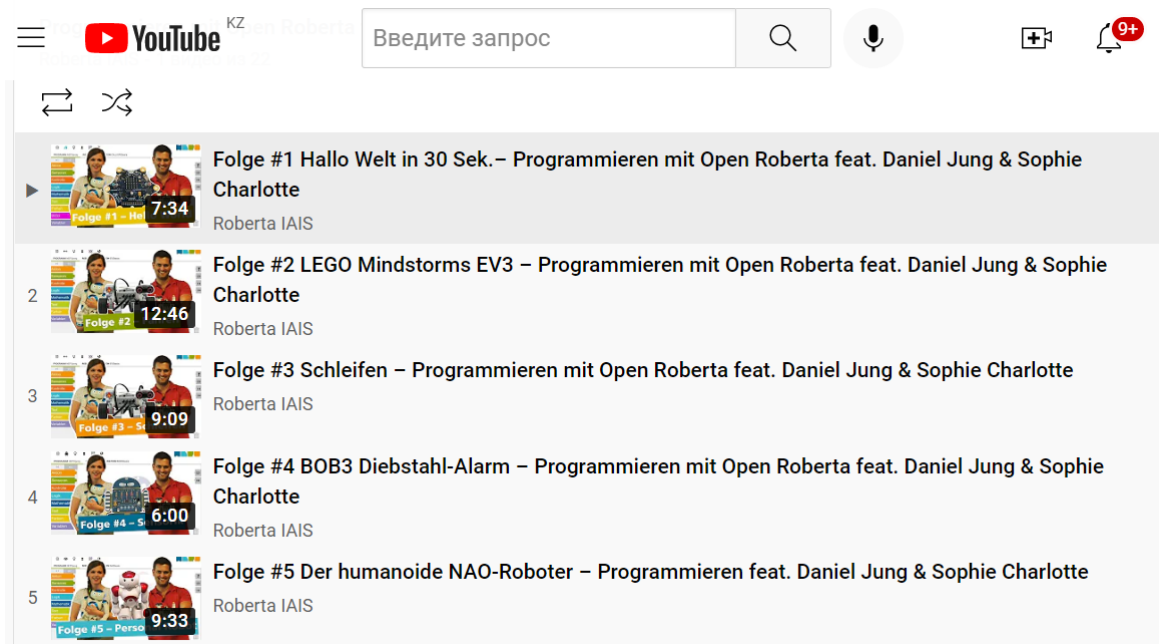


Рисунок 1.5. YouTube канал «Программирование с Open Roberta feat. Даниэль Юнг и Софи Шарлотта»

В Германии наблюдается повышенное внимание к курсам робототехники со стороны научных центров, таких как Heinz-Nixdorf-Museumsforum Paderborn, Deutsches Museum (Бонн), Odysseum (Кельн) или TUMLab (Мюнхен). Стоит отметить, что некоторые элементы курсов робототехники появляются в основных программах некоторых школ Германии. На данных сайтах можно ознакомиться с предложениями курсов на лето <https://mint-community.de/> <https://www.kommunale-koordinierung.com/adventureschool/>

С 2020 г. в Японии реализуется концепция STEM-образования (Science/Technology/Engineering/Mathematics). Она предполагает обучение точным наукам, технологии, инженерному делу. Программирование и робототехника стали обязательными предметами в школах Японии. Для школьников 8-15 лет составлена программа на 3 года (180 уроков) по следующим направлениям:

- Алгоритмика и креативное программирование в средах Scratch и AppInventor: программируем игру и мобильное приложение.
- Введение в умный дом и мобильная робототехника — создаем виртуальный умный дом и учим своего робота им управлять, готовимся с

роботами к соревнованиям по прохождению трасс и лабиринтов разной сложности.

- 3D прототипирование — моделирование и разработка трехмерных моделей разной сложности в программах Tinkercad и Freecad.
- Знакомство с технологиями дополненной реальности в Unity 3D и Unity AR - создание игр в AR, изучение основ дизайна.
- Продвинутое программирование в Python и Kotlin - базовые и продвинутые концепции программирования.

Содержание уроков построено на проектной работе, в процессе которой осуществляется обучение робототехнике и языкам программирования в контексте реальной социальной проблемы. [2]

В Корее в первом полугодии для детей 12-16 лет идет усвоение основных принципов робототехники и теоретических основ механики, физики, информатики, изучение принципов работы датчиков и исполнительных устройств, работы контроллера. Второе полугодие - занятия построены по принципу проектных технологий и разбиты по возрастам (12-14 лет, 14 - 16 лет) с изучением теории из смежных образовательных областей (химия, биология и т.д.). Структура УМК:

- краткая рабочая программа с таблицей моделей (Рисунок 1.6.);
- презентации для педагога (тренера) на каждое занятие в формате PPT - 70 шт. (34 презентации для 12-16 лет по темам занятий первого полугодия, 18 презентаций в рамках проектных технологий для 12 - 14 лет и 18 презентаций для 14-16 лет для второго полугодия);
- карты сборки моделей роботов для обучающихся для каждого занятия в PPT, примеры программ для программирования роботов;
- файлы "Дополнительные материалы к занятию";
- видеоматериалы с демонстрацией работы проектов;
- обучающее видео (ресурсы, находящиеся в открытом доступе в сети Интернет) по теоретическому материалу к занятиям;
- план-конспект к каждому занятию;
- пояснительная записка к каждому занятию. [3]












Занятие	Модель	Название	Что изучается	Занятие	Модель	Название	Что изучается
52		Робот-диагност (стоматолог)	Виды роботов, состоящих на службе у человека, принципы работы ИК датчика.	57		Проверка багажа	Основные принципы работы датчика магнитного поля, изучаются также понятия как «магнетизм», «магнитное поле», «металлоискатель».
53		Точка опоры	Понятия: "рычаг", "опора", принцип работы TFT - экрана.	58		Головоломка	Понятие "плоскость", изучение принципа работы датчика (гироскоп).
54		Сборщик космического мусора	Виды космических роботов, серводвигатель.	59		Рельсовый автобус	Изучение понятия "скорость".
55		Игрушка-геометрика	Основные задачи компьютерного зрения, датчик цвета.	60		Рельсовый автобус	Принципы работы двигателей постоянного тока.
56		Злая птичка	Изучение понятия "угол", его виды, варианты измерения.	61		Инкубатор	Изучение химического расстояния яйца, понятие "инкубатор", датчик огня.
				62		Игровая приставка	Понятие "электрический ток", датчик прикосновения.

Рисунок 1.6. Краткая Рабочая программа с таблицей моделей

Эстония в области робототехники использует комплекты роботов LEGO, комплекты Raspberry Pi для продвинутой робототехники, VEX-робототехнику, Arduino-платформу. В рамках проекта ERASMUS+ «Робототехника для школ» предусмотрены задачи и задания для школ, которые можно использовать без каких-либо технологических приспособлений, поэтому начать обучение робототехнике можно с низкоуровневого подхода. В Эстонии государственная школа Võru Kesklinna Kool интегрировала робототехнику в преподавание других предметов (информатика и естественные науки), используя Lego EV3. Роботы используются в практических работах по следующим темам:

- плоские геометрические фигуры: прямая, кривая, радиус, окружность, уравнения;
- скорость, длина и время движения, сила и взаимодействие, передача и крутящий момент, трение, гравитация, масса и инерция, потенциальная и кинетическая энергия, шаг и частота;
- температура, измерение, нагрев и охлаждение тел, теплоемкость и теплопередача, поток и представление результатов измерений на графике, и анализ результатов;
- основы программирования, языки программирования, переменная, константа, условное предложение, цикл, подпрограмма, интерфейс устройств, использующих Wi-Fi и Bluetooth, и т.д.

Первая Эстонская конференция по образовательной робототехнике состоялась 8 июня в Центре инноваций и бизнеса Mektory, Таллинн. В программу мероприятия вошли доклады о текущей ситуации в области

образовательной робототехники в Эстонии, Финляндии, Швеции и Великобритании, а также серия мастер-классов по актуальным темам и тенденциям в обучении робототехнике. [4]

К настоящему времени на русском языке выпущено большое количество авторской и переводной литературы по робототехнике. Опрос педагогов в России показывает, что наиболее часто ими используются книги и пособия следующих авторов: Д.В. Голиков, Д.Н. Овсяницкий, Д.Г. Копосов, А.В. Корягин, С.В. Косаченко, С.А. Филиппов, учебное пособие Киселева М.М. «Робототехника в примерах и задачах. Курс программирования механизмов и роботов», в котором собрано более 400 задач по образовательной робототехнике.

Кадровое и учебно-методическое обеспечение — острая проблема школьной образовательной робототехники. Число издаваемых пособий и учебников по образовательной робототехнике растет. Наибольшее количество методических материалов, курсов подготовки преподавателей создаются производителями образовательных конструкторов. Дефицит учебной и методической литературы сокращается, но сохраняет остроту. С одной стороны, многие преподаватели не имеют оперативной и полной информации о новинках и возможностях доступа к ним, с другой — объективно не хватает разработок, которые можно использовать в рамках основных общеобразовательных программ, особенно для старших классов и продвинутого уровня. Методические пособия, онлайн- и оффлайн-курсы производителей оборудования и аффилированных с ними структур в основном ориентированы на начальный ознакомительный уровень подготовки. Это приводит к тому, что преподаватели часто сами разрабатывают курсы на основе материалов, опубликованных в сети Интернет. [5]

Ниже представлена авторская разработка учебно-дидактического комплекса «Образовательная робототехника» для 5-9 классов, составитель Белов Роман Александрович, учитель информатики, г. Усть-Илимск, Россия. <https://infourok.ru/uchebno-didakticheskij-kompleks-obrazovatel'naya-robototehnika-dlya-zanyatij-s-uchashimisya-5-9-klassov-5177845.html>

Программа состоит из четырех направлений деятельности, которые тесно вплетаются в программу и взаимодействуют между собой. Это:

- **Лего-конструирование** – 1 год обучения (основы механики и конструирования). Цель – ознакомление с основами начального технического конструирования, моделирования и программирования;
- **Введение в робототехнику** – 2 год обучения (основы автоматического управления) предполагает использование компьютеров и специальных интерфейсных блоков совместно с конструкторами. Важно отметить, что компьютер используется как средство управления моделью; его использование направлено на составление управляющих алгоритмов для

собранных моделей. Воспитанники получают представление об особенностях составления программ управления, автоматизации механизмов, моделировании работы систем. Среда программирования EV3 позволяет визуальными средствами конструировать программы для роботов, т.е. позволяют ребенку буквально «потрогать руками» абстрактные понятия информатики, воплощенные в поведении материального объекта (команда, система команд исполнителя, алгоритм и виды алгоритмов, программа для исполнителя);

- **Соревновательная робототехника** – 3 год обучения, призвана, с одной стороны, стимулировать все учебные процессы на занятии, разнообразить методики обучения, развить навык работы индивидуально и в команде; с другой стороны, показать другие виды и возможности человеческой деятельности.
- **Спортивная робототехника** – 4 год обучения  
Программа составлена на 4 года обучения для ребят 11 - 16 лет (Таблица 1.1.).

Учебно-тематический план

Таблица 1.1.

№	Название раздела
<b>Кейс №1 «Технология и физика»</b>	
1	Введение в курс. Инструктаж по ТБ.
2	Обзор набора «Технология и физика». Способы и приемы соединения деталей. Комбинированные соединения.
3	Виды подвижных и неподвижных соединений в конструкторе. Особенности конструирования различных технических устройств.
4	Конструирование совместного объекта. «Роботизированная фабрика»
5	Конструирование совместного объекта. «Город будущего».
<b>Кейс №2 «Пневматика»</b>	
6	Знакомство с принципом работы пневмопривода. Измерения давления при помощи манометра. Конструирование простейших моделей с применением пневмопривода.
7	«Подъемный механизм»
8	«Пневмоподвеска транспортного средства».
<b>Кейс №3 «Альтернативные источники энергии»</b>	
9	Знакомство с принципом действия солнечной панели. Конструирование простейших моделей с применением солнечной панели
10	«Марсоход на солнечных панелях».

11	Знакомство с принципом действия суперконденсатора. Конструирование простейших моделей с применением суперконденсатора.
12	Знакомство с принципом действия элемента Пельтье.
13	Знакомство с принципом действия топливного элемента. Конструирование простейших моделей с применением топливного элемента.
<b>Кейс № 4 «Как научить робота думать, чувствовать, видеть»</b>	
14	Обзор набора LEGO® MINDSTORMS® Education EV3
15	Знакомство со средой программирования LEGO EV3
16	Конструирование робота-манипулятора
17	Использование датчика касания. «Управляемый робот».
18	Использование ультразвукового датчика. «Парковка робота»
19	Использования датчика освещенности (цвета)
<b>Кейс № 5 «Матрешка».</b>	
20	Обзор набора матрешка Амперка X.
21	Сборка простейших схем на базе Амперки X.
22	Сборка усложненных схем на базе Амперки X.
23	Проектный модуль «Создание автономного роботизированного устройства»
24	Проектирование автономного автоматизированного устройства.
25	Разработка основных узлов автоматизированного устройства. Эффективная работа над проектом.
26	Создание автономного роботизированного устройства.
27	Программирование роботизированного устройства.
28	Тестирование и отладка роботизированного устройства.
29	Защита проекта
30	Рефлексия.

С сентября 2022 года в силу вступят новые ФГОСы - федеральные образовательные стандарты для начальной и основной школы (с 1-го по 4-й и с 5-го по 9-й классы). Обязательным модулем дисциплины "Технология" станет робототехника: роботы-помощники, нейропротезы, "умный дом", 3D-дизайн и лазерная резка. [6]

## **Модуль 2. Тренировочный процесс по зарубежной методике для 5-9 классов**

Олимпиады, конференции и соревнования — это способы проверить свои способности. Как правило, задания на таких мероприятиях выше уровня школьной программы и требуют творческого подхода. С целью тестирования возможностей роботов разработчики придумывают множество разных

заданий. Нельзя рассматривать соревнования исключительно в развлекательном контексте. За зрелищностью всегда стоит длительная исследовательская и кропотливая практическая работа. На таких мероприятиях можно встретить достаточно много единомышленников и при необходимости найти наставника-практика. У каждого соревнования есть свой регламент, который размещается на сайте организатора или высылается на электронные почты участников. Исходя из требований школы участники начинают подготовку.

Успех команды на реальных соревнованиях показывает, насколько школьники стрессоустойчивы, «хладнокровны» и профессиональны. Применяя свои навыки, изучая регламент и критерии оценки работы, команда может добиться высоких результатов под руководством наставника, который на этапе тренировки акцентирует их внимание на важных аспектах работы.

Анализ работы с учениками в области робототехники выявил ряд проблем:

- обучающиеся испытывают трудности при чтении инструкции по выполнению задания, предпочитая объяснение преподавателем того, что требуется выполнить;
- не умеют разбивать задачи на более мелкие подзадачи,
- не умеют расставлять приоритеты при решении задач;
- не соблюдают тайминг;
- боятся принимать решения, предпочитают согласовывать каждый шаг с наставником

Участник соревнования должен уметь конструировать, работать в среде программирования, а также документировать процесс разработки робототехнической системы. В процессе подготовки к соревнованиям он также получит представление об общей организации и управлении ходом выполнения проектных, пуско-наладочных работ и сдачи в эксплуатацию робототехнической системы.

Подготовка к соревнованиям начинается с обсуждения регламента, чтобы не осталось «белых пятен» и были даны ответы на вопросы по заданию. Впоследствии участники предлагают и обсуждают варианты решения задачи, учитывая все факторы и условия поля и регламента. В процессе обсуждения школьники задаются вопросами:

- Какими функциональными возможностями должен обладать робот?
- На каком образовательном наборе мы можем реализовать свои идеи по конструкции (из тех, которые мы знаем/можем изучить за то время, что осталось до старта соревнований)?
- Какую движущую основу применить (колёсную, гусеничную, шагающую)?
- Какие сложности могут возникнуть?

На все эти вопросы зачастую ребята находят ответы, лишь только сконструировав и опробовав робота в реальных условиях, так как на этапе обсуждения многие ответы для них неочевидны.

Второй этап - создание программы.

Алгоритм решения большой сложной задачи надо уметь грамотно разбивать на простые подзадачи, которые ребята вполне научились решать в первые полгода-год обучения. Если всё же у ребят возникает страх перед большой комплексной задачей, то тренеру стоит задавать наводящие вопросы, чтобы они поняли, что вся задача состоит из подзадач. Чем больше вариантов по решению разработано командой, тем лучше она может быть подготовлена к быстрым переменам ситуации. В качестве тренировок наставнику следует задавать как можно больше вопросов типа «а что будет, если случится так...?», чтобы у учащихся в уме складывался ход решения обновленных условий.

Первым отличием инженерных соревнований по робототехнике от обычных робототехнических соревнований является именно инженерная составляющая. То есть необходимо не только создать робота и запрограммировать его для автономного решения им заданной задачи, но и представить инженерную книгу, в которой нужно подробно описать процесс создания робота и программы, материалов, из которых создан робот, этапы эволюции решения задачи, технические решения и находки и пр. Для успешного решения задачи важно правильное распределение ролей в команде. Довольно редко бывает, что роли в команде строго распределены, и каждый занимается сугубо своим делом, не переходя границы друг друга. Учащиеся могут исполнять такие роли как:

- *оператор-программист*, отвечающий за управление роботом, за правильно составленный алгоритм решения задачи, за программирование конструкции;
- *конструктор-механик*, отвечающий за создание, модернизацию, текущее обслуживание и усовершенствование робота, а также за текущее обслуживание всех запасных частей и аккумуляторных батарей;
- *технический писатель*, оформитель, отвечающий за создание, оформление инженерной книги, презентацию на этапе защиты проекта. Эту роль попеременно берут на себя оба участника проекта, так как инженерная книга – это, по сути, подробное описание этапов работы над проектом;
- *тренер/главный стратег*, отвечающий за принятие окончательного решения относительно стратегии, используемой в задаче. Эту роль обычно берёт на себя ученик–лидер (может быть как конструктором, так и программистом).

Опыт показывает, что все члены команды должны участвовать как в создании робота, так и в написании программы. Ведь если вдруг один из членов команды не сможет прийти на занятия, работа окажется в режиме паузы (будет сложно переделать конструкцию, так как «механика» нет, или перепрограммировать робота, так как аналогично нет «программиста»).

Согласно регламенту конкурсного задания, соревнования обычно проводятся 2 – 3 дня. В первый день участники должны представить свои



презентации и инженерные книги, а также продемонстрировать базовую функциональность своих роботов:

- движение по линии, в том числе через перекрестки (обычные и Т-образные)
- точные повороты, преодоление препятствия (горки)
- объезд препятствия (известный или неизвестный лабиринт)
- определение цвета
- захват и транспортировка

Второй соревновательный день предназначен для отладки робота и выполнения тестового задания.

Третий день посвящен выполнению оценочного задания.

Кажущаяся простота заданий первого дня может сыграть злую шутку, когда ученики не соблюдают тайминг, не скачивают программы на робота, а запускают все задачи непосредственно с компьютера через Bluetooth, не сдают вовремя робота на карантин и т.д. Некоторые из ребят относятся к простым задачам несерьёзно, предполагая, что если они демонстрируют финальное большое задание, то автоматически им засчитываются и все простые задачи как составные части целого задания. Финальная оценка за конкурсное задание соревнования складывается из всех оценок согласно критериям (кстати, не все критерии известны конкурсантам, например, такие как «порядок на рабочем месте», «взаимодействие с членами других команд» и др.). Невыполненное (не продемонстрированное) простое задание первого дня может лишить команду заветных баллов и не позволить занять достойное место победителя/призёра.

Следующей особенностью соревнований является подробное описание критериев оценивания конкурсного задания, в котором указаны не только навыки создания конструкции, сборки и отладки робототехнической системы, навыки программирования, отладки и настройки робототехнической системы, но и навыки документирования работ и подготовки сопроводительной документации, а также навыки взаимодействия, коммуникации и командной работы, навыки общей организации и управления ходом выполнения работ.

Довольно часто конкурсанты концентрируются на результате выполнения задания, упуская из внимания необходимость фиксации проб и ошибок, документирования работ, которое в дальнейшем могло бы позволить значительно сократить время и при подготовке сопроводительной документации, и при выборе наилучшего решения.

Пройдя все этапы подготовки и участия в соревновании, конкурсанты фактически проходят все этапы подготовки технического продукта от идеи до разработки, пуско-наладочных работ и сдачи в эксплуатацию робототехнической системы. Это позволяет ученику развить инженерные навыки в рамках практической предпрофильной подготовки.

Опыт работы по обучению школьников в робототехнической сфере и анализ их ошибок позволяет на основе сделанных выводов сформулировать рекомендации участникам соревнований:

1. Надежность конструкции и программа – самое главное, от чего зависит успех. Вы не сможете победить, если ваш робот не работает.

2. День, выделенный под тесты, имеет большее значение, чем день, потраченный на изменение «выпирающих деталей» и доведение робота до визуального «совершенства».

3. Перед каждой попыткой на соревновании нужно удостовериться, что батарея\датчики\моторы надежно закреплены и подключены, что батарея заряжена и на робота загружена последняя версия программы, которую вы донастраивали.

4. На соревновании нет места панике.

5. Будьте готовы к тому, что каждый элемент робота может выйти из строя, готовьтесь к худшему – надейтесь на лучшее! Будьте готовы произвести замену элементов.

6. Если что-либо пошло не так с роботом, не волнуйтесь. Попросите о помощи и попробуйте решить проблему.

7. Также стоит несколько раз уточнить у судей, верно ли вы поняли то, что от Вас требуется, не считайте ваш вопрос глупым.

8. Не теряйте надежды. Даже если команда переживает трудности, всегда есть шанс на победу. Продолжайте трудиться и постарайтесь максимально реализовать все возможности.

9. Будьте готовы к возникновению неконтролируемых ситуаций. Не паникуйте, сконцентрируйтесь на том, что вы можете контролировать.

10. Не бойтесь осматриваться и перенимать опыт других команд, также учите на чужих ошибках, да и на своих; плохой опыт – тоже опыт. Иногда обучение и вдохновение бывает спровоцировано достижениями других.

11. Максимально поддерживайте участников своей команды. Впоследствии может родиться хоть и одна, но гениальная идея.

12. Тяжелые задачи, ваша работа и энтузиазм в наибольшей степени способствуют скорейшему прогрессу в вашей деятельности.

13. Старайтесь мыслить позитивно и получать удовольствие от происходящего вокруг.

Именно на этапе подготовки к соревнованиям необходимо создавать (имитировать) условия участия в конкурсе, чтобы школьники фактически «вживую» проходили все этапы подготовки технического продукта от идеи до конечного результата. Именно тогда в рамках практической предпрофильной подготовки формируются элементы инженерного мышления, навыки инженерного способа работы. [7]

#### *Пример подготовительного этапа соревнований FLL 6 класс.*

Прежде чем определиться с темой проекта, обучающиеся провели исследование. С самого начала команда решила заняться проблемой браконьерства на территориях заповедников. Обучающиеся посещали городской зоопарк, общались с сотрудниками зоопарка, осуществляли звонки в

заповедник «Шушенский бор». Ребята изучали литературу и проанализировали горы информации в интернете. Шестиклассники сделали выводы о том, что в Саяно-Шушенском заповеднике очень остро стоит вопрос с браконьерством. «В средствах массовой информации мы очень часто слышим о животных, попавших в лапы браконьеров - утверждали обучающиеся, – и мы решили создать автоматизированную робототехническую систему из двух роботов, помогающих попавшим в ловушку животным».

На командном совете идею одобрили, и началась практическая часть. Перед тем как начать сборку робота, обучающиеся побывали на консультации у сотрудника зоопарка г. Зеленогорска. «Если рысь попала в капкан, и он не привязан, то она уходит в глубь леса. Если капкан на привязи, тогда она никого к себе не подпускает. Если перелом несерьёзный, то рану обрабатывают прямо на месте, а если есть перелом, то тогда подключается служба безопасности и увозит на базу. Нам нужна очень прочная сетка, чтобы обезвредить рысь и затащить ее в кузов робота-спасателя», – рассказали обучающиеся на презентации проекта.

Презентация проекта была яркой, выполнена в формате телевизионной передачи «Новости». Идеи, предложенные школьниками, были новаторскими, что и принесло им заслуженное призовое место.

Для диагностики подготовленности команды к соревнованиям используют технологические карты. [8]

Таблица 2.1. Карточка для оценивания работы №1:

№	Наименование критерия	Оценка (макс. 5 баллов)
1	Оригинальность и творческий подход	
2	Информативность. Грамотность речи	
3	Четкость. Доступность	
4	Артистичность	
5	Работа модели	

Таблица 2.2. Карточка для оценивания работы №2:

№	Наименование критерия	Оценка (макс. 5 баллов)
1	Правильность использования языка программирования	
2	Эффективность использования алгоритмических конструкций	
3	Управление моторами (направление, мощность)	
4	Оптимальное использование различных	

	типов датчиков (касания, освещенности, цвета, расстояния)	
5	Точность и полнота выполнения задачи	

Таблица 2.3. Карточка для оценивания работы №3:

№	Наименование критерия	Оценка (макс. 5 баллов)
1	Знание регламента (критерии, правила, подсчет очков)	
2	Эффективность выбора конструкции модели под поставленную задачу (жесткость, подвижность)	
3	Использование рычагов (1, 2, 3 рода)	
4	Использование передач (ременные, зубчатые, цепные, повышающие, понижающие)	
5	Максимальная грузоподъемность и количество степеней свободы	
6	Правильность соединения деталей	
7	Сложность конструкции	
8	Полнота выполнения задачи	

### Модуль 3. Методика оценивания робототехники, для 5-9 классов

Оценивание знаний в области робототехники проходит в двух направлениях:

- 1) оценивание за раздел «Робототехника» по материалам школьной программы;
- 2) оценивание спортивной робототехники на соревнованиях/конкурсах /фестивалях.

*Оценивание за раздел «Робототехника» по материалам школьной программы.*

По обновлённой программе в учебники информатики 5-9 классов внесён раздел «Робототехника», в котором учащиеся знакомятся с историей развития робототехники, с конструкторами, со средой программирования, запуском и отладкой робота. По завершению изучения данного раздела проводится СОР – суммативное оценивание за раздел. Школьники сдают СОРы в виде вопросов, тестов, контрольных. СОР основывается на критериальном оценивании знаний, умений и навыков, что исключает субъективный подход со стороны оценивающего.

Критерий – это составляющая стандарта. Если ученик знает критерий, он знает, как достичь стандарта темы, раздела. Критерии, в свою очередь, расшифровываются дескрипторами, прописанными в каждой суммативной

работе. При грамотно составленной СОР ребёнок может самостоятельно оценить качество своей работы, что является одной из целей современного подхода к обучению. Любой результат промежуточного оценивания – это рекомендация, какой критерий нужно подтянуть. Своего рода обратная связь. Чем конкретнее представлены критерии оценивания, тем лучше учащийся будет понимать, что ему нужно сделать для успешного выполнения задания. [9]

*Ниже представлен пример СОР в 5 классе:*

### ЗАДАНИЯ ПО СУММАТИВНОМУ ОЦЕНИВАНИЮ ЗА 3 ЧЕТВЕРТЬ

**Таблица 3.1 Суммативное оценивание за раздел «Робототехника»**

<b>Подраздел</b>	<b>Виды роботов и области их применения. Гироскопический датчик. Движение робота по линии. Робо – сумо</b>
<b>Цель обучения</b>	5.3.4.1 формулировать определение робота; 5.3.4.2 приводить примеры разновидностей роботов и области их применения. 5.3.4.4 объяснять принцип работы гироскопического датчика; 5.3.3.3 использовать датчик цвета для организации движения робота. 5.3.3.4 использовать датчик ультразвука для нахождения объекта.
<b>Критерий оценивания</b>	<i>Обучающийся:</i> ✓ формулирует определение робота; ✓ приводит соответствие разновидностей роботов и области их применения; ✓ различает функции датчиков цвета, ультразвука и гироскопического датчика; ✓ описывает простой алгоритм движения робота по линии; ✓ описывает на основе программы действия робота, содержащего датчики.
<b>Уровень мыслительных навыков</b>	Знание и понимание Применение Навыки высокого порядка
<b>Время выполнения</b>	20 минут

#### **Задания**

1. *Что такое робот?*

---



---



---



---

2. Роботы, в зависимости от их функций, бывают нескольких видов:

А) приведи соответствие:

<b>Промышленные</b>	
<b>Транспортные</b>	
<b>Бытовые</b>	
<b>Медицинские</b>	
<b>Образовательные</b>	

В) Какие виды роботов ты еще знаешь (назови два примера):

---

---

С) Для чего нужны роботы? (назови 2 причины)

---

---

3. *Рассмотри картинки датчиков. Напиши название каждого из них. Покажи стрелкой соответствие между устройством и его функцией*

---

---

---



Измеряет цвет и свет



Измеряет скорость вращения в градусах и секундах



Измеряет угол вращения

4. Опиши простой алгоритм движения робота по линии (вставь пропущенные слова)

Если датчик цвета видит \_\_\_\_\_ цвет, то робот будет двигаться в \_\_\_\_\_ направлении, а если увидит \_\_\_\_\_, то будет двигаться в \_\_\_\_\_ направлении.

5. Робо-сумо – одно из самых интересных соревнований роботов EV3. В этом соревновании робот должен вытолкнуть робота-соперника из круга, при этом сам он должен остаться в круге. Есть два типа площадки прохождения робо-сумо. Первый тип – это белый круг, окруженный черной линией, а второй – черный круг, окруженный белой линией. Какие два вида датчика нам понадобятся и для чего?

Таблица 3.2 Критерии и дескрипторы оценивания СОР

Критерии оценивания	№ задания	Дескриптор	Балл
		Обучающийся	
Формулирует определение робота	1	записывает определение слова «робот»	1
Приводит в соответствие разновидности роботов и области их применения	2 А	правильно определяет соответствие промышленного робота	1
		правильно определяет соответствие транспортного робота	1
		правильно определяет соответствие медицинского робота	1
		правильно определяет соответствие бытового робота	1
		правильно определяет соответствие образовательного робота	1
	2 В	называет один вид роботов	1

		называет второй вид роботов	1
Для чего нужны роботы?	2 С	называет одну причину	1
		называет вторую причину	1
Различает функции датчиков цвета, ультразвука и гироскопического датчика	3	пишет названия датчиков;	1
		верно определяет 1-2 соответствие между датчиком и его функцией	1
		верно определяет 3-4 соответствие между датчиком и его функцией	1
Описывает простой алгоритм движения робота по линии	4	верно определяет 5-6 соответствие между датчиком и его функцией	1
		правильно вписывает пропущенные слова	1
Описывает на основе программы действия робота, содержащего датчики	5	правильно указывает первый датчик, и для чего он нужен	1
		правильно указывает второй датчик, и для чего он нужен	1
<b>Итого</b>		<b>17 баллов</b>	

*Оценивание спортивной робототехники на соревнованиях/конкурсах /фестивалях.*

Оценивание соревнований по робототехнике также содержит в себе критерии и дескрипторы.

*Конкурс Best Robotics выдвигает следующие критерии оценивания команд-участниц (Рисунок 3.1.):*

- наличие заполненной инженерной тетради;
- презентация команды;
- выставка работ команды;
- дух и спортивное мастерство;
- производительность робота.

*Дополнительные баллы присуждаются за:*

- креативный дизайн;
- надежность конструкции.



Рисунок 3.1. Best Robotics



Также есть требования к роботу:

- Сборка робота производится только из комплектации набора.
- Размер робота не должен превышать 24\*24\*24 дюйма.
- Вес робота не более 24 фунтов.

Детали, размер и вес проверяются перед днём соревнований.

The BEST Award – это высшая награда в конкурсе. Ею награждаются три команды, которые лучше всего воплощают концепцию развития инженерии, науки и технологии посредством вышеуказанных критериев. [10]

**Всемирная Олимпиада роботов (WRO 2022)** в этом году разместила на своём сайте таблицу (Рисунок 3.2.), в которой прописаны все критерии и количество получаемых баллов за их выполнение.

<https://wro-association.org/wp-content/uploads/WRO-2022-RoboMission-Senior.pdf>



WRO 2022 – RoboMission - Senior

### Scoring Sheet

Team name: \_\_\_\_\_

Round: \_\_\_\_\_

Tasks	Each	Total	#	Total
<b>Doing the laundry</b>				
Laundry block completely in laundry area but not in a container or on top of a container	6	18		
Laundry block inside the laundry container of the <b>non-matching color</b> and the laundry container is still completely inside the laundry area	10	30		
Laundry block inside the laundry container of the <b>matching color</b> and the laundry container is still completely inside the laundry area	16	48		
<b>Bring water to the rooms (white block in front of the room, max. one per room counts)</b>				
Water is completely in the correct room but not on the table	6	12		
Water element partly or completely in a wrong room indicated by the marking blocks	-6	-12		
Water is on the table in the correct room, <b>not standing</b> , not damaged and the table is still inside the dark grey area	10	20		
Water is completely in the correct room, <b>standing</b> , not damaged and the table is still inside the dark grey area	14	28		
<b>Play a game (black block in front of the room)</b>				
Ball inside the game net (only one ball per net counts)	13	26		
<b>Park the robot</b>				
Robot stops in the start & finish area (only if other points, not bonus, are assigned)		13		
<b>Get bonus points</b>				
Per person that is not moved or damaged	4	24		
Per game net that is not moved or damaged	2	8		
Per table that is not moved or damaged	2	8		
<b>Maximum Score</b>		155		
				<b>Surprise Rule</b>
				<b>Total Score in this run</b>
				<b>Time in full seconds</b>

Signature Team

Signature Judge

Рисунок 3.2. Бланк подсчёта очков

Победители WRO награждаются дипломами, ценными подарками, поощрительными призами от организаторов. Награды Олимпиады содержат символику Олимпиады и название категории. [11]

### ***Российская Робототехническая Олимпиада 2022 (РРО).***

Организаторы РРО в заданиях уделяют особое внимание обучению с помощью роботов. В основной категории РРО ученики сосредотачиваются на развитии следующих областей:

- общие навыки программирования и базовые концепции робототехники (восприятие окружающей среды, управление, навигация);
- общие инженерные навыки (создание робота, который может толкать /поднимать объекты определенных размеров);
- разработка оптимальных стратегий для решения конкретных задач;
- вычислительное мышление (например, сборка, отладка, коллаборация);
- работа в команде, общение, решение проблем, творчество.

Для этого формата турнира рейтинг команд будет составляться на основе следующих критериев:

- сумма очков за лучшую попытку в 1-ый день и за лучшую попытку 2-ой день;
- сумма времени, потраченного на лучшую попытку в первый день и на лучшую попытку 2-ой день;
- очки лучшей попытки 1-ый день;
- очки лучшей попытки 2-ой день;
- время лучшей попытки 1-ый день;
- время лучшей попытки 2-ой день.

Если все предыдущие критерии у команд совпали, команды займут равные места. Предполагается, что каждый участник соревнований получит один из трех видов сертификатов/дипломов: сертификат участника, сертификат призера и диплом победителя с указанием места (1, 2, 3) на основании результатов работы робота (Таблица 3.3.). [12]

Таблица 3.3. Таблица присвоения сертификатов

<b>% от общего количества баллов (в возрастной группе) в лучшей попытке робота</b>	<b>Серти фикат</b>
<85%	Учasti е
> = 85%	Призёр
3 позиция в рейтинге	3 место
2 позиция в рейтинге	2 место
1 позиция в рейтинге	1 место

## Конкурс *RoboCupJunior 2022*

RoboCupJunior OnStage приглашает команды учащихся стать авторами сценической постановки, в которой примут участие автономные роботы, спроектированные, построенные и запрограммированные самой командой. Целью данных состязаний является создание живого или транслируемого роботизированного представления длительностью 1-2 минуты, в котором технологии станут предметом искусства. Формат выступления не регламентирован и может быть представлен любой зрелищной художественной формой. Это может быть танец, сказка, театральная постановка, художественная инсталляция и т.п. Выступление может сопровождаться музыкой, но это не является обязательным условием. Поощряется творческий и изобретательский подход команд к проекту – как к разработке самих роботов, так и к постановке выступления.

*Оценивание команд проходит в несколько этапов:*

- техническая документация;
- открытая техническая демонстрация - 30% от итогового результата (Таблица 3.4.);
- техническое интервью - 30% от итогового результата (Таблица 3.5.);
- выступление на сцене – 40% от итогового результата (Таблица 3.6.).

*Таблица 3.4. Критерии оценивания видеозаписи технической демонстрации*

<b>Категория</b>	<b>Критерий</b>	<b>Оценка</b>
Демонстрация работа(-ов)	Презентация полностью работающей роботизированной системы.	___/ 10
Процесс проектирования	Демонстрируются общие возможности работа(-ов), включая четыре выбранные ключевые особенности. Демонстрируются полностью работающие роботизированные системы без костюмов такими, какими они описаны в технической документации. Объяснение процесса проектирования при разработке роботизированных систем. Освещается то, как преодолевались трудности в процессе проектирования, особое внимание уделяется решению проблем командой. Рассказывается о ролях членов команды и их вкладе в работу различных систем (электромеханических, программных и т.д.).	___/ 6
Презентация	Четкость и качество презентации. Представлена хорошо отточенная демонстрация. Четко объяснены и	___/ 5

	представлены графики/чертежи, и сопроводительные материалы.	
Рассказ о технологиях	Передача информации. Эффективно, в сжатой и понятной форме до аудитории донесена информация о технических возможностях робота. Четко объясняются технически оригинальные, творческие или амбициозные концепции в роботизированном представлении команды.	___/ 5
Процесс выбора ключевых особенностей	Ключевые особенности. Командам присуждаются баллы за пояснение того, чем руководствовалась команда, выбирая четыре ключевые особенности, которые будут оцениваться во время их выступления.	___/ 4
	ИТОГО:	___/30

Таблица 3.5. Критерии оценивания технического интервью

Категория	Критерий	Оценка
Программное обеспечение	Способность объяснить, как работает программа, а также взаимодействие между программным и аппаратным обеспечением: - выбор языка программирования; - сложности в софте; - разработка соответствующих моделей, наборов данных и/или библиотек для решения программных задач; - инновационные программные решения; - эффективное и оптимизированное программирование с четкой документацией и комментированием.	___/6
Электромеханическое оснащение	Способность объяснить выбор электромеханической конструкции: - выбор материалов и приводов; - кинематическая система; - собственная разработка электроники (включая печатные платы); - управление питанием, регулирование, выбор батареи; - выбор микроконтроллеров; - конструктивные решения направлены на обеспечение надежности и долговечности систем. Объяснить, как системы соответствуют своему назначению, примеры включают в	___/9

	<p>себя: - комплексная мобильность; - робот, способный перемещаться в любом направлении/шагающий робот; - перемещение по различным поверхностям; - высокоточные системы, включая пневматику; - функциональные руки/ладони/лица; - роботизированные руки для манипулирования; - автоматическая система балансирования; - специальные компоненты</p>	
Системы датчиков и коммуникационные системы	<p>Способность объяснить роль датчиков и коммуникации в системах, и то, как роботы взаимодействуют со сценической средой: - роботизированные системы умеют динамически реагировать на незапланированные события; - роботы распознают свое окружение и, используя полученную информацию, динамически реагируют соответствующим действием; - интеграция многодатчиковых систем для разработки различных решений; - развитие коммуникации между датчиками; - разработка коммуникационных архитектур (асимметричная коммуникация) Объяснить, как системы соответствуют своему назначению, примеры включают в себя: - компьютерное зрение/голосовое распознавание; - разработаны системы ориентирования, навигации и управления; - взаимодействие типа робот-робот; - естественное взаимодействие робота и человека; - системы определения местоположения.</p>	___/9
Документация	<p>Продемонстрирована оригинальность проекта. Четко описаны четыре выбранные ключевые особенности. Четко описано выбранное аппаратное и программное обеспечение. Работа выполнена в правильном формате.</p>	___/6
Штрафные очки (по усмотрению судей до 15 баллов за каждый пункт)	<p>- судьи считают, что работа не была выполнена членами команды; - члены команды не могут объяснить своё техническое участие в проекте.</p>	
	ИТОГО:	___/30

Таблица 3.6. Критерии оценивания выступления

Категория	Критерий	Оценка
<p>Визуальное воздействие и качество всего выступления</p>	<p>Выступление роботов увлекает зрителей и делаются попытки коммуникации с ними. Например, на протяжении всего выступления прослеживается четкая линия/тема/идея/сообщение. Тема представления ясна и хорошо понятна. Выступление увлекательное и направлено на то, чтобы развлечь зрителей. Костюмы роботов дополняют представление, добавляют ценность и обеспечивают зрелищность. Эффективное использование сценического пространства в соответствии с темой или общей идеей. Взаимодействие с оригинальным и инновационным реквизитом или декорациями влияет на выступление таким образом, что оно захватывает зрителя и повышает ценность. Выполняются рискованные/сложные движения, которые дополняют тему. Эффектное и интересное взаимодействие между роботами и/или людьми.</p>	<p>___/16</p>
<p>Эффективная реализация особенностей, представляемых командой</p>	<p>Реализация ключевых особенностей/взаимодействие/интеграция систем: -0 баллов - не реализовано; -1 балл - низкий уровень реализации - работает не так, как ожидалось, и не добавляет ценности выступлению; - 2 балла - средний уровень реализации - работает так, как ожидалось, но не добавляет ценности выступлению; 3 балла - высокий уровень реализации и воздействия - работает так, как ожидалось, и добавляет ценность выступлению; 4 балла - очень высокий уровень реализации и воздействия - работает так, как ожидалось, и добавляет значительную ценность выступлению                      Особенность 1: ___ /4                      Особенность 2: ___ /4                      Особенность 3: ___ /4                      Особенность 4: ___ /4                      Взаимодействие: ___ /4                      Интеграция систем: ___/4</p>	<p>___/24</p>
<p>Штрафные очки (по 3 балла за</p>	<p>- каждое незапланированное вмешательство человека (включая дистанционные или</p>	

каждый пункт по усмотрению судей)	управляемые человеком действия); - один или более перезапусков; - каждые 10 секунд превышения временного лимита.	
		ИТОГО: ___/40

Итоговый балл каждой команды вычисляется как сумма баллов, набранных командой по результатам технического интервью, технической демонстрации, а также выступления на сцене. [13]

#### Модуль 4. Международные соревнования по робототехнике

Соревнования по робототехнике можно классифицировать по различным основаниям. Выделяют три основных подхода:

- *Первый подход* — условно «олимпиадный», когда ответ или решения конкурсных заданий состязаний заранее известны (WorldSkills, JuniorSkills, ряд направлений WRO и «РобоФеста»).
- *Второй подход* — условно «конкурсный», когда эталонного ответа на поставленную соревнованием задачу не существует (EUROBOT, часть направлений RoboCup, ABU Robocon, хакатоны). Этот подход в большей степени способствует развитию креативности участников, стимулирует научные исследования и технологические разработки. Этому также способствует снятие ограничений на платформы и некоторые другие искусственные ограничения.
- *Третий подход* — стимулирующий (мотивирующий), когда соревнования являются в большей степени праздником, игрой, общением (FLL, Junior FLL, «РобоФинист», часть региональных, городских, районных соревнований). Задачей таких соревнований могут быть выявление талантливых учеников, обмен опытом, мотивация участников на дальнейшее обучение, навыки управления и работы в команде.

Сравнение робототехнических соревнований со спортом породило термины *соревновательная робототехника* и *спортивная робототехника*. При этом в сообществе участников соревновательных движений нет единого мнения, одно и то же это или нет. [14]

#### США

Одно из первых крупных робототехнических соревнований, известное как *FIRST Robotics Challenge (For Inspiration and Recognition of Science and Technology)* и берущее свое начало в США еще в 1992 г., так и переводится: «первый конкурс робототехники» (Рисунок 4.1.). Идея конкурса принадлежит создателю сегвея Дину Кеймену. Это международное состязание для школьников было задумано как альтернативная форма передачи знаний и навыков молодому поколению при непосредственном общении учителя и

ученика. Задача конкурса — за шесть недель построить робота, который способен выполнить ежегодно устанавливаемую организаторами задачу, работая автономно и/или под управлением беспроводного устройства. Для построения робота командам предоставляется стандартный набор запчастей и деталей, который можно пополнять. Каждая команда состоит из школьников и их наставников — профессоров и опытных инженеров. За всю историю в конкурсе приняли участие более 29 000 команд общей численностью более 300 000 участников и 50 000 наставников, построивших более 25 000 роботов.



Рисунок 4.1. FIRST Robotics Challenge

Надо отметить, что этому конкурсу оказывают серьезную поддержку крупные промышленные компании, такие как General Motors, Daimler, Motorola, National Instruments и др. В рамках FIRST выделяются несколько программ: FIRST Robotics Competition (ученики старше 9 класса), FIRST Tech Challenge (ученики с 7 класса), FIRST LEGO League (с 4 по 8 классы) и младшая лига Junior FIRST LEGO League. В рамках FIRST реализуется специальная стипендиальная программа.

Следом за FIRST в США в 1993 г. под патронатом Texas Instruments прошел аналогичный по формату конкурс **BEST** (Boosting Engineering, Science and Technology). Его отличие состоит в том, что командам требуется не просто создать робота, но и представить его как полноценный продукт, готовый к выходу на рынок. Судейство в конкурсе осуществляется по четырем категориям: оформление технической документации, маркетинговая проработка и презентация проекта, представление команды, спортивный и командный дух, и уже с меньшим количеством баллов — качественные характеристики самого робота.

## Европа

Европейским ответом североамериканским инициативам стал *молодежный конкурс EUROBOT* (Рисунок 4.2.), впервые прошедший в 1998 г. Регламент соревнований меняется каждый год в зависимости от уровня развития и тенденций в робототехнике и оглашается в октябре. Постоянным остается лишь то, что матчи проходят на поле с произвольным ландшафтом размером 2\*3 м команда на команду и длятся полторы минуты, а роботы по габаритам должны вмещаться в куб 30\*40 см.





Рисунок 4.2. молодежный конкурс EUROBOT

Команды могут быть организованы студентами в рамках образовательных проектов или независимыми клубами и некоммерческими организациями. В настоящее время движение включает более 450 команд из 30 стран. Спонсорскую поддержку конкурсу оказывают концерны Renault и Siemens. Соревнования проводятся и в России.

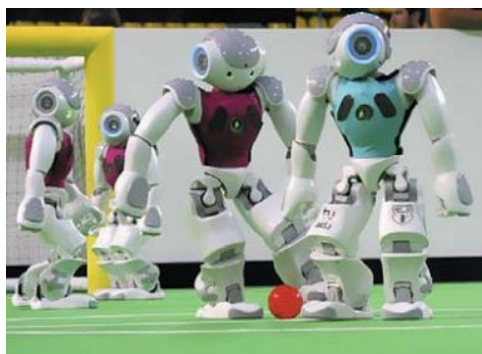
*Международный конкурс RoboCup* проводится с 1997 г. и изначально задумывался только как футбольное соревнование (Рисунок 4.3.).

К настоящему времени в основной футбольной категории выделяют пять лиг:

1. Соревнования в моделировании. Две подлиги: двумерное и трехмерное моделирование. Компьютерная программа управляет двумя командами по 11 игроков на виртуальном поле, игра идет по всем правилам. Процесс игры разделен на 6000 циклов, при этом алгоритмы участников должны выдавать готовое решение на каждом ходу не более чем за 100 мс. Эта лига предназначена, главным образом, для развития технологий искусственного интеллекта.
2. Соревнования футболистов малого размера. Матч проходит между двумя командами по пять мобильных, как правило, колесных роботов до 18 см в диаметре и высотой до 15 см. Позиционирование агентов может осуществляться с помощью бортовых камер или внешней глобальной системой технического зрения.
3. Соревнования футболистов среднего размера. Матч проходит между двумя командами по шесть мобильных роботов размером до 50 см на крытом поле размером 18\*12 метров. Каждый робот имеет бортовую вычислительную систему. Вмешательство человека в игру допускается только для замен, в остальном игроки полностью автономны и поддерживают связь между собой по беспроводному каналу.
4. Соревнования стандартных платформ. В игре участвуют одинаковые коммерчески доступные аппаратные платформы, так что участники концентрируются только на программном обеспечении. Изначально в лиге участвовали собакоподобные роботы SONY AIBO, поэтому она называлась «Лигой четырехногих роботов». Впоследствии стали использоваться антропоморфные роботы Aldebaran NAO, и лига была переименована. По правилам роботы должны быть полностью

автономными — ни удаленного управления, ни использования дополнительного внешнего оборудования не допускается.

5. Состязания гуманоидов. В игре участвуют человекоподобные роботы, собранные и запрограммированные участниками самостоятельно. Дополнительное усложнение по сравнению с современной лигой стандартных платформ связано с тем, что на робот не могут быть установлены сенсоры, отличные от органов чувств человека, например сонары или инфракрасные дальномеры.
- 6.



*Рисунок 4.3. Футбольное противостояние гуманоидов Aldebaran NAO в рамках RoboCup*

Впоследствии к футбольным в RoboCup добавились социально-значимая лига роботов-спасателей RoboCup Rescue, которая также разделяется на подлиги компьютерного моделирования и состязания аппаратных платформ. А с 2006 г. стали проводиться еще и соревнования сервисных роботов — RoboCup@Home. В лиге роботов-спасателей мультиагентные команды исследуют специально построенные лабиринты, симулирующие районы бедствий, ищут пострадавших, идентифицируют их признаки жизни и формируют карту местности с отметками мест нахождения пострадавших. RoboCup@Home фокусируется на применении роботов в повседневной жизни. В настоящее время соревновательные сценарии включают обстановку гостиницы и кухни, а в дальнейшем будут расширены на прочие типичные места, где бывает человек, например магазин или парк.

Ежегодная **Всемирная Олимпиада роботов (WRO)** официально проводится с 2004 г. В соревнованиях выделяются четыре возрастные категории: начальная (до 13 лет), средняя (13–16 лет), старшая (16–20 лет) и отдельная GEN II Football для участников от 10 до 20 лет. Миссия заключается в следующем: «Объединить молодых людей со всего мира для развития их творческих способностей, навыков проектирования и решения проблем с помощью сложных и обучающих соревнований и занятий с роботами». Было решено, что ежегодный международный финал WRO каждый год будет проводиться в разных странах. В 2004 году мероприятие было организовано Сингапуром, и в нем приняли участие 12 стран. В 2019 году, пятнадцать лет спустя, команды из 73 стран отправились в Дьёр в Венгрии на международный

финал. Сегодня соревнование, которое начиналось с небольшого турнира в нескольких азиатских странах, расширилось до более чем 85 стран на всех континентах, что сделало WRO поистине глобальным событием.

2004 —  Сингапур	2013 —  Джакарта
2005 —  Бангкок	2014 —  Сочи
2006 —  Наньнин	2015 —  Доха
2007 —  Тайбэй	2016 —  Нью-Дели
2008 —  Йокогама	2017 —  Коста Рика
2009 —  Пхохан	2018 —  Таиланд
2010 —  Манила	2019 —  Дьёр
2011 —  Абу-Даби	2020 —  Монреаль
2012 —  Куала Лумпур	2021 —  Дортмунд

Международный финал WRO 2022 пройдет в Германии, в городе Дортмунд, 17-19 ноября. Национальный организатор TECHNIK BEGEISTERT eV. Тему олимпиады организаторы обозначили как «**My Robot My friend**» (Рисунок 4.4.). В этом году ввели новую спортивную категорию, «вышибалы» и друзья-роботы. На сайте Всемирной робототехнической олимпиады (World Robot Olympiad) опубликованы правила и задания нового соревновательного сезона. <https://wro-association.org/competition/2022-season/>.



Рисунок 4.4. Эмблема олимпиады международного финала WRO 2022

В конкурсе есть регулярная, открытая или творческая номинации и футбольный чемпионат. В регулярной номинации участники должны сами собрать из конструктора LEGO и запрограммировать робота, который способен решить поставленную задачу. Эти задачи сводятся к определенным настольным манипуляциям, меняются год от года и анонсируются заранее. В открытой номинации участники самостоятельно готовят проект под определенную тематику. Футбольные соревнования в рамках WRO впервые прошли в 2010 г. В 2012 г. через сито соревнования прошло больше 17 000 команд из 30 стран,

среди которых команда физико-математического лицея № 239 из Санкт-Петербурга заняла первое место в старшей творческой категории с роботом «Грета», который умеет играть в ладунки и предназначен для социальной адаптации детей-инвалидов (Рисунок 4.5.). Также победу в прошлогоднем футбольном первенстве впервые праздновала московская команда.



Рисунок 4.5. Команда питерских чемпионов WRO 2012г., с роботом «Грета»

Возрастная категория 13-16 лет.

Победители и призёры олимпиады награждаются дипломами, ценными подарками, поощрительными призами от организаторов. Награды Олимпиады содержат символику Олимпиады и название профиля (категории). Награды по поручению Оргкомитета вручают специально приглашенные для этой цели видные деятели в сфере робототехники и представители общественности. [15]

### Соревнования по робототехнике в Германии

Существует много локальных соревнований, которые способствуют решению специализированных задач или региональные версии существующих соревнований с другими названиями. Примером такого местного соревнования в федерации NRW Германии является zdi-Roboterwettbewerb (Рисунок 4.6.). На основе FIRST Lego League ежегодно проводятся восемь соревнований. Участниками соревнований являются команды различных школ Германии.



Рисунок 4.6. Местные соревнования в Германии zdi-Roboterwettbewerb

Министерство науки Германии создало проект «Будущее через инновации NRW» для способствования детям и молодёжи увлечься естественными науками и технологиями. Соревнования роботов были центральным элементом инициативы с момента их успешного запуска в 2006 году. Ежегодно в региональных соревнованиях принимают участие около 170 школьных команд. Есть категория для смешанных команд, игры с роботами и категория для команд, состоящих только из девочек, конкурс выступлений роботов. К участию допускаются учащиеся в возрасте 10-16 лет. Соревнование роботов проводится при поддержке Hands on Technology EV, организатора First Lego League Central Europe.

(<https://mint-community.de/gruppen/zdi-roboterwettbewerb/>).

### **Категория «Игры с роботами»**

Так как в Германии плотность населения большая, то и потребление сырья высокое, поэтому восстановление сырья из отходов и экологически чистая утилизация очень важны. Это позволяет разрабатывать ресурсосберегающие технические решения или инновационные меры восстановления. В категории «Игры с роботами» команды выполняют курс заданий, посвященных экономике замкнутого цикла и переработке отходов (Таблица 4.1.). Задачи, которые даются на игровом коврике, должны быть решены. За каждую решённую задачу команда получает баллы. Максимальное суммарное количество баллов составляет 345. Задачи могут решаться в удобной для команды последовательности, и есть частичное оценивание за частичное их выполнение.

Таблица 4.1. Задачи в категории «Игры с роботами»

<b>№</b>	<b>Задача</b>	<b>Кол-во баллов</b>
1	Энергосберегающие лампы	37 баллов
2	Каршеринг	19 баллов
3	Утилизация смартфонов	36 баллов
4	Покупки в продуктовом магазине	66 баллов
5	Экологический рюкзак	80 баллов
6	Переработка пластиковых отходов	32 балла
7	Устойчивая 3D – печать	36 баллов
8	Совет по эффективности	20 баллов
9	Лекция по хранению CO <sub>2</sub>	19 баллов

При необходимости участники роботов маскируют под технику, животных, людей, используя цветную бумагу, фотографии, вырезки из журналов. Соревнование проходит с большим интересом, на позитиве.



Победителей награждают кубками, медалями и денежными призами для дальнейшего развития в области STEM.

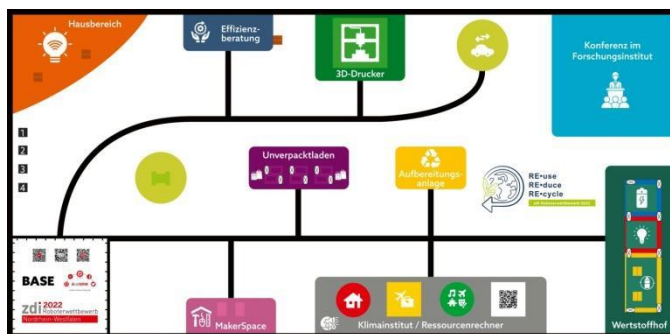


Рисунок 4.7. Коврик для игры с роботами

Ниже представлены фотографии победителей с конкурса роботов ФИНАЛ в Мюльхайме-на-Руре, 11 июня 2022 г.:

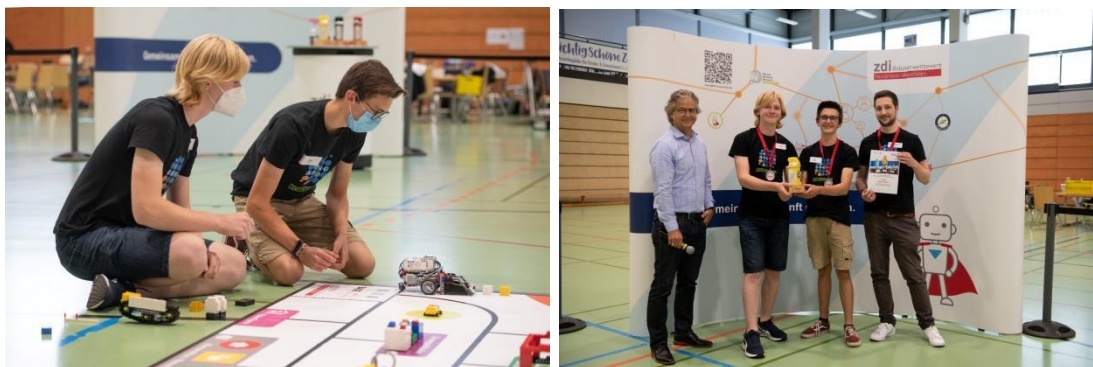


Рисунок 4.8. 1-е место команда «Робоган»

1-е место в категории «Игры с роботами» в возрастной группе средней школы досталось команде «Робоган» из средней школы ASGS в Марле. Помимо прочего, они получили денежный приз в размере 500 евро на дальнейшую деятельность в области STEM (Рисунок 4.8.).



Рисунок 4.9. 2-е место команда «abbeyRobots I»

Второе место в категории «Игры с роботами» в возрастной группе средней школы заняла команда «abbeyRobots I» из гимназии Св. Михаила в Падерборне и получили денежный приз в размере 400 евро на дальнейшую деятельность в области STEM (Рисунок 4.9.)



Рисунок 4.10. 3-е место команда «Lego-Stein-Heim1»

Третье место в категории «Игры с роботами» в возрастной группе средней школы заняла команда «Lego-Stein-Heim1» из средней школы в Штайнхайме. Денежный приз составил 300 евро на дальнейшую деятельность в области STEM (Рисунок 4.10.) [16]



Рисунок 4.11. Кубки за занятые 1, 2, 3 места в категории «Игры с роботами»

### **Международные соревнования IYRC (International Youth Robot Competition) в Корее.**

Соревнования были организованы **IYRA - Международной ассоциацией молодежной робототехники** и мэрией Тэджона.

Соревнования IYRC проводились как одно из 8 мероприятий третьего **Robot Fusion** фестиваля, в рамках которого прошли выставка корейских робот-компаний и множество робототехнических соревнований, в т.ч. гонки мобильных роботов, состязания программируемых андроидных роботов, управление квадрокоптерами.

В IYRC 2014 приняли участие около 200 команд из Австралии, Вьетнама, Индонезии, Китая, Кореи, Малайзии, России, Сингапура, Таиланда и других стран (Рисунок 4.12., Рисунок 4.13.).





Рисунок 4.12. Общее фото победителей IYRC 2014



Рисунок 4.13. Столы готовы к соревнованиям

### Категории и регламент IYRC

Как и большинство робототехнических соревнований, IYRC включает несколько категорий:

- ✓ РобоФутбол
- ✓ Доставка шара
- ✓ Лабиринт
- ✓ Гольф
- ✓ Сбор трехцветных шаров
- ✓ Робот-лайнтрекер
- ✓ Творческая категория (тема – “Мой друг, мой робот”).

Технической основой для всех категорий являются конструкторы **HUNA-MRT**. Южнокорейский бренд **HUNA – MRT (My Robot Time)** производит 6 серий робототехнического конструктора, рассчитанных на разные возрастные категории и сферы использования (Рисунок 4.14.).

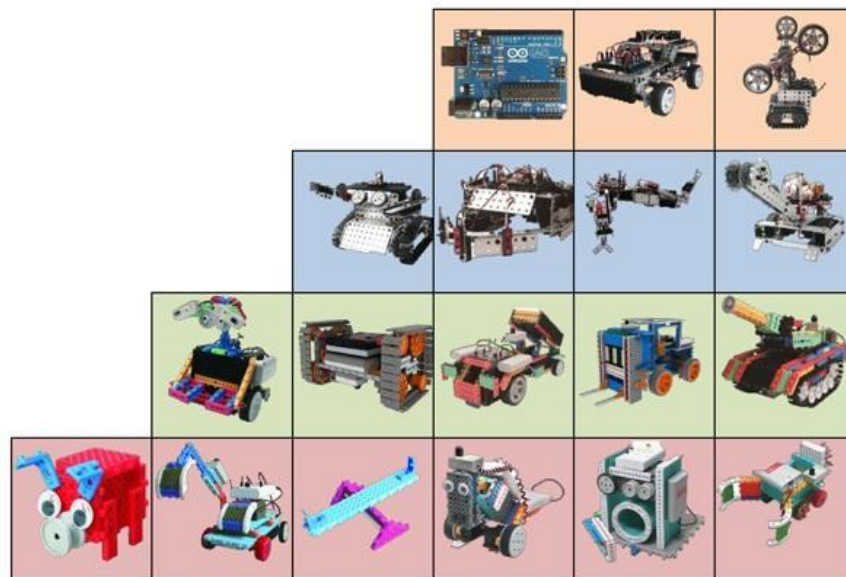


Рисунок 4.14. Наборы от простого (внизу) к сложному (вверху)



Более продвинутый набор для детей от 12 лет HUNITRONIC предназначен для сборки программируемых роботов на основе микроконтроллера Arduino Uno или его аналогов. Здесь имеется плата расширения Extension IO Shield для подключения датчиков, несколько сервомоторов, разнообразные сенсоры, червячные и реечные механические передачи (Рисунок 4.15.). Программирование и управление роботом осуществляется с помощью ПК или смартфона. Все наборы комплектуются открытым ПО с графической оболочкой. Пластиковые детали конструктора соединяются по всем 6 сторонам, что значительно расширяет возможности сборки моделей любых форм.



Рисунок 4.15. Составные части HUNITRONIC

### Категория «Робот-лайнтрекер»

Участники этой категории должны разработать программу следования по линии, выполняя некоторые задачи (Рисунок 4.16.). Участник, чей робот пройдет дистанцию быстрее всех, становится победителем. Это единственная категория IYRC (не считая творческой), где не используется пульт ДУ, и роботы работают автономно. Для возрастной категории 13-15 лет на программирование дается 2 часа. Участники имеют возможность 2 раза протестировать свою программу на реальной игре. На рисунке 4.17. представлен победитель Лайнтрекера из Малайзии. [17]



Рисунок 4.16. Поле для Лайнтрекера



Рисунок 4.17. Ng Teng Hee – победитель Лайнтрекера из Малайзии

<http://hunarobo.ru/mezhdunarodnyie-sorevnovaniya-iyrc-v-koree.html>

## Ежегодный Международный фестиваль робототехники «РобоФинист» в России

**Фестиваль "РобоФинист"** — долгожданная ступень в развитии робототехники среди детей и взрослых в России и Санкт-Петербурге. Это событие призвано собирать единомышленников, близких по увлечению людей и просто всех желающих погружаться в увлекательный мир робототехники (Рисунок 4.18.).



Рисунок 4.18. Организаторы фестиваля

Немаловажную роль играет международный статус фестиваля, это означает, что участники получают уникальную возможность делиться знаниями и общаться со сверстниками из за рубежа (Рисунок 4.19.).



Рисунок 4.19.

Также на фестивале демонстрируют умные технологии будущего, которые ребята применяют в своих проектах. Во время соревнований можно

увидеть, как умело роботы играют в футбол, спасают пострадавших, борются на ринге, бегают наперегонки, летают, проходят препятствия и даже выступают на сцене.

Соревнования проходят в более 30 категориях.

*Следование по линии:*

1. Следование по широкой линии
2. Следование по узкой линии
3. Следование по узкой линии экстремал
4. Гонки балансирующих роботов
5. Марафон шагающих роботов
6. Линия-профи
7. Эстафета
8. Слалом

*Сумо:*

1. Механическое сумо 15x15: для начинающих
2. Интеллектуальное сумо 15x15: образовательные конструкторы
3. Мини-сумо 10x10
4. Микро-сумо 5x5
5. Сумо андроидных роботов

*Большое путешествие:*

1. Большое путешествие младшая категория
2. Большое путешествие старшая категория

*Соревнования:*

1. Футбол управляемых роботов
2. Арканойд
3. Автономные воздушные аппараты
4. Практическая олимпиада по робототехнике
5. Ралли по коридору
6. Собираение шайб
7. Свободная творческая категория
8. Лунные кратеры
9. Аквароботы
10. Воздушные гонки

*RoboCup:*

1. RoboCupJunior

*Кегельринг:*

1. Кегельринг для начинающих
2. Кегельринг для продолжающих
3. Кегельринг-макро
4. Кегельринг андроидных роботов

*Лабиринт:*

1. Лабиринт для начинающих
2. Лабиринт туда и обратно

Победители в каждой категории награждаются дипломами международного уровня, образовательными конструкторами и умным оборудованием.

Важная составляющая фестиваля это симпозиум с участием ведущих специалистов в сфере образовательной робототехники. Гости фестиваля принимают участие в десятках увлекательных мастер-классах, получают уроки по 3D-моделированию, программированию, созданию собственных роботов и кибер-открыток (Рисунок 4.20.).



Рисунок 4.20. Симпозиум с ведущими специалистами робототехники.

### Категория «Воздушные гонки»

Сконструированный участником робот будет выполнять задания, не касаясь земли и воздушных препятствий (Рисунок 4.21.).

Задача - облететь полигон наибольшее количество раз. Однако это не так просто сделать, потому что траектория полета представляет собой «восьмерку».



Рисунок 4.21. Воздушные гонки

Воздушные гонки - соревнования дронов.

В командах по 1 участнику. Управление роботом осуществляется с помощью ноутбука.



Помимо офлайн соревнований, организаторы проводят онлайн соревнования по робототехнике, победители которых получают сертификаты на робототехнические конструкторы и бесплатный курс по нескучной математике (Рисунок 4.22.). [18]



Рисунок 4.22.

Крупнейшим российским молодежным робототехническим соревнованием является **Всероссийский робототехнический фестиваль — «Робофест»** (Рисунок 4.23.). Идея проведения фестиваля родилась у Олега Дерипаски в 2007 г. при посещении им Североамериканских всемирных соревнований роботов «FIRST» (For Inspiration and Recognition of Science and Technology/«Развитие и поощрение интереса к науке и технике»). В России на тот момент не было единой программы поддержки робототехники и инженерных специальностей. Возродить престиж инженерных профессий в России, которая всегда была знаменита своими физиками, математиками, изобретателями, инженерами, привить интерес к ним со школьной скамьи — ключевые задачи «РобоФеста». В 2018 г. фестиваль «Робофест» переименован в «PROFEST». Ежегодно на его площадке проводятся состязания по 19 видам, включая финалы национального отбора на престижные международные конкурсы: прозападные FIRST и Всемирную Олимпиаду роботов (WRO), о которых упоминалось выше, а также Азиатско-тихоокеанский ABU ROBOCON. «Робофест» собирает под своей крышей более 2000 участников и 500 роботов. [19]



Рисунок 4.23. Всероссийский робототехнический фестиваль — «Робофест» <http://www.russianrobofest.ru/>

Соревнования и конкурсы — важный элемент экосистемы образовательной робототехники, играющий существенную роль в мотивации участия детей и целых регионов, распространении знаний и технологий. На протяжении всего периода развития образовательной робототехники число соревнований (конкурсов) и численность их участников постоянно увеличиваются. Соревнования охватывают молодежь разного возраста (школьников, студентов) и отличаются по целям: олимпиадные, конкурсные и стимулирующие (мотивирующие).

При этом рост числа и разнообразия конкурсных мероприятий ставит вопрос о необходимости более рационального подхода к их организации для повышения эффективности расходования ресурсов: выстраивание траекторий преемственности между соревнованиями, экспертное сопоставление и др. [20]

Многие соревнования роботов используются фирмами-производителями для продвижения своих продуктов и различных устройств и не имеют каких-либо конкретных целей и задач, связанных с образованием. Участниками подобных соревнований являются компании и исследовательские институты. На вышеуказанных соревнованиях часто демонстрируются результаты, достигнутые отдельными учреждениями при реализации проектов на основе робототехники. Авторы этих проектов, как правило, являются специалистами в области информатики и робототехники. [21]

Залог успешного развития состоит в том, чтобы не гнаться слепо за всеми тенденциями, а определить конкретные направления, в которых компетенции наиболее сильны, и упорно работать над ними. Как показывает практика, развитие международных робототехнических соревнований — это инвестиция, которая окупается не сразу, но сполна. Для молодого поколения это стимул получать знания в области науки и техники, а для страны — вклад в интеллектуальное будущее. [22]

Победа — это хорошо, но поражение — это часть нашего пути.

## **Модуль 5. Спортивный психолог**

В настоящее время направление робототехники является активно развивающейся областью науки и техники. Роботы все активнее проникают в промышленность и жизнь людей. Шире становится и перечень задач, с которыми они способны справляться. По всему миру проходят соревнования роботов, школьники обмениваются знаниями, демонстрируют свои работы. Участие в такой деятельности учит юных робототехников добиваться успеха в высокотехнологичных проектах. Как и в любом спорте ребята стремятся к победе, соревнуясь своими роботами друг с другом. И конечно неизбежны психологические колебания – переживания, сомнения, апатия, неуверенность,

поэтому тренера при подготовке к соревнованиям приглашают спортивного психолога. Спортивный психолог – это специалист со знаниями психологии в области спорта, поддерживает благоприятный психологический климат в спортивном коллективе. Помогает спортсменам достигать высоких результатов за счет аутогенных тренировок. Оказывает психологическую поддержку.

### ***Как сформировать психологию победителя?***

Необходимо начать с самоанализа, постановки целей, ментальных тренировок и тренировки спортивных техник еще до основной подготовки. Это относится, кстати, не только к спортивной сфере. В основе ментальных тренировок лежит привычка к позитивному мышлению. Кроме этого, спортсмену, который рассчитывает на высокий результат, необходим весь его психологический ресурс:

- сильная внутренняя мотивация, отсутствие необходимости во внешней мотивации;
- победный настрой;
- умение проводить параллель между результатом и своей спортивной формой;
- готовность учиться на ошибках и способность выдерживать критику;
- позитивное, но реалистичное мышление, способность не сдаваться ни при каких обстоятельствах;
- контроль чувств и эмоций;
- концентрация внимания;
- стабильное состояние энергичности и бодрости;
- уверенность в себе во время соревнований;
- осознание своего потенциала и умение проявлять его в каждом выступлении.

Практически каждый спортивный психолог советует всем спортсменам поменять свое отношение к поражению или результату, который не оправдал ожиданий, и воспринимать их не иначе, как необходимый опыт на пути к более серьезным достижениям.

### **Чем еще может помочь спортивный психолог в подготовке спортсменов?**

Психологическая готовность - это "рулевое устройство", позволяющее спортсмену вести себя к победе. Способность «собраться» или мобилизоваться перед лицом грядущих вызовов дорогого стоит.

Во-первых, фактор работы мозга и шаблоны спортивного мышления и поведения не зависят от видов спорта и их специфики. Они формируются из трех, невидимых на первый взгляд, компонентов, которые сопровождают профессиональную деятельность спортсмена: психологической, интеллектуальной и эмоциональной подготовленности.

Во-вторых, в сознании и подсознании спортсмена формируются модели поведения до и во время соревнований, которые во многом зависят от волевых

качеств спортсменов, их предварительной морально-волевой и психологической подготовки к предстоящим соревнованиям.

В-третьих, взаимодействие внешнего и внутреннего мира спортсмена происходит через физическое тело и мысли, поскольку прежде, чем действие совершится в жизни, оно проживает целый сценарий в нашей голове, и только потом конечный результат начинает воплощаться в наших действиях.

Эти и ряд других вопросов находятся в зоне компетенции спортивного психолога, и в этой связи его ключевая роль заключается в разработке интегрированных программ по спортивной психологии для спортсменов, тренерского состава, группы сопровождения. Системный подход, начиная с анализа организации спортивного пространства, психологического климата, мониторинга психологического компонента тренировочной деятельности спортсменов, заканчивая комплексной психодиагностикой, тестированием и калибровкой психо-эмоционального состояния спортсмена, направлен на комплексную подготовку спортсмена высокого класса.

Мозг ничем не отличается от мышц: он должен работать, чтобы жить. Именно мозг, как дирижер, управляет каждым ударом нашего сердца, каждым движением, выработкой гормонов и другими процессами, и действиями. Соответственно, мускулы мозга нужно правильно воспитывать и тренировать.

Исследуя внешние факторы воздействия на результативность спортсмена, спортивный психолог разрабатывает индивидуальную программу работы со спортсменом, включая аутогенные и идеомоторные тренировки, направленные на формирование спортивного мышления, развитие внутренней саморегуляции и развитие навыков боевой готовности. Самым важным в процессе работы психолога является работа с нарушениями психо-эмоциональных состояний, где строго индивидуальный и тонкий подбор психотерапевтических методов коррекции направлен на диагностику и лечение нестабильных состояний спортсменов таких, как: страхи, тревожность, апатия, агрессия и тому подобное. При запущенных состояниях спортсменов, связанных внутренним психологическим давлением, наблюдаются фобии, панические атаки, сонные параличи и другие ярко выраженные отклонения.

Все в нашей голове, наши действия - следствие работы нашего мозга в его сознательной и бессознательной оболочке.

Взаимодействие психологии и спорта может значительно повысить эффективность тренировок и улучшить результаты как отдельного спортсмена, так и целой команды. [23]

#### **Полезные ссылки**

- <http://robotwars.wikia.com> — сайт боев роботов Robot Wars
- [www.theroboticschallenge.org/](http://www.theroboticschallenge.org/) — сайт конкурса DARPA Robotics Challenge
- [www.kuka-labs.com/en/innovationaward](http://www.kuka-labs.com/en/innovationaward) — сайт конкурса KUKA Innovation in Mobile Manipulation Award
- <http://robotics.sk.ru> — официальный сайт Russian Robotic Challenge фонда «Сколково»



- [www.robocup.org/robocup-rescue/](http://www.robocup.org/robocup-rescue/) — сайт конкурса Robocup Rescue
- [www.robocup.org/robocup-home/](http://www.robocup.org/robocup-home/) — сайт конкурса Robocup @Home
- [www.imaginecup.com](http://www.imaginecup.com) — официальный сайт Microsoft Imagine Cup
- [www.entrepreneurshipchallenge.org/](http://www.entrepreneurshipchallenge.org/) — официальный сайт Intel Global Challenge
- [www.bit-konkurs.ru/](http://www.bit-konkurs.ru/) — сайт конкурса Бизнес Инновационных Технологий
- [www.usfirst.org/roboticsprograms](http://www.usfirst.org/roboticsprograms) — сайт программы FIRST
- [www.bestinc.org](http://www.bestinc.org) — сайт фонда BEST
- [www.eurobot-russia.ru](http://www.eurobot-russia.ru) — российский сайт конкурса Eurobot
- [www.robocup.org](http://www.robocup.org) — официальный сайт конкурса Robocup
- <http://wroboto.ru/competition/wro> — сайт Всемирной Олимпиады роботов
- <http://archive.darpa.mil/grandchallenge05/> — архивный сайт конкурса DARPA Grand Challenge
- <http://robofest2013.ru/contests> — официальный сайт «Робофест»
- <http://robofestival.ru/> — сайт Всероссийского фестиваля мехатроники и робототехники «МиР»
- <http://robonaut.jsc.nasa.gov/> — описание проекта Robonaut

## Источники

- 1) Roberta Initiative — курсы для девочек в Германии [Электронный ресурс]. – Режим доступа: <https://www.roberta-home.de/en/>
- 2) Robbo Japan [Электронный ресурс]. – Режим доступа: [https://www.cnews.ru/news/line/2020-06-04\\_yarponiya\\_vlozhit\\_2 mln\\_v\\_sozdanie](https://www.cnews.ru/news/line/2020-06-04_yarponiya_vlozhit_2 mln_v_sozdanie)
- 3) Учебно-методический комплекс «Роботрек», Россия, г. Санкт-Петербург. Презентация, 2016 г., слайд 46, слайд 52 [Электронный ресурс]. – Режим доступа: <https://robotrack-rus.ru/wp-content/uploads/2017/06/УМК-ROBOTREK.pdf>
- 4) [Электронный ресурс]. – Режим доступа: <https://www.roboticsforschools.eu/conference>
- 5) Робототехника В России: Образовательный ландшафт. Часть 1. Д.А. Гагарина, А.С. Гагарин, 2019 г. – с. 65, с. 68
- 6) Учебно-дидактического комплекс «Образовательная робототехника» для 5-9 классов, составитель Белов Р.А., г. Усть-Илимск, Россия. [Электронный ресурс]. – Режим доступа: <https://infourok.ru/uchebno-didakticheskij-kompleks-obrazovatel'naya-robototekhnika-dlya-zanyatij-s-uchashimisya-5-9-klassov-5177845.html>
- 7) Актуальные вопросы и тенденции развития предметной области "технология". Сборник материалов II Международной научно-практической конференции. Отв. редактор М.Г. Корецкий, сост. Н.П. Шпаков. Москва, 2021. Издательство: Общество с ограниченной ответственностью "Онтопринт" (Москва) [Электронный ресурс]. – Режим доступа: <https://www.elibrary.ru/item.asp?id=46312165>
- 8) Бакалаврская работа, 44.03.01 – Педагогическое образование. Соревновательные мероприятия в процессе обучения робототехнике обучающихся 6-х классов. П.Е. Неудачин, 2017 г. 13-14 сс., 54-56 сс.
- 9) СОР и СОЧ: что это такое и для чего их придумали? [Электронный ресурс]. – Режим доступа: <https://informburo.kz/stati/soch-i-sor-chto-eto-za-sistema-i-pochemu-eyu-nedovolny-roditeli-shkolnikov.html>
- 10) Презентация Best Robotics Boosting Egeineering, Science & Technology. июль, 2021 г., слайды 11 - 13. [Электронный ресурс]. – Режим доступа: <https://www.bestrobotics.org/site/documents/Competition%20Overview.pdf>
- 11) WRO RoboMission Senior Game Rules Season 2022. [Электронный ресурс]. – Режим доступа: <https://wro-association.org/wp-content/uploads/WRO-2022-RoboMission-Senior.pdf>
- 12) Российская робототехническая олимпиада 2022 «Робот – друг человека». Нижний Новгород, 2022 г. Общие правила. 3 с., 13 с., 16-17 сс.
- 13) RoboCupJunior 2022. [Электронный ресурс]. – Режим доступа: <https://junior.robocup.org/>
- 14) Никитина Е.Н. Доклад по теме самообразования: «Формирование проектной деятельности обучающихся в условиях реализации ФГОС

- средствами робототехники». Россия, Ленинградская обл., Копорье, 2017 г., 4 с.
- 15) «Чемпионаты роботов». А. Бобцов, д. т. н., НИУ ИТМО; С. Колюбин, к. т. н., ООО «Икстурион» [Электронный ресурс]. – Режим доступа: <https://controleng.ru/innovatsii/chempionaty-robotov/>
  - 16) Zdi-Roboterwettbewerb [Электронный ресурс]. – Режим доступа: <https://mint-community.de/gruppen/zdi-roboterwettbewerb/>
  - 17) Международные соревнования IYRC в Корее. [Электронный ресурс]. – Режим доступа: <http://hunarobo.ru/mezhdunarodnyie-sorevnovaniya-iyrc-v-koree.html>
  - 18) Регламенты и инструкции портала РобоФинист [Электронный ресурс]. – Режим доступа: <https://robofinist.ru>
  - 19) Всероссийский технологический фестиваль «Робофест», Москва [Электронный ресурс]. – Режим доступа: <http://www.russianrobofest.ru/>
  - 20) Никитина Е.Н. Доклад по теме самообразования: «Формирование проектной деятельности обучающихся в условиях реализации ФГОС средствами робототехники». Россия, Ленинградская обл., Копорье, 2017 г., 4 с.
  - 21) Филиппов В.И. Дисс. на соиск. Ученой степени канд.тех.наук «Методика использования робототехники для формирования универсальных учебных действий у обучающихся во внеурочной деятельности по информатике», Москва, 2020 г. [Электронный ресурс]. – Режим доступа: <https://www.dissercat.com/content/metodika-ispolzovaniya-robototekhniki-dlya-formirovaniya-universalnykh-uchebnykh-deistvii/read>
  - 22) «Чемпионаты роботов», А. Бобцов, д. т. н., НИУ ИТМО; С. Колюбин, к. т. н., ООО «Икстурион» [Электронный ресурс]. – Режим доступа: <https://controleng.ru/innovatsii/chempionaty-robotov/>
  - 23) Психология победителя. Советы от спортивных психологов. [Электронный ресурс]. – Режим доступа: <https://olympic.kz/ru/article/12968-psikhologiya-pobeditelya-sovety-ot-sportivnykh-psikhologov>

## 2. Актуальные вопросы преподавания курса "Робототехника"

Образование в области STEM, в том числе робототехнике является основой подготовки специалистов в области высокотехнологичной индустрии. Раннее обучение основам программирования и робототехники в средней школе выступает как пропедевтика к привлечению обучающихся к инженерным специальностям, роботостроению и инженерно-техническим наукам.

Как бы не считалась «Робототехника», отдельным предметом или частью учебного предмета «Информатика», процесс обучения основам робототехники в школе построен на принципах традиционной методической системы обучения учебному предмету. В этом контексте можно выделить ряд факторов, оказывающих значительное влияние на внедрение робототехники в казахстанские школы, и образовательный процесс по робототехнике в целом.

Первое, что можно отметить, внедрение робототехники в общеобразовательную школу, включая начальную школу является результатом государственной политики в области образования и науки Республики Казахстан.

Анализ, проведенный в рамках Государственной программы "Цифровой Казахстан" на 2018-2022 годы показывает, что уровень компьютерной (цифровой) грамотности населения Казахстана составляет 76,2% и отмечается необходимость его повышения в ближайшие годы (<https://adilet.zan.kz/rus/docs/P1700000827>). В связи с этим пересмотр содержания среднего образования через развитие креативного мышления и технических навыков определен как один из приоритетных направлений для достижения целей и задач программы "Цифровой Казахстан" на 2018-2022 годы. Программа также предусматривает поэтапного внедрения в начальную школу предмета "Основы программирования", начиная со 2-го класса и актуализацию содержания учебной программы по информатике для 5-11-го классов, в первую очередь, в части пересмотра языков программирования с учетом включения STEM-элементов (робототехника, виртуальная реальность, 3D-принтинг и другие).

Следующим немаловажным фактором является инновационный характер образовательной робототехники, как и в содержательном, так и научно-технологическом аспекте. В условиях стремительной цифровизации общества и всевозможных сфер человеческой деятельности цифровая трансформация образовательной системы выступает основой концепции современного образования и подходов к его организации. Для современного человека образование представляет собой не просто знания, умения и навыки, но и целый ряд необходимых компетенций для конкурентоспособной личности в 21 цифровом веке. В настоящее время,

все более широкое распространение приобретают автоматизированные системы, роботизированные комплексы и устройства, избавляющие человека от многих рутинных и монотонных видов деятельности не только в производстве, но и обычной, повседневной жизни. В современном мире умение использовать возможности цифровых технологий становится жизненно необходимым практическим навыком для человека. Поэтому, главной целью изучения базовых элементов робототехники в средней школе является формирование таких качеств современной личности, как цифровая грамотность, вычислительное мышление, цифровая культура и другие. Робототехника, как прикладная инженерная наука отдельно или в интеграции с новейшими технологиями, например, виртуальной и дополненной реальностью, голопортацией и голограммой, Big Data и искусственным интеллектом будет вносить свой вклад в развитие образования. В этом плане робототехника в школе является современным междисциплинарным направлением в обучении, интегрирующим знания по STEM предметам и позволяющим вовлечь учащихся разного возраста в процесс инновационного научно-технического творчества. Также, следует отметить, что образовательные программы зарубежных стран с высокими образовательными достижениями обучающихся школ такие как, Сингапур, Финляндия, Южная Корея и другие показывают ценность поэтапного и долгосрочного внедрения инновационных цифровых технологий в систему образования.

Внедрение робототехники в школу инициирует проявление еще одного важного фактора - социально-экономического, требующего внимательного подхода. Социально-экономический фактор подразумевает создание необходимых условий для организации массового обучения основам робототехники в школе: готовность педагогических кадров к обучению детей основам робототехники и программирования; обеспечение школы материально-техническим оборудованием для проведения уроков по робототехнике; разработка учебников, учебно-методических пособий, методов обучения по образовательной робототехнике и другие.

В большинстве своем, актуальные вопросы преподавания курса "Робототехника" обусловлены выше перечисленными факторами. Кроме того, в целях системности и последовательности, актуальные вопросы преподавания курса «Робототехника» в средней школе рассматриваются в следующих контекстах в соответствии с методической системой обучения предметам в школе:

- Вопросы совершенствования и устойчивости содержания школьного курса робототехники
- Материально-техническое и аппаратно-программное обеспечение курса "Робототехника» в средней школе

- Учебные среды управления образовательными роботами (среды программирования) для средней школы
- Цифровые ресурсы для преподавания курса "Робототехника" в средней школе
- Влияние подготовленности учителей на качество преподавания робототехники
- Актуальные вопросы методики преподавания робототехники и организации процесса обучения робототехнике в средней школе

Как известно, компьютерные науки (Computer Science) - это проектирование и разработка автоматизированных вычислительных систем для решения задач. Вычислительное мышление является основой понимания компьютерной науки и освоения вычислительных (или цифровых) навыков. Вычислительное мышление - это мыслительные процессы, участвующие в формулировании проблемы и выражении ее решения таким образом, что компьютер (или робот, исполнитель) может эффективно ее выполнить. Вычислительное мышление включает в себя как навыки алгоритмического, пространственного мышления, так и методы решения проблем. Применение вычислений к решению проблем обычно приводит к созданию вычислительных систем, представителями которых являются различные виды компьютеров (персональные компьютеры, ноутбуки, планшеты, смартфоны, системы виртуальной и дополненной реальности), высокотехнологичные цифровые системы промышленного назначения (бытовая техника, роботы), программные приложения (электронные таблицы, поисковые системы, веб-сайты и все приложения, которые запускаются на цифровых устройствах) и другие.

Цифровая грамотность, в отличие от компьютерной науки, относится к способности человека использовать компьютеры и программные приложения, спроектированные и разработанные с использованием компьютерной науки, например, для поиска, оценки, создания, передачи информации в повседневной жизни. Кроме того, цифровая грамотность также включает в себя такие вопросы: «Как компьютерные технологии влияют на общество (например, конфиденциальность и безопасность информации)?», «Как организовать совместную работу и исследования с использованием приложений и других цифровых инструментов?», «Как использовать информационно-вычислительные системы для личной жизни (например, для получения услуги электронного правительства, образовательные, медицинские, банковские и другие услуги)?».

Вопросы, касающиеся формирования вычислительных навыков и развития вычислительного мышления, также привлекают внимание практиков – учителей школы. Особенно настораживает практиков ухудшение вычислительных навыков у выпускников средней школы, так

как это приводит к снижению вычислительной культуры обучающихся основной школы и математической культуры выпускников средней школы, и как следствие – их неготовность получать дальнейшее качественное образование. Учитель математики средней школы № 519 г. Санкт-Петербурга Смирнова А. А. отмечает, что в учебниках по математике для средней школы развивающие упражнения зачастую доминируют над заданиями с вычислительным компонентом, в результате такого построения учебного материала многие умения и вычислительные навыки у учащихся общеобразовательных школ остаются не полностью сформированными. Автор рекомендует включить в учебные пособия для школ основные (базовые) вычислительные задания по теме, усвоение которых должно быть полным и целенаправленным. По мнению автора, не следует недооценивать значение решения вычислительных заданий для воспитания учащихся, так как безошибочное и аккуратное выполнение математических вычислений формирует трудолюбие, что переходит постепенно в добросовестное отношение к любой работе [1].

Реалии современной информационной эпохи показывают, что содержание школьного образования, в том числе научного и естественно-математического образования, нуждается в значительном пересмотре. Повсеместному обязательному обучению основам программирования в общеобразовательной школе, включая начальное образование также способствовал такой фактор, как появление многочисленных доступных и простых инструментов для обучения программированию в разном возрасте. Программирование считается наиболее эффективным инструментом, с помощью которого обучение становится более интересным, быстрым, точным и простым, а знания и навыки, полученные в процессе обучения, являются научными, глубокими и обобщенными. Именно поэтому во многих странах мира начали активно внедрять в учебные планы школы учебные предметы, обучающие основам программирования.

### *Драйверы развития робототехники в образовании*

Этапы развития образовательной робототехники отличаются не только масштабом развития области (количество кружков, охват детей, охват регионов), но и различием основных акторов и драйверов этого процесса, полнотой элементов экосистемы (соревнования, профессиональные сообщества, профильные медиа).

Европейские исследователи выделяют ряд стейкхолдеров — заинтересованных сторон, определяющих содержание и развитие образовательной робототехники:

- учащиеся — те, кто непосредственно обучается робототехнике или участвует в мероприятиях, связанных с образовательной робототехникой;

- родители как мотиваторы занятия детей робототехникой;
- школы как места, где происходит обучение, включая:
- учителей, проводящих обучение;
- администрацию, принимающую решение об обучении, бюджете и программах;

- коммерческие и некоммерческие организации дополнительного и неформального образования (клубы, кружки, центры и т. п.), предлагающие программы по робототехнике;

- университеты, разрабатывающие и предоставляющие технологии и методы для робототехники и образования;

- промышленность как будущий работодатель;

- правительственные и другие организации, определяющие политику в области образования.

Казахстанский список стейкхолдеров, на наш взгляд, практически совпадает с европейским, кроме того, что значимым участником процесса становятся производители оборудования и конструкторов для обучения робототехнике.

#### *Технологические компании и бизнес*

Еще одним значимым драйвером развития робототехнического образования на определенном этапе становятся технологические компании и бизнес, которые благодаря автоматизации и роботизации производств могут получить экономический эффект, выйти на новые рынки. В конечном итоге именно корпорации, будучи заинтересованными в разработке технологических решений, автоматизации, интеграции в производственные процессы роботов, предъявляют требования к кадрам.

Технологические компании развивают направления школьной и студенческой робототехники самостоятельно или косвенно через участие в качестве партнеров соревнований. Они заинтересованы во влиянии на образовательные программы и подготовке востребованных на их предприятиях кадров, актуальных для них исследовательских работах.

#### *Соревнования и конкурсы по робототехнике*

Соревнования — значимый элемент системы робототехнического образования. Их количество и разнообразие растет. Они дифференцированы по возрасту, уровню подготовки, используемому оборудованию.

Типы и задачи соревнований. Соревновательная и спортивная робототехника



Соревнования по робототехнике можно классифицировать по различным основаниям. Нам представляется оптимальным выделение среди прочих следующих трех подходов.

Первый подход — условно «олимпиадный», когда ответ или решения конкурсных заданий состязаний заранее известны (WorldSkills, JuniorSkills, ряд направлений WRO и «РобоФеста»).

Второй подход — условно «конкурсный», когда эталонного ответа на поставленную соревнованием задачу не существует (EUROBOT, часть направлений RoboCup, ABU Robocon, хакатоны). Этот подход в большей степени способствует развитию креативности участников, стимулирует научные исследования и технологические разработки. Этому также способствует снятие ограничений на платформы и некоторые другие искусственные ограничения.

Третий подход — стимулирующий (мотивирующий), когда соревнования являются в большей степени праздником, игрой, общением (FLL, Junior FLL, «РобоФинист», часть региональных, городских, районных соревнований). Задачей таких соревнований могут быть выявление талантливых учеников, обмен опытом, мотивация участников на дальнейшее обучение, навыки управления и работы в команде.

## *2.2. Материально-техническое и аппаратно-программное обеспечение курса "Робототехника" в школе*

Неотъемлемой частью методической системы обучения учебному предмету в школе является, как и средства обучения, так и среда обучения. Поэтому материально-техническое и аппаратно-программное обеспечение курса "Робототехника" в школе, может быть одним из наиболее актуальных вопросов преподавания данного курса «Робототехника» в начальной школе.

Согласно обновленной методике, которая была использована в Национальном докладе о состоянии и развитии системы образования Республики Казахстан (по итогам 2020 года) одним из новых показателей в индексе материально-технической базы (МТБ) и информационно-коммуникационных технологий (ИКТ) является доля учебных кабинетов, оснащенных новым оборудованием. В 2020 г. Данный показатель составил 9,6%. по всей республике, в частности из всех учебных кабинетов оснащено современным оборудованием 14900 кабинетов. Кроме того, анализ индекс МТБ И ИКТ и ресурсообеспеченности школьного образования показывают на неравенства материально-технической обеспеченности школ регионов в стране [3] (Рисунок 2.5, 2.6).

**Таблица 3.3.31. Индекс МТБ И ИКТ школьного образования по показателям группы D**

	2020			2019	
	Позиция	Значение, балл	Разница с 2019, балл	Позиция	Значение, балл
ВКО	1	0,836	-0,004	3	0,840
Павлодарская	2	0,820	-0,075	1	0,895
СКО	3	0,795	-0,018	4	0,813
г. Нур-Султан	4	0,793	+0,052	9	0,741
Актюбинская	5	0,780	+0,029	7	0,751
г. Алматы	6	0,774	-0,091	2	0,865
Карагандинская	7	0,758	+0,010	8	0,748
Атырауская	8	0,730	+0,179	17	0,551
Жамбылская	9	0,714	-0,068	5	0,782
Акмолинская	10	0,712	-0,001	10	0,713
Костанайская	11	0,694	+0,015	13	0,679
Кызылординская	12	0,691	-0,070	6	0,761
Туркестанская	13	0,687	+0,072	15	0,615
ЗКО	14	0,669	-0,026	11-12	0,695
Алматинская	15	0,649	+0,003	14	0,646
Мангистауская	16	0,644	-0,051	11-12	0,695
г. Шымкент	17	0,570	+0,016	16	0,554

**Рисунок 2.5 – Индекс МТБ И ИКТ школьного образования в РК**

*Источник: Национальный доклад о состоянии и развитии системы образования Республики Казахстан (по итогам 2020 года).- Нур-Султан: Министерство образования и науки Республики Казахстан, АО «Информационно-аналитический центр», 2021. - 310 стр.*

**Таблица 3.3.32. Индекс сводного показателя «Ресурсообеспеченность»**

	2020			2019	
	Позиция	Значение, балл	Разница с 2019, балл	Позиция	Значение, балл
г. Нур-Султан	1	0,668	+0,025	3	0,643
Павлодарская	2	0,645	-0,044	2	0,689
СКО	3	0,617	+0,014	5	0,603
г. Алматы	4	0,605	-0,091	1	0,696
Актюбинская	5	0,594	+0,026	7	0,568
ЗКО	6	0,590	+0,122	13	0,468
Мангистауская	7	0,572	+0,131	14	0,441
Атырауская	8	0,571	+0,156	16	0,415
Карагандинская	9	0,537	-0,009	9	0,546
Жамбылская	10	0,536	+0,014	11	0,522
Костанайская	11	0,534	-0,011	8	0,545
Кызылординская	12	0,534	-0,091	4	0,625
Акмолинская	13	0,526	+0,017	12	0,509
г. Шымкент	14	0,512	+0,089	15	0,423
ВКО	15	0,510	-0,074	6	0,584
Алматинская	16	0,463	-0,069	10	0,532
Туркестанская	17	0,431	+0,018	17	0,413

**Рисунок 2.6 – Индекс ресурсообеспеченности школьного образования в РК**

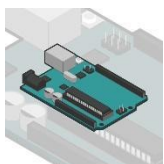




*Источник: Национальный доклад о состоянии и развитии системы образования Республики Казахстан (по итогам 2020 года).- Нур-Султан: Министерство образования и науки Республики Казахстан, АО «Информационно-аналитический центр», 2021. - 310 стр.*

Создание специализированной лаборатории для проведения занятия по робототехнике и блочному программированию может быть сложной задачей для начальной школы, так как эта работа требует не только материально-технического или аппаратно-программного обеспечения, но и также необходимо изучить особенности оборудования для начальной школы и успешных опытов в этом направлении. Тем не менее, в соответствии с требованиями Normы оснащения оборудованием и мебелью организации начального образования для кабинета информационно-коммуникационных технологий в начальной школе предусмотрены следующие наборы робототехники (<https://adilet.zan.kz/rus/docs/V1600013272>):

- робототехнический набор, содержащий конструкционные элементы, микроконтроллер, моторы, датчики, запасные части, зарядное устройство и методическое пособие
- робототехнический набор, содержащий конструкционные элементы, микроконтроллер, картридер, датчики и методическое пособие
- робототехнический набор для тренировок и соревнований, содержащий программируемый контроллер, радиомодуль, пульт дистанционного управления, датчики, конструкционные и электрические элементы и методическое пособие
- стартовый набор для развития навыков программирования, содержащий игровое поле, программируемую робот-игрушку, карточки с заданиями и методическое пособие
- 3D Ручка с набором пластика и методическое пособие
- набор конструкторский для изучения принципа работы механизмов, основ механики, состоящий из конструкционных элементов, фигурок и объектов, технологических карт и методического пособия
- набор конструкторский для изучения основ инженерных наук, возобновляемых источников энергии, пневматики, состоящий из конструкционных элементов, объектов, приборов, технологических карт и методического пособия

Доступные платформы образовательных роботов по типу применения (по уровням образования) и структуре комплектации отличаются (Таблица 2.2)

Таблица 2.2 – Платформы образовательных роботов по типу применения (по уровням образования) и структуре комплектации

Платформа робототехники	Начальная школа	Средняя школа	Старшая школа	Графический язык	Текстовый язык	Виртуальный вариант	
 Arduino (Shield-Bot)			<input type="checkbox"/>		<input type="checkbox"/>		<a href="https://www.parallax.com/education/robotics/">https://www.parallax.com/education/robotics/</a>
 LEGO EV3	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		<input type="checkbox"/>	<a href="https://education.lego.com/ru-ru/downloads/mindstorms-ev3/software">https://education.lego.com/ru-ru/downloads/mindstorms-ev3/software</a>
 LEGO SPIKE Prime	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		<input type="checkbox"/>	<a href="https://education.lego.com/ru-ru/products/-lego-education-spike-prime/45678#spike%E2%84%A2-prime">https://education.lego.com/ru-ru/products/-lego-education-spike-prime/45678#spike%E2%84%A2-prime</a>
 VEX IQ	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>			<a href="https://www.vexrobotics.com/iq">https://www.vexrobotics.com/iq</a>
 VEX V5		<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		<a href="https://www.vexrobotics.com/v5">https://www.vexrobotics.com/v5</a>

Наиболее популярным конструктором для организации занятий по робототехнике в казахстанских школах является различные версии конструктора LEGO MINDSTORMS.

Конструкторы LEGO MINDSTORMS выпускаются с 1998 года и широко распространены во многих странах мира. Высокое качество деталей конструктора LEGO сочетается с достаточной прочностью, безопасностью, простотой сборки, не требующей специальных инструментов. Системы программирования конструкторов адаптированы для соответствующего возраста детей. На данный момент выпускают два основных набора:

LEGO MINDSTORMS NXT (ПервоРобот) - это образовательный робототехнический конструктор, программируемый в среде NXT-G или Robolab. Благодаря программируемому блоку, исполнительным механизмам и разнообразным датчикам, возможно, конструировать роботов, автоматизированные системы и автономные лабораторные установки. Набор включает в себя:

- Программируемый 32/64-битный блок с беспроводным соединением через Bluetooth и USB-порт (Рисунок 2.7). Программируемый матричный дисплей. Четыре входных и три выходных порта. Шесть подключаемых цифровых платформ. Громкоговоритель мощностью 8 кГц. Можно вводить непосредственно, вводить в блок задания для выполнения простых команд. Еще больше дополнительных заданий и инструкций можно найти в ПО. Работает на шести батарейках AA или от аккумуляторной батареи.






Рисунок 2.7 – Программируемый блок NXT

- Стандартный набор датчиков для микрокомпьютера NXT: датчик освещенности (Рисунок 6), содержит в себе красный светодиод, который можно включить и выключить программно, а также фототранзистор, который, собственно и измеряет яркость попадающего на него света. Включенная подсветка позволяет измерять свет, отражённый от

поверхности объекта, в то время как при выключенной подсветке фотоэлемент измеряет яркость окружающего освещения; датчик звука, позволяет измерять громкость звука, или уровень звукового давления (SPL), измеряется в единицах, называемых Децибелами (дБ); датчик касания, это самый простой датчик, позволяющий сообщать контроллеру о нажатии и отпуске оранжевой кнопки, расположенной в передней части датчика; ультразвуковой датчик; датчик, имеющий собственный микропроцессор. Наличие собственного «интеллекта» позволяет датчику измерять и сообщать основному блоку расстояние в абсолютных единицах, а не в каких-то абстрактных масштабных величинах, как это делают датчики звука и освещённости. Датчик работает по принципу сонара, посылая короткий импульс на частоте 40КГц. Затем он измеряет время, за которое звук дошёл до объекта, отразился от него и вернулся назад; четыре сервопривода. В сервопривод встроен датчик вращения (датчик оборотов). Что дает возможность роботу двигаться точно в заданном направлении. Этот датчик измеряет обороты мотора в градусах (точность  $\pm 1^\circ$ ). Скорость вращения привода зависит от условной мощности, подающейся на сервомотор. Мощность варьируется от -100 до 100 (Таблица 2.3).

Таблица 2.3 – Стандартный набор датчиков для микрокомпьютера NXT:

Датчики	Название и функции
	Датчик освещенности
	Датчик звука

	<p>Датчик касания</p>
	<p>Датчик расстояния</p>
	<p>Сервопривод</p>

Микрокомпьютер EV3 имеет программный интерфейс, позволяющий создавать программы и настраивать регистрацию данных непосредственно на микрокомпьютере EV3 (Рисунок 2.8).



Рисунок 2.8 – Микрокомпьютер EV3

Микрокомпьютер EV3 поддерживает функции Bluetooth и WiFi для связи с компьютерами и его возможности во много раз превосходят возможности микроконтроллера NXT, микроконтроллер EV3 также совместим с мобильными устройствами, и питается обычными батареями. Использование Робот- конструктора LEGO Mindstorms с учетом возрастных особенностей младших школьников позволяет обучающимся начальных классов возможность приобретать важные знания, умения и навыки в процессе обучения робототехнике и блочному программированию.

Кроме того, для раннего обучения, например, для детей дошкольного возраста или обучающихся 1-2 классов начальной школы также рекомендуется роботы, которые можно обучать основам алгоритмизации и программированию без экрана и текста.

При выборе робототехнический набора важными критериями выступают применимость данного оборудования для максимального образовательного эффекта и выступления на соревнованиях и конкурсах в робототехнике. К настоящему времени принятых международных стандартов или единых требований к выбору образовательной робототехники с соответствующими критериями отсутствуют, однако учителями-практиками, методистами и специалистами разрабатываются ряд критериев для оценки полезности образовательной робототехники, например, критерии оценки образовательной робототехники, разработанные Ассоциацией участников технологических кружков Российской Федерации. Критерии оценивания образовательной робототехники по А. Овсянникову и др. также раскрывают актуальные проблемы использования образовательной робототехники не только в дополнительном образовании, но и в целом учебном процессе обучения робототехнике и блочному программированию в школе. Критериями оценки выступают: доступность в приобретении, стоимость расходных материалов, взаимозаменяемость с другим оборудованием, длительность функционирования, устойчивость механическим повреждениям (вандалоустойчивость), доступность сервисов по обслуживанию и ремонту, обеспеченность методическими материалами, распространенность среди школ и другие (Таблица 2.4). Следует отметить, что эти показатели для оценки полезности образовательной робототехники являются рекомендательными и разработаны для российских школ.



Таблица 2.4– Критерии оценивания образовательной робототехники по А. Овсянникову и др.

Наименование критерия	Оценка критерия		
Стоимость оборудования	<p>стоимость оборудования завышена (относительно аналогов с подобными характеристиками и функционалом) и/или его приобретение возможно только под специальные проекты / гранты / программы развития</p>	<p>стоимость оборудования позволяет оборудовать одно рабочее место или приготовить одну команду в среднестатистическом региональном учебном заведении. Оборудование слишком дорогое, чтобы покупать или обновлять его каждый год/сезон</p>	<p>стоимость оборудования позволяет оборудовать рабочий кабинет на несколько команд, участников или рабочих мест почти в любом учебном заведении. Оборудование достаточно дешевое и может быть приобретено или обновлено раз в год/сезон</p>
Доступность в приобретении	<p>оборудование невозможно приобрести за счет бюджетных средств через систему государственных закупок. Приходится заказывать его из-за границы, получать по специальным программам развития. Приобретение оборудования сопряжено с большими трудностями и возможно раз в несколько лет/сезонов)</p>	<p>покупатель может приобрести оборудование только у одного поставщика. Процесс приобретения связан с трудностями в доставке, в сроках поставки (очередь на заказ, долгая доставка из-за границы, долгая комплектация</p>	<p>оборудование широко распространено, во всех регионах есть несколько поставщиков или крупные продающие сети. Покупатель может выбрать наиболее подходящий ему вариант поставок и продавца</p>
Стоимость расходных материалов	<p>оборудование не выполняет свою практическую функцию без наличия дополнительных</p>	<p>цена расходников позволяет применять оборудование для выполнения ключевых функций при</p>	<p>цена расходников позволяет активно использовать оборудование на каждом занятии, в том</p>

лов	расходных материалов, стоимость которых сопоставима со стоимостью самого оборудования	необходимости, но не позволяет активно экспериментировать. Секция может позволить себе приобретение ограниченного количества расходных материалов раз в год/сезон, и далее приходится их экономить и бережно использовать	числе и для «экспериментов», не опасаясь их перерасхода в случае неудачного выполнения оборудованием своих функций. Или расходники широко распространены и представлены широкой номенклатурой (часто на занятиях можно использовать более дешевые расходники). Или оборудование не предусматривает расходников (содержит их в комплекте поставки) на весь срок эксплуатации до морального или физического устаревания
Взаимозаменяемость с другим оборудованием	расходники/детали/комплектующие оборудования выполнены по проприетарным стандартам и подходят только для него. Например, специализированные картриджи, фирменные крепежи и разъемы, материал для заготовок только определенной марки/фирмы	оборудование имеет расходники/детали/комплектующие, общие с подобным специализированным оборудованием или с другими моделями от того же производителя (например, специализированные для 3D-принтеров драйверы шаговых моторов, фирменные блоки питания от одного производителя для всех его серий оборудования)	в оборудовании используются унифицированные, широко распространенные расходники/детали/комплектующие. Например, стандартные приводы/электронные модули/сенсоры/крепежи, детали наборов Lego Education совместимы и заменяемы с любым другим набором Lego и т.д.
Длительность функционирования	оборудование с высоким уровнем вероятности выхода	оборудование способно выполнять свою функцию в течение	оборудование способно выполнять свою функцию в

<p>нирован ия, при которой решаютс я поставле нные задачи</p>	<p>из строя уже в течение первого года/сезона при его активном использовании в кружке или секции. Или оно предпо- лагает однократное применение в образова- тельных проектах</p>	<p>одного-двух лет/сезонов при активном его исполь- зовании в кружке или секции. Или обо- рудование позволяет подготовить один-два образовательных проекта, далее оно с большой вероятностью выйдет из строя или перестанет выпол- нять образовательную функцию</p>	<p>течение нескольких лет/сезонов при его активном использовании в кружке или секции. Или на оборудо- вании можно подготовить множество образовательных проектов без его выхода из строя</p>
<p>Вандало устойчи вость</p>	<p>оборудование может перестать функциони- ровать при использовании штатных команд/ операций, или достаточно бытового неак- куратного взаимодействия с оборудованием (падения с высоты стола или из рук ребенка, разряда статического электричества, усилий ребенка для соединения/разъедине ния деталей)</p>	<p>возможен выход из строя оборудования при совпадении нескольких предельных параметров работы или длительной работе на предельных условиях (напри- мер, одновременное включение всех приводов на максимальную мощность выводит из строя питание)</p>	<p>для выведения оборудования из строя не- обходимо выполнить нестандартные опе- рации, превысить допустимые параметры взаимодействия с ним (усилия, нагрузки, температуры, напряжения). Оборудование способно работать в секции в течение нескольких лет/сезонов без значительного ухудшения своей функциональности</p>
<p>Удобств о многокр атного использо вания</p>	<p>одна единица оборудования не позволяет организовать процесс обучения или под- готовки к соревнованиям для нескольких человек/команд. В начале сезона или</p>	<p>оборудование позволяет организовать учебный процесс для нескольких детей или подготовку нескольких команд, но имеет какие- либо ограничения в этой области. К таким ограничени- ям могут относиться:</p>	<p>оборудование позволяет организовать учебный процесс для большого количе- ства детей в плотном графике занятий или организовать подготовку к соревно- ваниям сразу нескольких команд на</p>

	<p>этапа обучения подразумевает жесткая привязка оборудования к ученику или команде, изменение этой привязки проблематично (например, требует долгой пересборки конструкции, перекладки проводки или пересчета большого количества параметров)</p>	<p>необходимость длительной сборки/разборки работа, недоступной на каждом занятии; изменения прошивки или комплектности для работы с определенным возрастом / уровнем учеников, что не позволяет использовать одну и ту же единицу оборудования в учебном процессе с учениками, находящимися на разных уровнях подготовки и т.д.</p>	<p>одной единице техники. Например, конструктор подразумевает быструю сборку и разборку работа, которую ученики успевают выполнять на каждом занятии. Или оборудование не подразумевает изменение конструкции, а подготовка к соревнованиям разными командами подразумевает только перепрограммирование</p>
<p>Уровень поддержки</p>	<p>после выпуска оборудования к нему не выходят (или выходят со значительным опозданием, более нескольких месяцев) даже критические обновления прошивок и программного обеспечения. О проблемах и недостатках в оборудовании известно (например, озвучено на форумах), но разработчик не может их оперативно исправить. Поддержка пользователей отсутствует в России или производится в течение очень длительного времени, превышающего срок</p>	<p>на оборудование выходят (или планируются к выпуску) обновления прошивок, программного обеспечения, достаточные для исправления критических ошибок и поддержания функционирования. Поддержка пользователей присутствует в одном или нескольких регионах/центрах, позволяет получить консультации по электронной почте</p>	<p>на оборудование регулярно и в течение долгого времени выходят (или планируются к выпуску) обновления прошивок, программного обеспечения, адаптация к новым инструментам разработки (языкам программирования, библиотекам и т.д.). Поддержка пользователей доступна в большинстве регионов, позволяет получить консультацию по электронной почте или телефону. Поддержка охватывает техническую документацию, инструкции по наладке и использованию</p>

	подготовки проекта или участия в соревнованиях		оборудования
Сервис обслуживания	сервисное обслуживание отсутствует в России или производится в течение очень длительного времени, превышающего срок подготовки проекта или участия в соревнованиях	сервисное обслуживание присутствует в одном или нескольких регионах/центрах, позволяет получить ремонт, обслуживание или замену неисправных деталей в течение нескольких недель	сервисное обслуживание доступно в большинстве регионов, позволяет получить ремонт, обслуживание или замену неисправных узлов и модулей в течение короткого периода времени, достаточного для оперативного продолжения занятий без серьезных пауз
Доступность ремонта	специализированный ремонт от производителя недоступен в России. Самостоятельный ремонт затруднен отсутствием проприетарного инструмента или документации	специализированный ремонт от производителя доступен только в одном или нескольких центрах. Самостоятельный ремонт возможен, но требует специализированного оборудования (паяльных станций, измерительной аппаратуры), редко присутствующего в секциях	специализированный ремонт от производителя доступен в каждом регионе или множестве центров. Или оборудование возможно починить самостоятельно в большинстве секций технического творчества
Применимость при подготовке к инженерным соревнованиям	оборудование плохо подходит для обучения компетенциям, важным при подготовке к инженерным соревнованиям (даже если оно хорошо подходит для обучения	оборудование подходит для обучения детей одного возраста различным техническим компетенциям (важным при подготовке к инженерным соревнованиям), близко связанным друг с другом. Например,	оборудование подходит для обучения детей разных возрастов множеству различных технических компетенций, требуемых в множестве инженерных конкурсах и соревнованиях. Оборудование

	другим компетенциям)	оборудование подходит для обучения только электронике, но не механике или программированию	позволяет развивать компетенции и знания, связанные со школьной программой по нескольким предметам или по одному предмету в течение нескольких лет обучения
Количество доступных соревнований	оборудование подходит для одного или нескольких специализированных соревнований/конкурсов	оборудование подходит для ограниченного количества соревнований/конкурсов, обусловленного его спецификой (например, подходит для определенного возраста).	оборудование подходит для множества соревнований с открытой платформой. В России проводится множество подходящих соревнований
Обеспеченность методическими материалами	методики на русском языке отсутствуют	присутствуют методики от производителя или от одного/нескольких преподавателей/центров. Методики закрыты (например, доступны при приобретении оборудования или покупке курса)	методики широко распространены. Существует множество доступных и открытых методик как от производителя, так и от сообщества
Распространенность среди преподавателей	оборудование не имеет документации на русском языке, ни от производителя, ни от сообщества. Или это профессиональное оборудование, не использовавшееся ранее в образовательной среде. Или оно новое и еще не получило распространения в	оборудование применяется несколькими преподавателями или центрами, имеет техническую документацию от разработчика, но не от сообщества	оборудование широко распространено в образовательной среде, многие преподаватели уже подготовили собственные методики проведения занятий на нем

	образовательной среде		
Взаимодействие с другим оборудованием	оборудование подразумевает закрытую проприетарную среду: дополнительные детали и модули не сопрягаются с оборудованием стандартными (предусмотренными разработчиком) способами. Для этого требуется изменять конструкцию / программное обеспечение оборудования	оборудование может взаимодействовать с другими модулями и деталями. Производитель предоставляет стандартные интерфейсы взаимодействия с другим оборудованием. Но процесс сопряжения плохо документирован и требует глубокого изучения, а список подходящего (протестированного) внешнего оборудования сильно ограничен. Например, Lego EV3 работает с ограниченным числом Web-камер по USB, а интерфейсы используют протоколы с измененной скоростью передачи данных	оборудование содержит стандартные, широко распространенные интерфейсы связи. Программно передача данных по этим интерфейсам происходит с использованием стандартных команд, предоставленных разработчиком. Для механического сопряжения со сторонними деталями производитель открыто описал размеры контактных площадок и узлов

*Источник: Исследование рынка технологических продуктов для кружков робототехники. Весна 2021 / А. Овсянников, Р. Соловьев, М. Тезина, О. Кускова, А. Федосеев, А. Андрюшков. — М.: Ассоциация участников технологических кружков, 2021, 102 с.*



Для хранения робототехнических наборов рекомендуются специальные боксы, так как они способствуют сохранению комплектующих частей образовательной робототехники и порядка их использования в процессе работы с набором. Например, мобильные боксы удобны для транспортировки и позволяют перемещаться с набором по разным учебным комнатам или помещениям (Рисунок 2.12).



Рисунок 2.12 – Мобильные боксы для хранения и транспортировки набора образовательной робототехники

Для постоянных учебных кабинетов информатики или ИКТ в начальной школе рекомендуются стационарные боксы в виде органайзеров для группы обучающихся (Рисунок 2.13).



Рисунок 2.13 – Стационарные боксы для хранения наборов образовательной робототехники в школе

### *2.3. Учебные среды управления образовательными роботами (среды программирования) для начальной школы*

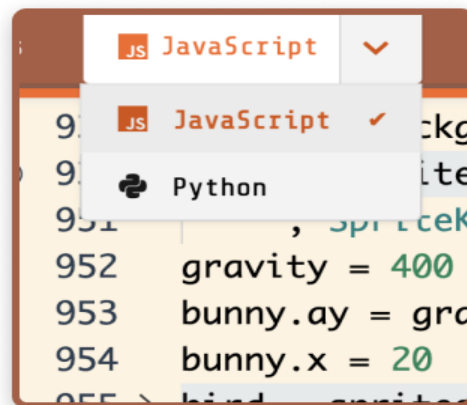
Методология использования роботов в обучении программированию в начальной школе имеет эволюционную и достаточно обоснованную эмпирическую основу. В большинстве проблемы и сложности преподавания робототехники и блочного программирования в начальной школе обусловлены высокой абстрактности в обучении программированию. Поэтому, используемый язык программирования является еще одним фактором, отличающим технологии обучения робототехнике и программированию. В образовательной робототехнике используются как текстовое кодирование, так и языки визуального программирования (VPL, visual programming languages). Кодирование текста является синонимом технического языка и строгого синтаксиса, в то время как VPL предлагают команды в визуальных блоках, которые предотвращают синтаксические ошибки и способствуют восприятию абстрактных понятий программирования. Некоторые среды программирования (например, MakeCode) включают оба варианта: дети могут использовать программные блоки в визуальной среде, наблюдая, как их команды появляются в другой окне в виде кода на JavaScript или Python, который знакомит их с основой программирования на языке высокого уровня (Рисунок 2.14).

Как известны, компьютерные программы (или алгоритмы/инструкции для робота) не материальные объекты, каждый человек видит и понимает их по-разному. Младшие школьники, которые впервые программируют для управления роботами сталкиваются с различными абстрактными понятиями. В большинстве, обучающиеся начальных классов воспринимают правильность выполняемых ими действия по внешним признакам результатов своей работы. Поэтому, не случайно базовые понятия курса «Цифровая грамотность» в начальной школе такие, как «объект», «алгоритм» и «команда» (Рисунок 3) изучается с помощью «Исполнителя», некоторого устройства, возможно и виртуального, способного выполнять систему (набор) знакомых ему команд.

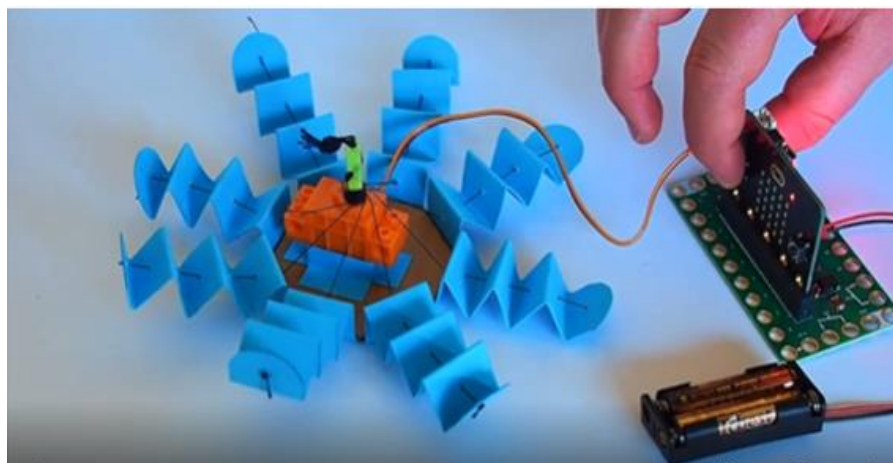
В 1967 году одним из первых, американский программист, психолог и педагог Сеймур Пейперт в своих экспериментах с младшими школьниками использовал робота-исполнителя «Черепашку», управляемого с помощью компьютера. Робот – исполнитель «Черепашка» выполняя команды алгоритма, рисовала на экране компьютера геометрические фигуры, что делало исполнение алгоритма ещё более наглядным, а процесс программирования увлекательным. Однако, как считает ученый, реальный, материальный, который можно держать на руках и видеть как он движется на поверхности механическая «Черепашка», оказалось намного эффективнее в обучении алгоритмизацию, нежели чем экранная «Черепашка» [4].



Блочное кодирование



Кодирование на JavaScript /Python



Загрузка кода на устройство/ управление роботом

Рисунок 2.14 – Принципы работы среды программирования MakeCode

Таким образом, обучение основам робототехники и блочного программирования в начальной школе построены на основе конструктивистского подхода к обучению с использованием концепции «исполнитель», примерами которой являются современные образовательные робототехники. В начальной школе предпочтение отдается конструкторским наборам роботов, управляемого с помощью специального средства программирования, которое называется «среда управления образовательными роботами».

В настоящее время существуют множество таких сред управления образовательными роботами. Однако, ни одна среда управления роботами (или среда программирования) не подходит для всех аудиторий. Вместо этого следует использовать сред от микромиров, через графических сред программирования в начальной школе до текстовых языков, обеспечивающего

эффективного обучения основам робототехники и программирования в старшей школе.

По назначению и функциональным возможностям среды управления роботами можно разделить на следующие группы:

- Среда управления роботами семейства LabVIEW: Lego WeDo Software, Среда NXT-G, Robolab, среда EV3-G и др.
- Scratch и Scratch-подобные среды: Microsoft MakeCode, Blockly, AppInventor и др.

### *Среды управления роботами семейства LabVIEW*

Программирование в LabVIEW осуществляется на визуальном языке программирования потоков данных G. Язык G моделирует процесс вычислений, ориентированный на данные, в котором явно задаётся не последовательность выполнения операторов, а связи между блоками по данным. Блок программы может предоставлять некоторые выходные данные, которые могут служить входными данными для другого блока. Блоки начинают исполняться, когда имеют данные на всех входах. Если сразу несколько блоков имеют данные на всех входах, они исполняются параллельно. Такой подход довольно сильно отличается от подхода, принятого в императивном программировании, но, тем не менее, он широко распространён среди инженеров и учёных. Среда поддерживает огромное количество аппаратных платформ и предоставляет десятки библиотек программных компонентов. В частности, имеются модули, обеспечивающие поддержку в LabVIEW робототехнических конструкторов Lego Mindstorms NXT и Lego Mindstorms EV3. Система LabVIEW достаточно сложна для изучения в школе, поэтому используются различные ее адаптации такие как, NXT-G, Robolab и другие.

*Среда NXT-G.* Среда NXT-G специально создавалась для начинающих, поэтому довольно проста и удобна в работе. По мнению некоторых пользователей, она даже слишком эргономична, поскольку не даёт произвольно размещать блоки на диаграмме, автоматически (и не всегда удачно) прокладывает соединительные линии между блоками и т. д. . Имеются блоки чтения и записи значения в переменную, блок, считывающий значение константы, блоки, считывающие показания с сенсоров, и блоки элементарных арифметических операций, не предоставляющих, впрочем, даже операции извлечения арифметического квадратного корня. Большинство свойств элементов не отображается на диаграмме, а доступна только через редактор свойств, что делает невозможным отображение всей программы на проекторе.

### *Lego WeDo Software*

Учебные среды управления образовательными роботами предназначенные для детей дошкольного возраста и начальной школы имеют свои особенности. В частности, это касаются интерфейса среды и привлекательность среды. Важными для такой среды критериями являются ее «интуитивность», дружелюбность интерфейса к маленьким пользователям и

наличие привлекательных игровых элементов, способствующих вызвать интерес обучающихся к обучению. Lego WeDo — довольно популярное в мире образовательное решение от Lego, ориентированное на дошкольников и учащихся первого-четвертого классов. С помощью Lego WeDo можно собирать простейшие подвижные конструкции, которые, при желании, могут оборудоваться датчиками расстояния и наклона, входящими в комплект. Программирование осуществляется в визуальной среде, очень похожей на NXT-G и EV3-G (среда также основана на LabVIEW), однако с некоторыми упрощениями. Среда содержит множество примеров в виде картинок с моделями, собранными из конструктора, дружелюбна к пользователю, русифицирована. Однако язык G может оказаться сложным для детей дошкольного возраста. Кроме того, существуют отрицательные результаты юзабилити-экспериментов, которые, хоть и проведены с NXT-G, касаются WeDo Software (к примеру, назначение не всех пиктограмм элементов в WeDo интуитивно понятно, зачастую они отличаются друг от друга лишь мелкими значками). Ко всему прочему, Lego WeDo Software не является бесплатной средой, цена сравнима со стоимостью самого конструктора.

Следующая среда, *Robolab*, разработанная на основе LabVIEW с учетом особенностей преподавания робототехники и программирования в школе. Среда управления Robolab является средой с двумя уровнями сложности для обучения программированию. Каждый из этих уровней разбивается среды еще на несколько подуровней, на каждом из которых пользователь ознакомится с новой функциональностью программирования роботов. На самом простом, пилотном уровне доступны только некоторые возможности визуального языка, и программа строится заполнением пустых мест в шаблоне посредством выбора блоков из всплывающего меню. Это позволяет создавать только самые простые программы, имеющие стандартную структуру: команды управления моторами, за которыми следует блок, ожидающий наступления какого-либо события. Идея такого разделения в том, чтобы дать возможность детям в начальной школе или даже детском саду пользоваться программой — в столь раннем возрасте они вполне могут не уметь читать. На втором уровне сложности пользователи могут рисовать диаграммы, размещая произвольным образом блоки из палитры и соединяя их линиями, определяющими поток управления. Разбиение на уровни и подуровни организовано так, чтобы дети могли осваивать среду программирования практически без помощи учителя, руководствуясь лишь интуицией. В отличие от NXT-G, Robolab позволяет описывать произвольные математические выражения в текстовом виде. Развитие Robolab идёт в основном путём добавления новых блоков, сама среда давно не изменялась. В частности, в среде отсутствует поддержка Lego EV3. Несмотря на указанные недостатки, Robolab на данный момент является одной из наиболее широко используемых в школах сред программирования роботов.

*Среда EV3-G* — программное обеспечение, поставляемое в комплекте с конструктором Lego Mindstorms EV3. EV3-G также создана на основе LabVIEW

и позволяет программировать контроллеры NXT и EV3 на языке G. Среда во многом аналогична NXT-G: имеет современный пользовательский интерфейс, набор примеров исполнения диаграмм на роботе.

### *Scratch и Scratch-подобные среды*

*Scratch (Скретч)* – среда программирования, которая дает возможность учащимся младшего и среднего школьного возраста создавать игры, фильмы, анимированные истории и многое другое.

Одним из достоинств данной среды является то, что она является свободно распространяемым программным продуктом, таким образом, любая образовательная организация может скачать программу из интернета по ссылке <https://scratch.mit.edu> и приступить к непосредственному изучению и работе в новой среде программирования. Среда программирования Scratch не требует установки.

Проект по созданию Scratch был инициирован в 2003 году при поддержке компаний Science Foundation, Intel Foundation, Microsoft, MacArthur Foundation, LEGO Foundation, Code-to-Learn Foundation, Google, Dell, Fastly, Inversoft и MIT Media Lab. Сама среда Scratch разработана и поддерживается группой разработчиков под руководством Митчела Резника (Mitchel Resnick) из Массачусетского института технологии (MIT Media Lab). Scratch является свободно распространяемым продуктом, что немаловажно для общеобразовательных школ. Его разработка ведется на языке Squeak, одном из наследников Smalltalk. Разработчик языка Smalltalk, Алан Кей, американский ученый в области теории вычислительных систем, один из пионеров в области объектно-ориентированного программирования и графического интерфейса, очень заинтересованно относится к развитию мышления и креативности детей. Алан Кей считает, что такое развитие должно начинаться как можно раньше: «...современные мультимедийные технологии, которые мощно формируют наши способы мышления, должны быть доступны как можно раньше (the media that powerfully shape our ways of thinking must be made accessible as early in life as possible)».

### **3. Методическая система преподавания курса "Робототехника"**

#### **1. Возрастные особенности учеников 10-11 классов**

- 1.1. Знакомство
- 1.2. Основные навыки
- 1.3. Ведущая деятельность этих классов
- 1.4. Влияние внешних факторов
- 1.5. Образность мышления

#### **2. Состав стандартного урока(план, часы, проекты)**

2.1. Методика разделенная на несколько частей - 1) новая тема, промежуточный тест(с несколькими вариантами : 1 ответ, 2-3 ответа, открытый тест и т.д), соревновательное/творческое занятие, 2) новая тема, проходной тест(с несколькими вариантами : 1 ответ, 2-3 ответа, открытый тест и т.д), соревновательное/творческое занятие, 3) проект.

2.2. Время проведения 1 урока - минимум 1-1,5 часа(астр.ч)

2.3. Процентное соотношение:

5 класс - 70 конструирование/ 30 программирование

6 класс - 50 конструирование/50 программирование

7 класс - 30 конструирование/70 программирование

8 класс - спортивная/самостоятельная(творческая) робототехника

9 класс - спортивная+проектная робототехника

2.4. План урока(с 9 класса вариант перевернутого урока ):

Сдача домашнего задания(учитывать продолжительность урока и не тратить слишком много времени)

Повторение

Логические задачи, задачи на внимательность

Теория новой темы по конструированию(на дом. можно отправлять видеоматериалы либо ссылки)

Сборка

Теория новой темы по программированию(доп.изучение на дом в пред.уроке)

Задачи на закрепление

Доп задачи, либо свободное время на улучшение программы/сборки

Домашнее задание(теория след.урока)

#### **3. Формат работы с учениками 5-9 классов**

3.1. Работа в команде, социализация

3.2. Дисциплина и порядок

3.3. Тех.безопасность во время обучения

3.4. Закрепление материалов при помощи дз и тетрадок для конспектов

#### **4. Методика оценивания учеников 5-9 классов**

4.1. В виде соревнований

4.2. Тайминг и ответы на вопросы

4.3. Оценивание домашней работы, закрепление знаний на уроке(на вопрос зачем нужен преподаватель)

4.4. Взаимная оценка участников команд



- 4.5. Система мотивации
- 4.6. Самостоятельное оценивание
- 5. На что стоит обращать внимание при обучении учеников 5-9 классов**
  - 5.1. Завлечение к уроку
  - 5.2. Повторение прошлых тем для закрепления информации
  - 5.3. Дополнительные задачи для развития: логического мышления, памяти, внимательности и мотивации.
  - 5.4. Темы занятий с плавным переходом от легкого - к сложному. Профориентация, использование школьных знаний, и постоянная практика в виде соревнований
  - 5.5. Доп.материалы для урока (видео, гифки, интересные факты и т.д)
    - Практика
    - Программирование
    - Работа с ошибками
    - Самостоятельная работа
- 6. Подготовка рабочего пространства/подготовка к уроку**
  - 6.0. Объяснить детям и родителям важность самостоятельной подготовки/практики(привести примеры)
  - 6.1. Подготовка к уроку с поиском дополнительной информации, которая отсутствует в методичке
  - 6.1. Порядок и сортировка набора при использовании конструкторов
  - 6.2. Проверка рабочего состояния датчиков, хабов и батареек
  - 6.3. В плане урока добавлять информацию о дополнительных необходимых материалов для ведения занятий(поля, канц.товары, кегли, ватманы, приложения(tinkercad, LDD, Studio 2.0 и т.д) либо спорт.программы, 3Д принтеры(для спортивной/творческой роб-ки), 3д ручки, PowerPoint, Prezi, Tilda и т.д)
- 7. Информация про спортивную робототехнику**
  - 7.1. Перспектива робототехники
  - 7.2. Подготовка к соревнованиям/проведение внутренних соревнований
  - 7.3. Регламенты профессиональных спортивных турниров по робототехнике
  - 7.4. Информация про турниры
  - 7.5. Информация про творческо-проектные категории спортивной робототехники
  - 7.6. Распределение ролей внутри команды из нескольких человек(программисты, краудфандинг(внеш.вид), инженеры, PR-деятельность(внеш.вид))

Основные навыки этого возраста: порядок, своё место в мире, самостоятельность и система. На первое место выдвигаются мотивы, связанные с жизненными планами учащихся, их намерениями в будущем, мировоззрением и самоопределением.

В среднем школьном возрасте (от 10–11 до 14–15 лет) определяющую роль играет общение со сверстниками. Ведущими видами деятельности являются учебная, общественно-организационная, спортивная, творческая, трудовая.

В этот период ребенок приобретает значительный социальный опыт, начинает постигать себя в качестве личности в системе трудовых, моральных, эстетических общественных отношений. У него возникает намеренное стремление принимать участие в общественно значимой работе, становиться общественно полезным. Эта социальная активность подростка обусловлена большей восприимчивостью к усвоению норм, ценностей и способов поведения, существующих во взрослых отношениях.

Также в этом возрасте появляется потребность в признании взрослых и потребность в социальном признании, которая обнаруживается в речевой форме «я тоже имею право, я могу, я должен». Как субъект учебной деятельности подросток склонен утверждать позицию своей исключительности, что может усиливать познавательную мотивацию. Социальная активность школьника среднего возраста в основном обращается на усвоение норм, ценностей и способов поведения. Поэтому важность заключается в реализации всех принципов обучения, инициирующих умственную деятельность подростка: его проблематизацию, диалогизацию, индивидуализацию и др. Содержание учебной деятельности должно вводиться в современные условия общественно-экономических и социально-бытовых отношений.

Ведущей деятельностью становится будущее. Это новая социальная позиция изменяет для них и значимость учения, его задач и содержания. Теперь ученики оценивают учебный процесс с точки зрения того, что он дает для их будущего. Они начинают иначе, чем раньше, смотреть на школу.

Школьники, только начинающие учиться в средней школе, в связи с низким уровнем сформированности абстрактно-логического мышления уже с первых дней начинают испытывать значительные трудности в обучении, и, кроме того, у них может в связи с этим сформироваться стойкое отрицательное отношение к учению и интеллектуальной деятельности в целом.

Целенаправленное формирование абстрактно-логических форм мышления должно явиться основной задачей развития учащихся средней школы, начиная с младшего подросткового возраста.

Наблюдается следующая динамика мотивов общения со сверстниками: в 12—13 лет подростку важно занять определенное место в коллективе

сверстников. Содержание общения младших подростков сосредоточивается вокруг процессов учения и поведения — лидер среди сверстников тот, кто лучше учится и правильно себя ведет, положительный образ является ведущим.

Примерно с 12 лет у подростков начинает интенсивно развиваться личностная и межличностная рефлексия, в результате которой они склонны видеть причины своих неудач, конфликтов или успехов в общении в особенностях собственной личности. Появляется способность брать ответственность на себя за успешность своего общения с окружающими.

### **Особенность возраста в 11-12 лет и методика по робототехнике:**

- База знаний алгоритмов продолжает пополняться, и это будет идти ещё долгое время, хотя и начинает терять приоритет в развитии ребёнка.
- Основные вопросы из прошлого дня продолжают решаться, но теперь они приобретают ещё более важное значение. Теперь вопросы группы и авторитета в группе стоят особенно остро, равно так и вопрос собственного призвания и стремлений в будущем.
- Вопрос самостоятельности лучше всего рассматривать в ключе ведущей деятельности, но в основе этого навыка лежит стремление сделать всё “по-своему”, наработать собственный опыт на своих ошибках, даже если верный путь лежит прямо перед глазами.
- Теперь возникают вопросы пути к правильному ответу или тому или иному суждению.
- Помимо этого, появляется стремление всё чаще пользоваться накопленными алгоритмами при решении задач.
- Теперь, когда база знаний алгоритмов в голове сформирована, её нужно привести в порядок, поэтому характерным для учебного процесса становится систематизация знаний по различным предметам, установление межпредметных связей.
- Все это создает почву для овладения общими законами природы и общественной жизни, что приводит к формированию научного мировоззрения.
- Ученик в своей учебной работе старается пользоваться различными мыслительными операциями, рассуждает логически, запоминать осмысленно.
- В то же время познавательная деятельность учеников приобретает свои особенности.
- Если раньше это было стремление разобраться, что собой представляет то или иное явление, то теперь это стремление разобраться в разных точках зрения на этот вопрос, составить мнение, установить истину.

### **Возраст детей: 13+**

Вторая ведущая деятельность – *выстраивание отношений*. Отношения

между сверстниками, со взрослыми и с теми, кто младше. масштабно начинается образование коллективов. Теперь к ученикам следует относиться как к полноценным членам коллектива, а не “подчинённым”.

Фигура преподавателя теряет неприкосновенность, теперь, чтобы дети были заинтересованы и слушались, преподаватель должен завоевать авторитет в их глазах. Самый простой способ – доказать, что преподаватель может что-то сделать руками, что-то практическое. Тем самым добывается авторитет, можно передать настрой классу и найти среди детей “личных учеников”.

В 14—15 лет доминирует стремление к автономии в коллективе и поиск признания ценности собственной личности в глазах сверстников. Содержание общения фокусируется на вопросах личностного общения, индивидуальности — наиболее привлекательной становится личность “интересная”, “сильная”, “особенная”. У многих подростков оказывается фрустрированной потребность “быть значимым” в их среде.

Поиски общения со взрослыми у них даже выше, чем в другие возрастные периоды. Стремление иметь взрослого друга объясняется тем, что решить вставшие проблемы самосознания и самоопределения самому бывает очень трудно.

Цель. Не стоит путать её с целью из прошлой ведущей деятельности, хотя они и связаны между собой. Новое понимание “Цели” - это “Пригодится ли мне это потом? Нужно ли мне это?”. Теперь в обучении всё чаще ученик начинает руководствоваться сознательно поставленной целью, появляется стремление углубить знания в определенной области, возникает стремление к самообразованию. В этом возрасте встречаются два типа учащихся: для одних характерно наличие равномерно распределенных интересов, другие отличаются ярко выраженным интересом к одной науке.

Если раньше их учебные интересы определяли желаемый выбор профессии, то теперь наблюдается обратное: выбор профессии способствует формированию учебных интересов, изменению отношения к учебной деятельности. Если ребёнок понимает, что он яро хочет стать программистом в будущем, то акцент смещается на технические предметы, в то время как гуманитарные начинают страдать. Это нужно учитывать и в работе с робототехникой, потому что это мультидисциплинарное обучение. Стоит помнить, что ребёнок берёт только то, что реально полезно ему, а это значит, что для повышения мотивации работать нужно объяснить ему, что это будет полезно для него конкретно.

В этом возрасте, особенно у тех, кто ходит давно, начинает пропадать интерес – и их нужно замотивировать обратно.

Для начала, нужно понять причину этого, чему может способствовать разговор по душам, выяснение проблем (у ребёнка, в семье?). Мелкие одноразовые проблемы можно оставить на поблажку на занятии, крупные надо решать. Нужно постараться стать детям старшим товарищем ментором и наставником. Также важно соблюдать правила вместе с ними.

Выбор решения. Ученикам становится скучно, если нет задач для ума. Они любят исследовать и экспериментировать, творить и создавать новое, оригинальное. Их интересуют не только вопросы теории, но и сам ход анализа, способы доказательства.

Им нравится, когда преподаватель заставляет выбирать решение между разными точками зрения, требует обоснования тех или иных утверждений; они с готовностью, даже с радостью вступают в спор и упорно защищают свою позицию. Как и в математике, у задач может быть несколько путей решения, задача преподавателя здесь дать ребёнку возможность осознать, что это решение содержит лишние шаги или неточности, или дать ему возможность доказать обратное.

Дайте им возможность побыть самостоятельными.

Однако, уделяется больше внимания алгоритмам и типовым решениям – краски закончились, работаем серьёзно. Сперва решаем типовые задачи, оттачиваем их. Потом учимся принципу декомпозиции - дробим сложную задачу на подзадачи, которые решаем. Потом, освоив типовые решения, учимся редукции – разбиваем задачу на типовые подзадачи, которые уже умеем решать. Стремление к систематизации достигается за счёт типовых решений типовых задач. Ребёнок пошагово расширяет свой инструментарий и получает готовые “инструменты” в виде типовых решений.

Самостоятельность они могут проявить в различных задачах для ума, либо (особенно это касается очень способных учеников), в виде дополнительных задач, усложненного типа, в которых им бросается некоторый “вызов”, и они должны с ним справиться.

В мышлении подростка открывается способность абстрагировать понятие от действительности, формулировать и перебирать альтернативные гипотезы, делать предметом анализа собственную мысль. По этой причине важно в методике давать большое количество самостоятельных задач. Выделять на них как можно больше времени и внимания.

Мощным фактором саморазвития в старшем подростковом возрасте становится появившийся интерес к вопросу “Каким я могу стать в будущем?” Именно с таких размышлений начинается перестройка мотивационной сферы, обусловленной ориентацией на будущее. Поэтому у большинства учащихся не будут проблемы с концентрацией внимания, т.к. выбирая данный предмет они уже заинтересованы в изучении.

## Состав стандартного урока

Время проведения занятий по робототехнике должно составлять не менее 2 академических часов. По классам время проведения занятий можно разделить:

5-6 класс - 3 ак.ч

7 класс - 2.5 ак.ч

8 класс - 2 ак.ч

9 класс - 1.5 ак.ч

План занятий можно разделить по времени и виду деятельности. Для каждого класса и урока - план составляется индивидуально, учитывая тему урока и выделенное время. Пример:

### 1 урок. Знакомство с набором Lego Mindstorms

Знакомство	15 минут
Правила	10 минут
Теория	30 минут
Упражнения для глаз	5 минут
Сборка	45 минут
Перерыв	15 минут
Подключение	10 минут
Программирование	30 минут
Свободное время	20 минут

Рисунок 1. План урока

Занятия по робототехнике у учеников средних классов строятся на четырех составляющих:

1. Установление связей (формируются умения определять причинно-следственные связи, постановка целей и задач);
2. Конструирование (изучаются практические навыки работы с механизмами);
3. Развитие (поощрение творческой активности детей, желание экспериментировать, предлагать собственные решения вопросов);
4. Рефлексия (проведение исследований, подведение итогов, обобщение выводов).

Разработка уроков робототехники определяется материально-технической базой каждой школы. При этом исследователи выделяют две методики преподавания: спортивное и STEM.

Первое подразумевает подготовку моделей к соревнованиям и не является универсальным для проведения общих уроков.

Второе направление более актуально, так как развивает навыки ребенка в комплексе, основывается на проектной деятельности.

Исходя из вышеперечисленных описаний занятия по робототехнике для средних классов можно поделить на процентные соотношения:

Для классов: 5-6	Соотношение между конструированием и программированием 50 на 50. Так как здесь идет установление связей и общая теория для изучения и развития практических навыков как в конструирований так и в программирований.
Для классов: 7	30% времени на изучение и знакомство с новыми темами а оставшиеся 70% на развитие, то есть поощрение творческой активности детей, эксперименты и самостоятельные задачи
Для классов: 8-9	Знакомство с Ардуино. По этой причине 70% времени выделяется на конструирование и программирование, оставшиеся 30% времени на проведение исследований, спортивной робототехнике и подведению итогов.

Начиная с 9 класса можно предложить план «перевернутого урока» для учеников. То есть основной план занятий провести наоборот.

Так в «перевернутом уроке» изучение теоретического материала и выполнение домашнего задания по нему меняются местами. Ученики просматривают лекции перед уроком в качестве домашнего задания. Время на уроке посвящено обсуждениям, интерактивным упражнениям и самостоятельной работе, которая раньше выполнялась бы дома – все под руководством преподавателя, который присутствует и готов ответить на любые вопросы, которые могут возникнуть. Этот метод настраивает ученика на успех.

План перевернутого урока отлично подходит ученикам средних и старших классов. Используя данную методику, можно развить их потенциал, т.к. в этом возрасте для них важно приходить к решению задач самостоятельно, и они открыты к новым знаниям.

## Формат работы с учениками 5-9 классов

Давайте рассмотрим роли в команде, а также этапы реализации Идеи в таких группах.

Рассмотрим выполнение наших Идеи по ролям и стадиям.

### **Творец**

Он же “Генератор идей”, бесконечный источник мыслей, задумок и рационализаторских предложений. Однако, у идеального Творца есть существенный недостаток - низменная методичная и рутинная реализация собственных идей повергает его в уныние, поэтому он почти не способен довести дело до конца.

### **Критик**

Этот человек также необходим для группы, потому что его задача - первый уровень “фильтрации” Идеи™ на предмет Смысла™. Под этим подразумевается безжалостная атака на Идею™ кучей вопросов «Зачем?», «Почему?», «А надо ли?» и т.д., пока она или не выстоит под натиском, либо не окажется несостоятельной. Недостаток этой роли - идеальные Критики не очень любят возиться с генерацией и выполнением. Могут, но не очень это любят, особенно с реализацией. Идеальная позиция для них - роль советника-эксперта.

### **Реалист**

Он же «Специалист по реальности». Он также необходим, но уже после проверки Идеи™ на предмет Смысла™. Его задача - проверить Идею™ на предмет реалистичности выполнения/доступных ресурсов/технологий и т.д. Его главная цель - проверить, возможно ли вообще реализация всей задумки. Идеальный Специалист делит недостатки с Идеальным Критиком, но если Критик больше тяготеет к Творцу, то Специалист к Исполнителю. Он может сам что-то поделывать, но нечасто.

### **Исполнитель-доводчик**

Кратко «Исполнитель». Тот самый человек, который выполняет 90% работы и воплощает созданную, раскритикованную, проверенную Идею в жизнь. Однако, идеальный Исполнитель не жалуется, наоборот, он любит что-то выполнять и реализовывать.

Самый главный недостаток идеального Исполнителя противоположен недостатку Творца - идеальный Исполнитель не любит думать над новыми вещами, он любит выполнять чётко поставленные задачи, и, если задачи закончились, то он терпеливо будет ждать новых указаний. Описанные портреты - это идеальные примеры таких ролей. В реальном мире, конечно, все люди сочетают в себе несколько ролей (обычно 2, редко 3, и совсем редко все 4), но это не мешает среди этих ролей иметь некую преобладающую, которая определяет ведущую роль в группе.



## Как это задействовать в работе с группами?

Стараться формировать сбалансированные группы из детей с разными ролями. Это сложнее с детьми младшего возраста, потому что чувство критики появляется у детей годам к 10-11, до этого развито очень слабо, но старшие дети вполне могут работать полноценным составом.

Следить за детьми - даже в паре, если вы видите, что одна группа не замолкает и «совершенствует» конструкцию и/или задачу, но за занятие делает примерно ничего, то это пара двух Творцов. Если другая пара как роботы выполняют задачу за задачей, но сидят и ждут новых, то это пара двух Исполнителей. Верным решение будет поменять по человеку в каждой паре, чтобы каждый Творец получил «руки» для своих идей, а Исполнитель - источник новых задач. Не бойтесь пересаживать детей.

В этом возрасте группа в состоянии выработать правила, и сама их поддерживать. Чтобы группа следовала им, это должно быть, как и раньше, именно выработка правил, совместно. Например, на первом занятии каждый ребёнок предлагает одно правило на вопрос «Что нам нужно, чтобы хорошо работать на занятиях?». Ответы записываются. Начинается мозговой штурм, во время которого отсекаются ненужные правила, остаются только самые нужные. ВАЖНО! Преподаватель – часть коллектива, поэтому правила должен соблюдать и он. Иначе – потеря авторитета в группе. Появляются бунтари. Появляются, чаще всего, от нехватки внимания. Это следствие цели «Быть лучше, чем остальные».

Решение – дать ему небольшую власть, что угодно, что возвысит его над простыми учениками. Сделать его держателем правил, ответственным за доску/наборы и т.д. таким образом превращаем пагубное воздействие в полезное.

В любом возрасте важно учитывать правила техники безопасности во время обучения по робототехнике. Ниже указан пример, дополнительно для класса вы также можете добавить свои правила исходя от техники, которой вы пользуетесь:

### **Правила поведения обучающихся в кабинете робототехники**

Находясь в кабинете робототехники, обучающиеся обязаны:

1. Соблюдать дисциплину и порядок, правила техники безопасности и чистоту;
2. Занимать рабочие места согласно указаниям преподавателя и не менять их самовольно;
3. Заниматься только тем видом деятельности, которую определил преподаватель;
4. Немедленно сообщать преподавателю о любых замеченных неисправностях оборудования или неверной работе программного обеспечения;

5. Немедленно сообщать преподавателю о любом случае травматизма в кабинете, особенно от электрического тока.

Требования безопасности во время работы:

1. С техникой обращаться бережно: не стучать по мониторам, не стучать мышкой о стол, не стучать по клавишам клавиатуры;
2. При возникновении неполадок: появлении изменений в функционировании аппаратуры, самопроизвольного её отключения необходимо немедленно прекратить работу и сообщить об этом преподавателю;
3. Не пытаться исправить неполадки в оборудовании самостоятельно;
4. Выполнять за компьютером только те действия, которые говорит преподаватель;
5. Контролировать расстояние до экрана и правильную осанку;
6. Не допускать работы на максимальной яркости экрана дисплея;
7. В случае возникновения нештатных ситуаций сохранять спокойствие и чётко следовать указаниям преподавателя.

Запрещается:

1. Эксплуатировать неисправную технику;
2. При включённом напряжении сети отключать, подключать кабели, соединяющие различные устройства компьютера;
3. Касаться экрана дисплея, тыльной стороны дисплея, разъёмов, соединительных кабелей, токоведущих частей аппаратуры;
4. Касаться автоматов защиты, пускателей, устройств сигнализации;
5. Во время работы касаться труб, батарей;
6. Самостоятельно устранять неисправность работы клавиатуры;
7. Нажимать на клавиши с усилием или допускать резкие удары;
8. Требования безопасности по окончании работы.

По окончании работы дожидаться пока преподаватель подойдёт и проверит состояние оборудования, сдать работу, если она выполнялась

Медленно встать, собрать свои вещи и тихо выйти из класса, чтобы не мешать другим обучающимся

Ответственность за нарушение правил техники безопасности:

1. При нарушении техники безопасности обучающемуся будет объявлен выговор, взыскание вплоть до отстранения от работы за оборудованием.
2. При регулярных нарушениях техники безопасности обучающийся будет отстранён от занятий робототехники вплоть до исключения из учебного заведения (или других санкций, предусмотренных учебным заведением).

## Методика оценивания учеников 5-9 классов

Практико-ориентированный подход в освоении новых знаний. Первый и главный принцип – всё осваивать и проверять на практике. Наши занятия и методы не могут быть сугубо теоретическими, поэтому выполнение любой задачи проверяется только на практической демонстрации в «полевых условиях». Если задача решена в коде, без его проверки, то она не решена. Если задача решена, проверена несколько раз, и работает стабильно, то только тогда она решена. Этому же принципа мы придерживаемся везде, и никогда не верим на слово, всё нужно доказать на практике.

Система «Ментор-ученик». На любой ступени каждый кого-то учит и у кого-то обучается. Преподаватель не находится в позиции Мастера-эксперта, который знает всё на свете и разбирается во всём подряд. Важно не потерять способность принимать знания от детей, с которыми вы работаете, зачастую они могут случайно подсказать решение вашей задачи. В идеальном варианте, преподаватель налаживает прочную связь с несколькими учениками (старшими), и создаёт с ними систему "Ментор-ученик", когда преподаватель выходит за рамки робототехники и образования, а консультирует уже по любым жизненным вопросам. В вопросах организации контрольно-оценочной деятельности на уроке накоплен большой опыт. Но актуальность данного вопроса не снижается. Необходимо совершенствовать практику оценивания знаний обучающихся с целью повышения качества и эффективности урока. Оценить – значит определить уровень, качество выполнения учащимися задач, поставленных перед ними в процессе обучения. Отметка – это установленное государственными стандартами обозначение степени знаний обучающегося. Контроль – процедура проверки и оценки учебных достижений учащихся, направленная на установление степени соответствия реально достигнутых результатов учебной деятельности каждым учащимся планируемым результатам обучения в предметно-деятельностной форме, определенных образовательными стандартами и учебными программами. Вместо отметки в данном направлении мы в основном используем баллы - как мотивацию для продолжения обучения, ну и как способ сохранения дисциплины.

### КРИТЕРИИ ОЦЕНКИ ИТОГОВОГО ПРОЕКТА

1. Проект (Максимум баллов: 50)	1. Оригинальность и качество решения – Проект уникален и продемонстрировал творческое мышление участников. Проект хорошо продуман и имеет реалистичное решение / дизайн / концепцию.	25
	2. Исследование и доклад – Команда продемонстрировала высокую степень изученности проекта, сумела четко и ясно сформулировать результаты исследования.	15
	3. Зрелищность – Проект имел восторженные отзывы, смог заинтересовать на его дальнейшее изучение.	10
2. Программирование (Максимум баллов: 45)	1. Автоматизация – Проект работает автономно, либо с небольшим вмешательством человека. Роботы принимают решения на основе данных, полученных с датчиков.	15
	2. Логика – Программа написана грамотно, выполнение происходит логично на основе ввода данных с датчиков.	15
	3. Сложность – Алгоритм программы содержит нетривиальные (непримитивные, сложные) формы линейной, условной и циклической структуры, а также структуры декомпозиции.	15
3. Инженерное решение (Максимум баллов: 45)	1. Техническое понимание – Команда продемонстрировала свою компетентность, сумела четко и ясно объяснить, как их проект работает.	15
	2. Инженерные концепции – В конструкции проекта использовались хорошие инженерные концепции.	10
	3. Эффективность механики – Общий дизайн проекта демонстрирует эффективность использования механических элементов (т.е. правильное использование зубчатых передач, средства для снижения трения; экономное использование деталей; простота ремонта/изменений, и т.д.)	10
	4. Стабильность конструкции – Конструкция устойчива и проект может быть неоднократно запущен без дополнительного ремонта (или исправлений).	5
	5. Эстетичность – Проект имеет хороший внешний вид. Команда сделала все возможное, чтобы проект выглядел профессионально.	5
4. Презентация (Максимум баллов: 40)	1. Успешная демонстрация – Проект работает так, как и предполагалось, с высокой степенью воспроизводимости.	15
	2. Навыки общения и аргументации – Участники смогли рассказать, о чем их проект, и объяснить, как он работает и ПОЧЕМУ они решили его сделать	10
	3. Быстрота мышления – Участники команды с легкостью ответили на вопросы, касающиеся их проекта	5
	4. Плакаты и оформление – Материалы, используемые для презентации, понятны, лаконичны и упорядочены.	5
	5. Видеоролик о проекте	5
5. Командная работа (Максимум баллов: 20)	1. Уровень понимания проекта – Участники продемонстрировали, что все члены команды имеют одинаковый уровень знаний о проекте.	10
	2. Сплоченность коллектива – Команда продемонстрировала, что все участники коллектива сыграли важную роль в создании и презентации проекта.	5
	3. Командный дух – Все члены команды проявили энтузиазм и заинтересованность в презентации проекта другим.	5

На основании баллов, заработанных командой, выстраивается общий рейтинг.

**Таблица 1. Пример оценивания проекта**

Соответственно, когда мы формируем у ученика способность выполнять какое-либо действие, то сначала он выполняет его развернуто, подробно фиксируя в своем сознании каждый шаг выполняемого действия, значит способность выполнять какое-либо действие формируется у ребенка как умение. Это умение совершенствуется по мере тренировки и частоте его выполнения, соответственно процесс выполнения данного действия свертывается и оно начинает выполняться автоматизировано – следовательно, умение ученика переходит в навык.

Контроль – выявление и сравнение результата учебной деятельности ученика с требованиями, которые задаются к этому результату программой. Контроль включает в себя: проверку знаний; процесс оценивания и выставление отметки, т.е. результат оценивания.

Основной целью контроля и оценки знаний являются определенные качества усвоения учащимися материала, уровня овладения знаниями, умениями и навыками, которые предусматривает учебная программа.

Контроль, как составная часть процесса обучения, носит образовательное, развивающее и воспитательное значение.

Определение меры ответственности каждого ученика за результаты своей учебы является основной задачей контроля.

Контроль знаний должен быть систематическим, всесторонним и разнообразным по формам, а также должен быть основан на единстве требований учителей.

Контроль знаний, умений и навыков в начальной школе выполняет следующие функции:

- 1) воспитывающая (воспитание у школьника чувства ответственности за результат учебы)
- 2) обучающая функция (повышение качества знаний, их систематизация)
- 3) развивающая (стимулирование для познавательной активности школьников)
- 4) контролирующая и диагностическая (выявление и диагностика результатов обучения у школьников)
- 5) прогностическая (управление процессом усвоения знаний и умений и его коррекция).

Методы контроля – это способы, благодаря которым определяются результативность учебно-познавательной деятельности учителя и учащихся.

Существуют следующие методы контроля:

По месту контроля в процессе обучения:

1. Текущий – может быть как устным, так и письменным: индивидуальный опрос, фронтальный опрос, дискуссия, проектная исследовательская деятельность, тесты, практическая работа.

2. Тематический – может осуществляться при проведении тестирования, практической работы.
3. Промежуточный – проводится в конце каждого класса, может реализоваться через контрольную работу или итоговое тестирование, проводится письменно. Для старших классов можно проводить в виде экзамена по билетам.
4. Итоговый – в конце выпуска, также реализуется через контрольную работу, итоговое тестирование, который проводится письменно.

По способу организации:

1. Устный (устный опрос, дискуссия, пересказ по заданию учителя, рассказ дополнительного материала (статьи, примет, загадок, стихов), рассказ по иллюстрации, по опорной схеме;
2. Письменный (тесты открытой и закрытой формы, письменный опрос, карточки различных видов, заполнение таблиц, дополнение схем)

По количеству участников:

1. Фронтальный
2. Групповой
3. Индивидуальный

По преобладающему виду деятельности:

1. Вербальный (словесная): работа с текстом устная и письменная, с таблицами, схемами, опрос по названиям деталей)
2. Практический (выполнение рисунка, воспроизведение опыта, самостоятельная сборка, практическая проверка на знание передач)

По субъекту проверки:

1. Проверка в парах (взаимопроверка)/
2. Проверка дежурными
3. Проверка в группах
4. Самопроверка
5. Проверка педагогом

Средства проверки:

1. Иллюстрации,
2. Схемы, таблицы,
3. Макеты, модели,
4. Карточки, тесты,
5. Авторские тетради для проверки знаний.
6. Оценка – это определение в оценочных ситуациях степени усвоения учебного материала, его качества.
7. Отметка – оценка, выраженная баллом.

Таблица. Пример оценивания учеников

Показатели (оцениваемые параметры)	Критерии	Степень выраженности оцениваемого качества	Возможное количество баллов	Методы диагностики
<b>1. Предметные результаты</b>				
1.1. Овладение знаниями (по основным разделам учебно-тематического плана)	Соответствие знаний программным требованиям	- <i>минимальный уровень</i> (объем усвоенных знаний составляет менее чем ½ объема знаний предусмотренных программой), - <i>средний уровень</i> (объем усвоенных знаний составляет более ½), - <i>максимальный уровень</i> (освоил весь объем знаний, предусмотренных программой)	1 2 3	Тесты Соревнования
1.2. Понимание смысла основных понятий	Осмысленность и правильность использования специальной терминологии	- <i>минимальный уровень</i> (не употребляет специальные термины), - <i>средний уровень</i> (использует специальную терминологию), - <i>максимальный уровень</i> (осознанно употребляет терминологию)	1 2 3	контрольно-диагностические работы
1.3. Уровень развития инженерного мышления	сформированность способности мыслить в предмете; умение анализировать, сравнивать, классифицировать явления	— <i>низкий уровень</i> — <i>средний уровень</i> — <i>высокий уровень</i>	1 2 3	психодиагностика
<b>2. Метапредметные результаты</b>				
2.1. Работа с литературными источниками и Интернет-ресурсами	Самостоятельность в подборе и анализе литературы, в пользовании электронных ресурсов	- <i>минимальный уровень</i> умений (испытывает серьезные затруднения при работе с информацией) - <i>средний уровень</i> (работает с помощью педагога) - <i>высокий уровень</i> (работает с информацией самостоятельно, не испытывает трудностей)	1 2 3	проекты
2.2. Умение выполнять проектную работу	Самостоятельность в проектной деятельности	- <i>минимальный уровень</i> умений (испытывает серьезные затруднения) - <i>средний уровень</i> (работает с помощью педагога) - <i>высокий уровень</i> (ведет работу)	1 2 3	Участие в научно-практических конференциях и соревнованиях

		самостоятельно, не испытывает трудностей)		
<b>3. Личностные результаты</b>				
3.1 Познавательная активность	Уровень познавательной активности (участие в соревнованиях)	<i>Низкий уровень:</i> принимает участие иногда <i>Средний уровень:</i> активное участие <i>Высокий уровень:</i> является инициатором идей и проектов	1 2 3	Наблюдение Участие в проектах
3.2. Умение слушать и слышать педагога, товарища	Адекватность восприятия информации	- <i>минимальный уровень</i> умений восприятия информации - <i>средний уровень</i> умений восприятия информации - <i>высокий уровень</i> умений восприятия информации	1 2 3	Наблюдение
3.3.. Умение выступать перед аудиторией	Свобода владения и подачи подготовленной информации	- <i>минимальный уровень</i> умений владения и подачи информации (с листа) - <i>средний уровень</i> умений владения и подачи информации (с листа с привлечением ТСО) - <i>высокий уровень</i> умений владения и подачи информации (свободно с использованием ТСО)	1 2 3	Открытые занятия, конференции, семинары, круглые столы,
3.4.. Умение вести полемику, участвовать в дискуссии	Самостоятельность в построении дискуссионного выступления	- <i>минимальный уровень</i> умений в построении дискуссионного выступления - <i>средний уровень</i> умений владения в построении дискуссионного выступления - <i>высокий уровень</i> умений в построении дискуссионного выступления	1 2 3	Дискуссии, круглые столы с подведением итогов
		- <i>высокий уровень</i> умений в построении дискуссионного выступления		

Максимальное количество баллов 27 баллов.

Уровень личностного развития: от 13 до 15 – низкий уровень, от 16 до 21 – средний, от 22 до 27 – высокий

Градация принята условно (информация может быть представлена ученику только в процессе строго индивидуальной работы в совокупности с картой саморазвития, заполняемой учеником)



Традиционная система оценивания страдает расплывчатостью, произвольностью норм выставления отметки, зависит от внешней оценки, повышает уровень тревожности у детей и мало способствует эффективному обучению. Возникает непонимание между учениками и учителем. При традиционной системе оценивается не процессуальное, содержательное движение ученика к цели, а лишь конечный результат.

Если получать хорошие оценки будет слишком просто, то ученики потеряют побудительный мотив к учению. Незаслуженно плохая оценка может привести к тому, что ученик вообще перестанет учиться, а не объективная оценка может отрицательно повлиять на весь образовательный процесс. Система оценивания должна ориентировать учеников на успех, способствовать развитию их самооценки, поощрять и стимулировать учение, давать информацию об уровне выполнения программы.

Практические работы способствуют развитию практических умений и навыков, необходимых в жизни (умение измерять, наблюдать, анализировать результаты и т.д.) и вместе с тем коррекции интеллектуальной недостаточности, развитию умения выделять главное и второстепенное в предмете, уточнять и сравнивать предметы.

К практическим методам относятся наблюдение, проведение проектных и самостоятельных работ, работы по определению и описанию программ и т.д.

Главным методом, используемым при знакомстве младших школьников с окружающим миром является наблюдение. На первом этапе используются в основном общеклассные кратковременные эпизодические наблюдения. Постепенно они становятся более длительными, систематическими. При написании прошлой методики, мы отметили важность добавления общеобразовательных тем в методологическое пособие(путешествие по миру, путешествие по времени и т.д) помогут легче закрепить необходимый материал



Рисунок 2. Методика Роботек

## Антарктида



12

Copyright © ИП Шапкарбеков Санжар Жақсыбекович.

Рисунок 3. Методика Роботек

## Пингвин



13

Copyright © ИП Шапкарбеков Санжар Жақсыбекович.

Рисунок 4. Методика Роботек



Сборка «Пингвин»

37

Copyright © ИП Шапкарбеков Санжар Жақсыбекович.

Рисунок 5. Методика Роботек

На рисунках 2-5 показан пример занятия для 1 классов. Как связать самопознание, географию и путешествие с методикой по робототехнике.

Проверка и оценка достижений школьников является существенной составляющей процесса обучения. Этот компонент учебно-воспитательного процесса должен соответствовать современным достижениям педагогической науки, требованиям социальной ситуации и приоритетам образования в первом звене школы.

Однако имеется целый ряд затруднений в выборе способа и средств процессуального оценивания, критериальной базы для оценки достижений детей, выстраивания схемы анализа результатов.

Несомненным достоинством этих заданий является быстрота выполнения: учителю не нужно проводить предварительную организационную подготовку (ксерокопировать листы для учащихся или оформлять на доске тексты заданий), учащиеся сразу получают результат, а со временем они и сами смогут составлять подобные задания, проводить взаимоконтроль. Ученик вовлечен в творческий поиск: он мыслит, анализирует, сопоставляет.

Проведенный анализ педагогической и методической литературы по проблеме исследования позволил выделить следующие методы и приемы оценивания знаний учащихся на уроках окружающего мира:

- 1) графические программированные задания;
- 2) графический диктант;
- 3) цифровой диктант;
- 4) работа с ключами;
- 5) задания тезисного характера;
- 6) тестирование.

1. Графические программированные задания включают в себя до 10 высказываний, некоторые из которых – неверные. Ученики в тетрадях чертят отрезок длиной 10 см, делят его на 10 частей и проставляют под ним цифры от 1 до 10. Одно деление – это ответ на вопрос. Учитель читает вслух высказывания – дети графически отмечают на отрезке свой ответ (если высказывание неверное – ставят прочерк, если верное – знак «+»). Для проверки задания учитель может изготовить из кальки или прозрачной пленки шаблон, что позволит ему быстро проверить правильность ответов. Пользуясь таким шаблоном, дети и сами смогут определять правильность выполнения ими задания. Еще один вариант проверки – по образцу, открытому на доске.

2. Графический диктант может включать до 10 высказываний, некоторые из которых – неверные. Ученики в тетрадях чертят отрезок длиной 10 см, делят его на 10 частей и проставляют под ним цифры от 1 до 10. Одно деление – это ответ на вопрос. Учитель читает вслух высказывания – дети

графически отмечают на отрезке свой ответ (если высказывание неверное – ставят прочерк, если верное – знак «^»). В результате выполнения работы у них образуется «график», учитель, пользуясь шаблоном легко может проверить правильность написания диктанта.

3. Цифровой диктант. В программированных карточках под порядковыми номерами перечисляются объекты природы, которые требуется разделить на группы. Ученики могут предложить сгруппировать эти объекты по разным признакам. Учитель выслушивает все варианты ответов. Некоторые варианты можно обсудить, а один из них учащиеся выполняют самостоятельно. Можно предложить выполнить это задание по вариантам. Названия групп записывают в тетрадь, рядом указывают номера объектов этой группы. Приведем пример. Тема: «Какие бывают зубчатые передачи» (2-й класс) Что проверяет работа: 1) знание важнейших признаков; 2) умение группировать передачи на основе этих признаков.

4. Работа с ключами. Программированная карточка состоит из перечня вопросов, которые имеют свой порядковый номер, и перечня ответов, расположенных ниже под буквенными символами. При коллективном опросе учитель зачитывает вопросы, а ключи обязательно записывает на доске. Каждому вопросу соответствует свой ключ – правильный ответ. Вопросы должны быть сформулированы четко и понятно. Ключи не должны содержать подсказки. Ученик вовлечен в творческий поиск, он должен размышлять, анализировать, сопоставлять.

5. Задания тезисного характера. Задание состоит из 8-10 тезисов, в которых говорится о каком-либо объекте: описывается его внешний вид, перечисляются свойства, повадки и т. д. Например, дается перечень объектов, Учащиеся должны расставить по списку номера соответствующих тезисов. Обязательное условие: приводимые в тезисах признаки и свойства объектов должны быть знакомы детям. Оценка результатов. За каждый верный ответ – 1 балл. Высокий уровень – 10 баллов. Уровень выше среднего – 8-9 баллов. Средний уровень – 5-7 баллов. Низкий уровень – менее 5 баллов.

6. Тестирование. Существуют 4 формы заданий, которые используются при составлении теста: 1. Задания закрытой формы 2. Задания открытой формы 3. Задания на установление соответствия 4. Задания на установление правильной последовательности. Использование тестовых заданий сопряжено с необходимостью соблюдения требований, которые предъявляются к ним.

## Проверочный тест Уровень С4

За каждый правильный ответ - 1 балл. Максимум - 13 баллов.

ФИО \_\_\_\_\_

1. Укажите, какие передачи повышающие, а какие - понижающие (макс. 4 балла)

**А** 3:2:9:2:5

**Б** 15:2

**В** 8:45

**Г** 10:4:9:2:6

Повышающая:

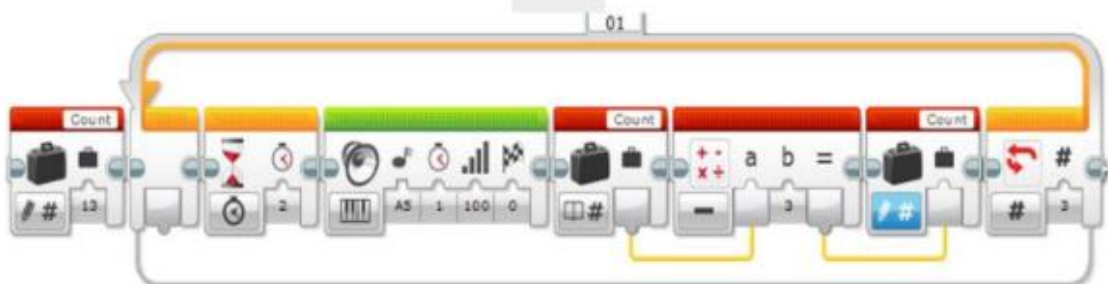
Понижающая:

\_\_\_\_\_ Баллов

2. Перечислите преимущества и недостатки гусеничных роботов (макс. 6 баллов)

\_\_\_\_\_ Баллов

3. Какое начальное значение переменной Count и чему будет она равна после выполнения всей программы? (макс. 2 балла)

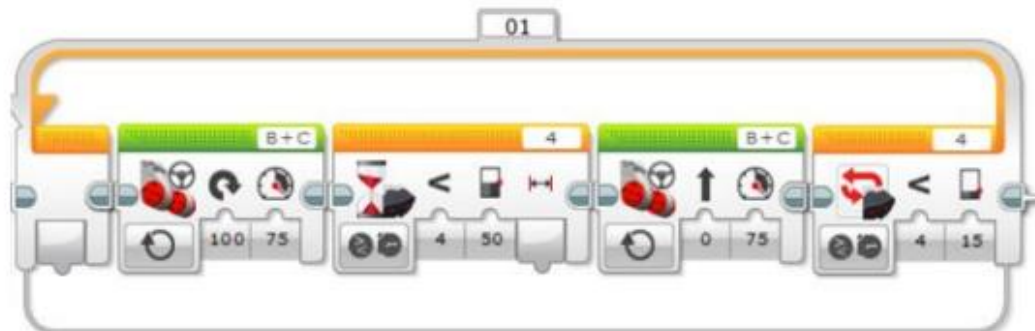


Начальное значение:

Конечное значение:

\_\_\_\_\_ Баллов

4. Объясни программу(макс. 1 балл):



\_\_\_\_\_ Баллов

**Общий итог:** \_\_\_\_\_

Рис. Пример тестирования в методике школы Роботек

Педагогический опыт показал, что в сочетании с другими видами проверки использование тестовых заданий является весьма эффективным инструментом, стимулирующим подготовку учащихся к каждому уроку и повышающим мотивацию к изучаемому предмету. Еще одним преимуществом тестов по сравнению с другими формами контроля является то, что все обучающиеся находятся в равных условиях, позволяющих объективно сравнивать их достижения; исключается субъективность учителя; результаты тестирования поддаются статистической обработке.

Главная задача уроков с применением тестов – способствовать осознанному усвоению материала, развитию логического мышления. Использование тестовых заданий позволяет определить, как учащиеся овладевают знаниями, умениями и навыками, а также проанализировать свою педагогическую деятельность. Обучающиеся смогут узнать о своих достижениях или пробелах в изучении определенной темы, тем самым у школьников развивается самоконтроль и возможность корректировать систему своего обучения. Успешное и эффективное применение методов тестирования целиком зависит от двух основных факторов. Во-первых, это отсутствие доступа посторонних к данным, содержащим информацию о правильных ответах. Во-вторых, это качество тестовых заданий.

Анализ реализации программированного контроля в начальной школе в течение ряда лет выявил его преимущества. Он позволяет повысить эффективность обучения, поскольку обеспечивает возможность за короткое время (5–15 минут) выявить уровень усвоения знаний, умений и навыков учащихся всего класса. Предложенные формы работы на уроке активизируют познавательную деятельность, вовлекая детей в творческий поиск, способствуют развитию желания учиться. Учитель, работая по этой методике, быстро получает наглядную картину успеваемости, ему сразу видны пробелы в знаниях как отдельных учеников, так и всего класса, не только по теме в целом, но и по отдельным ее фрагментам.

Интересы ребенка в период обучения могут меняться по мере его взросления. Задача как педагогов, так и родителей направить его и показать перспективы развития в области робототехники.

В первую очередь робототехника — это междисциплинарное обучение. Дети изучают на практике как общеобразовательные предметы, так и дополнительные навыки. Комплексный подход к образованию помогает познать окружающий мир во всем его многообразии. Кроме того, робототехника для детей обязательно подразумевает проектную деятельность, которая разбита на несколько этапов, а каждый из этапов приносит новые знания:

Постановка задачи: учит определять цель и прорабатывать шаги для ее достижения.





## Задание №2

Присоединяем ультразвуковой датчик

Робот шагает вперёд.

Если видит препятствие на расстоянии 30 см - шагает назад

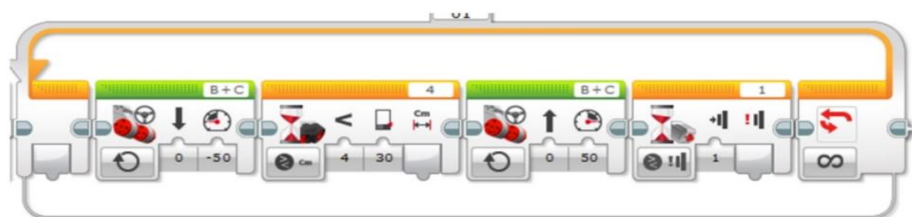
Когда робот коснется стены/препятствия - снова шагает вперёд

25

Copyright © ИП Шалкарбеков Санжар Жаксыбекович, 2021



## Ответ



26

Copyright © ИП Шалкарбеков Санжар Жаксыбекович, 2021

Рис. Пример построений практических задач в методике Роботек

- Разработка проекта: ребенок выявляет причинно-следственные связи, разрабатывает несколько вариантов решений задач с помощью технического творчества.
- Создание продукта, конструирование роботов: освоение практических навыков (программирование, работа с электроникой и тд).
- Тестирование: на этом этапе ведется выявление неполадок, их устранение.
- Обсуждение: закрепляются навыки работы в команде, общения со взрослыми, умение делать выводы.

Подобное построение занятия подразумевает развитие и инженерного, и креативного мышления. Дети учатся работать в команде, развивая коммуникативные навыки. Они привыкают к самостоятельной деятельности и параллельно за интересным занятием постигают азы робототехники, закладывают фундамент будущего профильного образования.

Робототехника для детей полезна в любом случае. Она формирует представление о развитии данной сферы, логическое мышление, инженерные



навыки. В дальнейшем все это подлежит развитию и пригодится, даже если ребенок не собирается связывать свою профессиональную сферу деятельности с конструированием роботов. Такие занятия полезны для расширения возможностей и кругозора ребенка.

Бок о бок с робототехникой идет соревновательная деятельность. Дети совместно с преподавателями могут принимать участие в различных соревнованиях, что само по себе делает их более собранными, организованными и дисциплинированными. Не говоря уже об углублении знаний в процессе подготовки. Подразумеваются разные уровни: внутри коллектива, города, области, возможен выход на WorldSkills компетенции Мобильная робототехника и так далее.

По итогу занятий робототехникой даже в рамках общеобразовательной программы дети получают навыки ведения проектной деятельности, самостоятельному поиску разнообразных решений, закрепляют теоретические знания по предметам на практике. Робототехника выступает на уроках новым средством наглядности, стимулирующим активное восприятие материала целого ряда дисциплин, поскольку демонстрации с использованием робототехнических средств отличаются более высоким качеством и скоростью предъявления данных, обеспечивают необходимое число повторений, а также могут сопровождаться различными эффектами (визуальными, механическими, звуковыми), что способствует концентрации внимания школьников на наиболее важных элементах учебного материала и повышает познавательный интерес. На основе проектов учащиеся работают в группах, чтобы «исследовать реальный мир». Образовательная робототехника помогает создать прекрасную среду для учащихся, где они могут найти решения интересных проблем и продемонстрировать их не только в теоретическом, но и в практическом аспектах. На сегодняшний день во многих странах мира наблюдается значительный интерес к данному научно-техническому направлению.

При конструировании робота ребенку предстоит решить ряд вопросов:

1. Какую задачу должен выполнять робот.
2. Какие функции для этого нужны.
3. Какие части робота нужны для выполнения функций.
4. Как объединить части в целую систему.

Таким образом, ребенок намечает план своих действий и поэтапно начинает достигать цели. Ответ на первый вопрос научит правильно и полно формулировать техническое задание к работе. Второй вопрос помогает установить причинно-следственные связи. При решении третьего вопроса ребенок может приступить к практической части. В итоге он получает знания о

базовых возможностях робота, о его узлах и системах, понимает механику движений и так далее.

На занятиях дети изучают механику роботов, программирование, комбинируют компоненты для роботов, создавая свои уникальные модели.

Этот способ обучения отлично подходит для тех, кто действительно заинтересован в робототехнике и хочет знать больше. Классы в таких секциях оборудованы всем необходимым для обучения. А сами занятия строятся по проверенному принципу: переход от простого к сложному. Очень важно живое общение: разбивать пару «ученик и педагог» - нельзя. Наставник даёт детям возможность самостоятельного изучения и творчества: на этапе «детской робототехники», то есть младшего школьного возраста, это очень важно для дальнейшего саморазвития и достижения успеха. «Взрослая» робототехника, предназначенная для изучения в старших классах, предполагает расширение круга задач.

Зачастую школьники уже знают, какое образование хотят получить и имеют конкретную цель. Педагоги часто готовят своих воспитанников к соревнованиям, расширяя и углубляя учебную программу.

Европейские ученые говорят о том, что инструментами развития образовательной робототехники являются не только учебные материалы, компьютерная литература и т.д., но и сами «драйверы» процесса:

- Учащиеся, которые непосредственно заинтересованы в образовании;
- Родители, которые могут выступать в качестве мотиваторов обучения;
- Учителя и школьная администрация, которые принимают решения о проведении уроков и о том, как выстраивать уроки;
- Коммерческие и некоммерческие организации, способствующие продвижению направления и оказывающие посильную помощь образовательным учреждениям.
- Правительственные инициативы;
- Университеты и предприятия-работодатели.
- Педагогическая деятельность, как известно, может осуществляться в разных формах, среди которых особое место занимает урок.

## Тематический план программы курса робототехники для 5-9 классов

Направление «Робототехника» с началом нового тысячелетия в большинстве стран стало занимать существенное место в школьном и университетском образовании. Предпосылками к этому стало быстрое развитие микроэлектроники, которое дало возможность создавать умные небольшие машины, а отсюда возникла большая потребность в специалистах, способных разрабатывать новые робототехнические комплексы. На данный момент, почти в каждом регионе Казахстана существует школьное образовательное направление «Робототехника», как в качестве дополнительного образования, так и как часть урока технологии.

В университетской же среде сообщество ученых, понимающих тенденции развития робототехники, растет с каждым годом. Отсюда появляется все больше научных работ, связанных со встраиваемыми робототехническими комплексами. Всемирно известные робототехнические организации, такие как JARA и FIRA, прогнозируют большой бум рынка сервисной робототехники в ближайшем будущем. Все перечисленные движения ведут с каждым годом к появлению роботов и робототехнических устройств в повседневной жизни, подобно тому, как телефон с циферблатом трансформировался в мощный карманный компьютер за двадцать с небольшим лет.

В настоящее время активное развитие робототехники увеличивает запрос к навыкам и знаниям специалистов. Уже в школьном возрасте учащимся необходимо уметь взаимодействовать с робототехническими устройствами, а тем, кто выбирает робототехнику своей профессией, – создавать и программировать роботов. Растущий объем информации диктует изменять привычный образовательный подход, а вместе с ним и инструментарий.

Именно по этой причине появились различные робототехнические конструкторы, такие как LEGO, HUNA, Tetrix, которые сейчас используются во многих образовательных учреждениях по всему миру. При этом с выхода первой линейки LEGO RCX уже прошло 18 лет. Назрела потребность в новом образовательном инструментарии, а вместе с ним и подходе обучения – образовательных программах. Данная программа является как продолжающей так и стартовой в курсе робототехники, рассчитана на учащихся 5-9 классов вне зависимости от их стартового уровня знаний и умений. Программа направлена на формирование и развитие инженерного мышления.

Инженерное мышление – мышление, направленное на обеспечение деятельности с техническими объектами, осуществляемое на когнитивном и инструментальном уровнях и характеризующееся как политехническое, конструктивное, научно-теоретическое, преобразующее, творческое, социально-позитивное

1. Инженерное мышление – это системное творческое техническое мышление, позволяющее видеть проблему целиком с разных сторон, видеть связи между ее частями;

2. Цель программы: формирование у учащихся инженерного мышления, в процессе приобретения знаний, умений и навыков конструирования, программирования, разработки робототехнических проектов.

Задачи:

### **Личностные**

1. Повышение мотивации учащихся к изобретательству и созданию собственных роботизированных систем;
2. Формирование у учащихся стремления к получению качественного законченного результата;
3. Организация и участие в играх, конкурсах и состязаниях роботов в качестве закрепления изучаемого материала и в целях мотивации обучения.

### **Метапредметные**

1. Развитие мелкой моторики, внимательности, аккуратности и изобретательности;
2. Развитие креативного мышления и пространственного воображения учащихся;
3. Формирование навыков проектного мышления, коммуникативных навыков работы в команде.

### **Предметные**

1. Ознакомление учащихся с комплексом базовых технологий, применяемых при создании роботов;
2. Реализация межпредметных связей с информатикой и математикой;
3. Решение ряда кибернетических задач, результатом каждой из которых будет работающий механизм или робот с автономным управлением;
4. Освоение умений и навыков конструирования, программирования и эффективного использования кибернетических систем.

### **Основные особенности программы:**

Программа предусматривает подготовку учащихся в области робототехники.

Учащиеся учатся проектированию, моделированию, конструированию, программированию и использованию роботизированных устройств.

Программа вариативная так, как в рамках ее содержания можно разрабатывать разные учебно-тематические планы и для ее освоения возможно выстраивание индивидуальных программ, индивидуальных траекторий (маршрутов) обучения. Программа открытая, предполагает совершенствование, изменение в соответствии с потребностями учащихся.

Содержание данной программы позволяет расширить кругозор и углубиться в основные направления робототехники: теорию автоматического

управления, техническое зрение и обработку информации. В основе программы лежит V-образный подход обучения, который предполагает низкий порог вхождения с постепенным погружением. В основе программы лежит множество практических задач. Результатом реализации всех задач становится законченное автономное робототехническое устройство, выполняющее поставленную задачу. Программа плотно связана с массовыми мероприятиями в научно-технической сфере для детей (выставками НТТ, состязаниями, конференциями), что позволяет, не выходя за рамки учебного процесса, принимать активное участие в конкурсах различного уровня: от уровня учреждения до регионального.

В основу представляемого курса робототехники положены такие принципы как:

- Целостность и непрерывность, означающие, что данная ступень является важным звеном единой подготовки по информатике и информационным технологиям. В рамках данной программы продолжается осуществление дополнительного более глубокого изучения предмета.
- Научность в сочетании с доступностью, строгость и систематичность изложения (включение в содержание фундаментальных положений современной науки с учетом возрастных особенностей обучаемых).
- Практико-ориентированность, обеспечивающая отбор содержания, направленного на решение практических задач: планирование деятельности, поиск нужной информации, инструментирования всех видов деятельности на базе общепринятых средств информационной деятельности.
- Принцип дидактической спирали как важнейший фактор структуризации в методике обучения информатике: использование имеющегося опыта обучаемых, затем его последующее развитие и обогащение, создающее предпосылки для дальнейшего научного обобщения.
- Принцип развивающего обучения – обучение ориентировано не только на получение новых знаний в области информатики и информационных технологий, но и на активизацию мыслительных процессов, формирование и развитие у учащихся обобщенных способов деятельности, формирование навыков самостоятельной работы.

5 класс.

Таблица 1. Тематический план 5 класса

Название модуля	Темы предмета	Урок
<b>Модуль 1. Конструирование и Программирование робота</b>	Датчики и их параметры.	4
	Основы программирования и компьютерной логики	9
	Практикум по сборке роботизированных систем	8
	Творческие проектные работы и	5

	соревнования	
		<b>26</b>
<b>Модуль 2. Спортивная робототехника</b>	Знакомство с соревнованиями	2
	Подготовка роботов для соревнований	4
	Отладка и улучшение программного кода	2
		<b>8</b>

### **Цели обучения.**

Модуль 1. Конструирование и Программирование робота:

1.1 Датчики и их параметры - Ознакомить с датчиками: датчиком цвета, ультразвуковым датчиком, датчиком касания, гироскопическим датчиком. Подключить датчики к базовой модели. Изучить принципы работы датчика Касания и его применения для написания простой программы. Изучить принципы работы ультразвукового датчика и его применения для написания простой программы. Изучить принципы работы датчика Цвета и его применения для написания простой программы. Изучить принципы работы гироскопического датчика и его применения для написания простой программы.



Рисунок 6. Ультразвуковой датчик расстояния в EV3 Mindstorms



Рисунок 7. Датчик касания в EV3 Mindstorms



Рисунок 8. Датчик цвета в EV3 Mindstorms



Рисунок 9. Гироскопический датчик наклона в EV3 Mindstorms

1.2 Основы программирования и компьютерной логики - Научить работать в среде программирования с блоками: Блок Экран, Блок Текст. Ознакомить с блоками вкладки «Управление операторами»: блок «Начало», «Ожидание», «Цикл», «Переключатель», «Прерываний цикла». Использовать блок «Цикл» для выполнения заданных действий: неограниченное количество раз; с подсчетом; по времени; с использованием логического значения. Применять циклы при решении задач на движение.

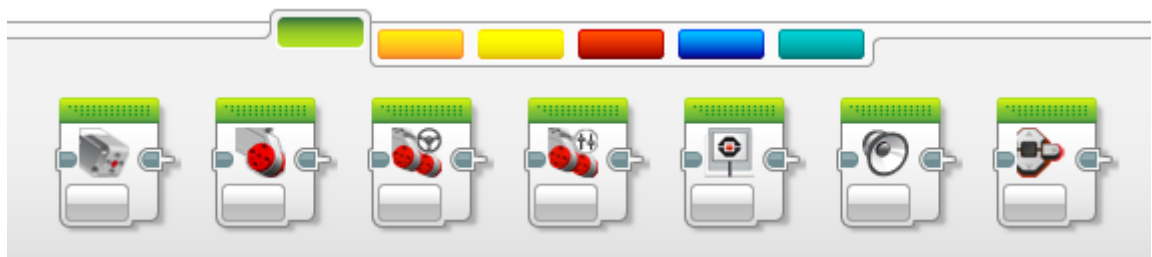


Рисунок 10. Блоки «Движения» в среде программирования EV3 Mindstorms

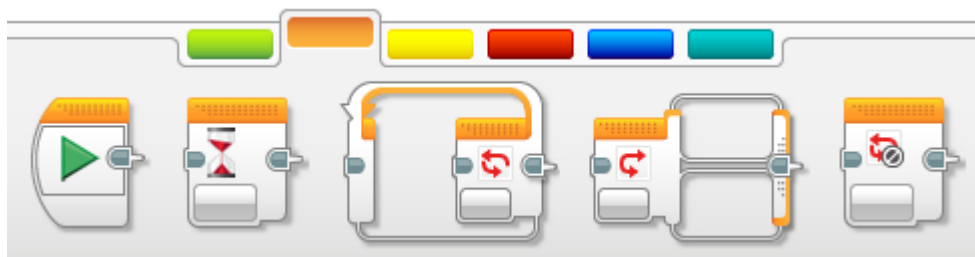


Рисунок 11. Блоки «События» в среде программирования EV3 Mindstorms

1.3 Практикум по сборке роботизированных систем - "Собрать модели роботов для решения задач с использованием разных видов датчиков:

Задачи на измерение освещенности, определения и распознавания цветов (датчик Цвета)

Задачи на измерения расстояний до объектов, сканирование местности (ультразвуковой датчик)

Задачи на движение: криволинейное движение, по замкнутой траектории (гироскопический датчик).

Задачи на реакцию робота на звук, цвет, касание. Использование таймера. Решение задач на выход из лабиринта. Ограниченное движение."

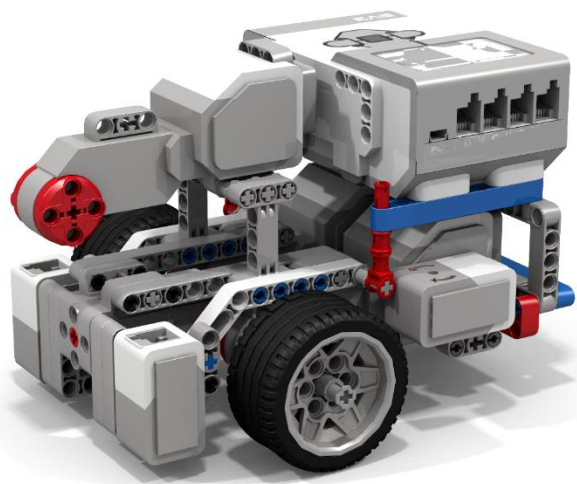


Рисунок 12. Пример сборки робота на базе набора Lego Mindstorms EV3

1.4 Творческие проектные работы и соревнования - "Анонсировать соревнования в классе. Презентовать идеи. Ознакомить с правилами соревнований.

Сконструировать проекты «Движение по заданной траектории», «Кегельринг» для проведения соревнований роботов на полигонах.

Сконструировать собственные модели роботов. Запрограммировать и провести испытания собственных моделей роботов.

Завершить создание моделей роботов для итоговой выставки. Подвести итоги работы учащихся и обсудить за круглым столом."





Рисунок 13. Соревнования по Кегельринг

## Модуль 2. Спортивная робототехника

2.1 Знакомство с соревнованиями - "Познакомится с соревнованиями по робототехнике. Изучить регламент и правила разных соревнований"

2.2 Подготовка роботов для соревнований - Сборка и программирование робота по правилам соревнований

2.3 Отладка и улучшение программного кода - Тестирование робота, исправление выявленных проблем

## 6 класс.

Таблица 2. Тематический план 6 класса

Название модуля	Темы предмета			Урок
<b>Модуль 1. 3D моделирование на Sketchup</b>	Введение.	Основные понятия	Трехмерное пространство	1
	Интерфейс программы	Инструменты рисования	SketchUp.	2
	Камеры,	навигация	в сцене, ортогональные проекции (виды)	2
	Инструменты	модификации: вдавить/вытянуть		2
	Инструменты модификации: Follow Me (Следуй за мной)			2
	Инструменты модификации: Offset			2

	(Контур), Move (Перемещение)	
	Инструменты модификации: Rotate (Повернуть) и Scale (Масштабировать)	2
	Инструменты измерения	1
	Управление инструментами рисования	2
		<b>16</b>
<b>Модуль 2. Программирование робота на языке программирования Python</b>	<b>Основы программирования и компьютерной логики на языке программирования Python</b>	18
		<b>18</b>

## **Цели обучения.**

### Модуль 1. 3D моделирование на Sketchup

1.1 Введение. Основные понятия компьютерной графики. Трёхмерное пространство - Ознакомить с понятиями компьютерной графики: 2D и 3D. Сформировать понятия трёхмерного пространства

1.2 Интерфейс программы SketchUp. Инструменты рисования - Ознакомить с интерфейсом программы, рабочей областью, панелями инструментов: основная и дополнительная.

1.3 Камеры, навигация в сцене, ортогональные проекции (виды) - Отработать навыки с командами меню «Камера»: «Орбита», «Панорама», «Масштаб» и т.д. Ознакомить с режимами навигации в сцене. Отработать действия навигации в сцене на готовом объекте.

1.4 Инструменты модификации: вдавить/вытянуть - Отработать навыки создания графических примитивов. Освоить инструмент Push / Pull («Тяни-толкай») при модификации графического примитива для создания объёмных фигур и регулировки толщины объекта

1.5 Инструменты модификации: Follow Me (Следуй за мной) - Сформировать навыки использования инструмента Follow Me («Следуй за мной») для выдавливания граней по выбранной направляющей линии: одиночной прямой, кривой, по краю другой грани. Отработать навыки создания графических примитивов с использованием инструмента Follow Me («Следуй за мной»)"

1.6 Инструменты модификации: Offset (Контур), Move (Перемещение) - Сформировать навыки использования инструментов Offset (Контур), Move (Перемещение). Отработать навыки модификации графических примитивов с использованием инструментов Offset (Контур), Move (Перемещение)

1.7 Инструменты модификации: Rotate (Повернуть) и Scale (Масштабировать) - Сформировать навыки использования инструмента Rotate

(Повернуть) для поворота граней фигур. Сформировать навыки использования инструмента Scale (Масштабировать) для изменения размеров и пропорций объектов целиком и их отдельных элементов для искажения. Отработать навыки модификации контуров графических примитивов с использованием инструментов Rotate (Повернуть) и Scale (Масштабировать)

1.8 Инструменты измерения - Сформировать навыки использования инструментов Tape Measure (Рулетка), Protractor (Угломер), Axes (Оси), Dimension (Указатели размеров) для проведения измерений фигуры и отображения их на чертеже. Отработать навыки измерений фигуры с помощью инструментов Tape Measure (Рулетка), Protractor (Угломер), Axes (Оси), Dimension (Указатели размеров).

1.9 Управление инструментами рисования - Сформировать навыки использования инструментов рисования Line (Линия), Arc (Дуга), Rectangle (Прямоугольник), Circle (Окружность), Polygon (Многоугольник), Freehand (От руки). Отработать навыки рисования фигур с помощью инструментов Line (Линия), Arc (Дуга), Rectangle (Прямоугольник), Circle (Окружность), Polygon (Многоугольник), Freehand (От руки)"

#### 1.10

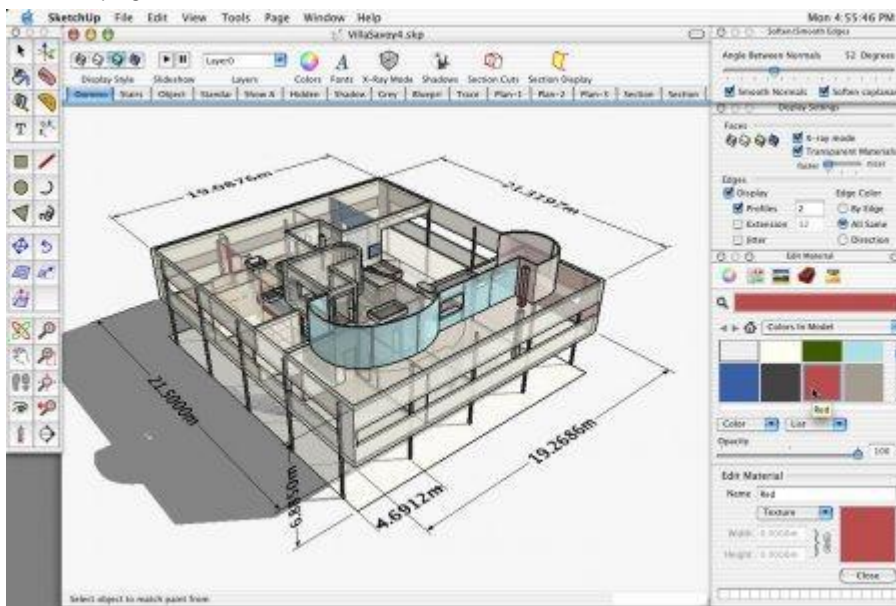


Рисунок 14. Интерфейс программы Google SketchUp

## Модуль 2. Программирование робота на языке программирования Python

2.1 Основы программирования и компьютерной логики на языке программирования Python - Собрать модели роботов для решения задач с использованием разных видов датчиков. Научить программировать движения робота и показания датчиков на языке Python.

7 класс.

Таблица 3. Тематический план 7 класса

Название модуля	Темы предмета	Урок
Модуль 1. 3D моделирование и печать	Инструменты панели Sandbox (Песочница). Работа с рельефом	2
	Инструменты Text (Текст), 3D Text (3D-текст)	2
	Инструмент Paint Bucket (Заливка)	2
	Группы и компоненты	2
	Экспорт, импорт растровых, векторных, 3D-моделей, распечатка на 3D-принтере	2
	Творческая практическая работа «Конструирование дома»: возведение фундамента, стен и крыши; окна и дверные проемы, лестница и балкон. Моделирование приусадебного участка двора.	6
		<b>16</b>
Модуль 2. Программирование микроконтроллера MicroBit	Что такое МикроБит?	2
	Алгоритмы	3
	Переменные	2
	Условные операторы	4
	Циклы	2
	Сенсоры	3
	Радио и связь	2
		<b>18</b>

### Цели обучения.

#### Модуль 1. 3D моделирование и 3D печать

1.1 Инструменты панели Sandbox (Песочница). Работа с рельефом - Сформировать навыки использования панели инструментов Sandbox (Песочница) для создания, изменения изображения местности. Отработать навыки работы с рельефом, используя набор контурных линий

1.2 Инструменты Text (Текст), 3D Text (3D-текст) - Сформировать навыки использования панели инструментов Text (Текст), 3D Text (3D-текст):

настройка форматов шрифтов и их атрибутов, типов и размеров выносных линий, другие параметры. Отработать навыки работы со шрифтами.

1.3 Инструмент Paint Bucket (Заливка) - Сформировать навыки использования инструмента Paint Bucket (Заливка) для текстурирования поверхностей и придания модели реалистичного вида. Отработать навыки работы с окном Materials (Материал), изменения атрибутов текстур в Редакторе материалов (Edit Material)

1.4 Группы и компоненты - Ознакомить с возможностями работы с группами и компонентами, различия между данными опциями. Сформировать навыки создания более сложных моделей с использованием групп и компонентов.

1.5 Экспорт, импорт растровых, векторных, 3D-моделей, распечатка на 3D-принтере - Ознакомить с возможностями экспорта, импорта 2D изображений. Ознакомить с особенностями импорта и экспорта 3D-моделей, форматами импорта/экспорта 3D-моделей. Сформировать навыки импорта/экспорта 3D-моделей для корректной работы с моделями в сторонних программах. Сформировать навыки распечатки на 3D-принтере

1.6 Творческая практическая работа «Конструирование дома»: возведение фундамента, стен и крыши; окна и дверные проемы, лестница и балкон. Моделирование приусадебного участка двора - Продемонстрировать приобретенные в процессе обучения навыки работы с инструментами:

Графические примитивы и их модификация;

Использование материалов;

Создание рельефа местности

Добавление компонентов и настройки отображения;

Сохранение и распечатка на 3D-принтере.

## Модуль 2. Программирование микроконтроллера MicroBit

2.1 Что такое МикроБит? - Знакомство с оборудованием MicroBit и средой программирования MakeCode; скачивание и запуск программ; с помощью дисплея и кнопок; дизайн и прототипирование с материалами. Виртуальные питомцы научит ваших учеников делать виртуального питомца в стиле тамагочи. Это отличный проект для обучения игровой логике, так как они узнают, как создавать системы, вызывающие у питомца чувство голода или жажды, а также узнают, как программировать кнопки для кормления и поения питомца.

2.2 Алгоритмы - Понимать четыре компонента, из которых состоит компьютер, и их функции. Поймите, что micro:bit принимает ввод и после обработки ввода производит вывод. Изучите разнообразие различных типов информации, которую micro:bit принимает в качестве входных данных. Примените эти знания, создав программу micro:bit, которая принимает входные данные и производит выходные данные.

2.3 Переменные - Понять, что такое переменные, зачем и когда их использовать в программе. Узнайте, как создать переменную, установить для нее начальное значение и изменить значение переменной в программе micro:bit. Узнайте, как создавать осмысленные и понятные имена переменных. Поймите, что переменная может хранить одно значение за раз. Поймите, что когда вы обновляете или изменяете значение, хранящееся в переменной, новое значение заменяет предыдущее значение. Узнайте, как использовать основные математические блоки для сложения, вычитания, умножения и деления переменных. Примените полученные знания и навыки для создания уникальной программы, в которой переменные используются как неотъемлемая часть программы.

2.4 Условные операторы - Понять, что такое условные операторы, а также зачем и когда их использовать в программе. Научитесь использовать логические блоки «if...then» и «if...then...else». Практикуйтесь в использовании логических блоков, чтобы разные условия давали определенные результаты. Продемонстрируйте понимание и примените навыки, сотрудничая с одноклассниками, чтобы создать игру, в которой используется micro:bit, и программу, которая правильно и эффективно использует условные выражения.

2.5 Циклы - Понимание ценности итерации в программировании. Понимание цикла как формы итерации. Узнайте, как и когда использовать циклические блоки «повторять», «пока» и «для». Примените вышеуказанные знания и навыки для создания уникальной программы, в которой итерации и циклы используются как неотъемлемая часть программы.

2.6 Сенсоры - Наклон и измерение шага (тряска, жесты); Знакомство с техникой и концепцией кодирования — случайные числа. Программирование - отображение направления компаса. Программирование — простой термометр

2.7 Радио и связь - Понять, как использовать радиоблоки для отправки и получения данных между микробитами. Понимать конкретные типы данных, которые можно отправлять по радио

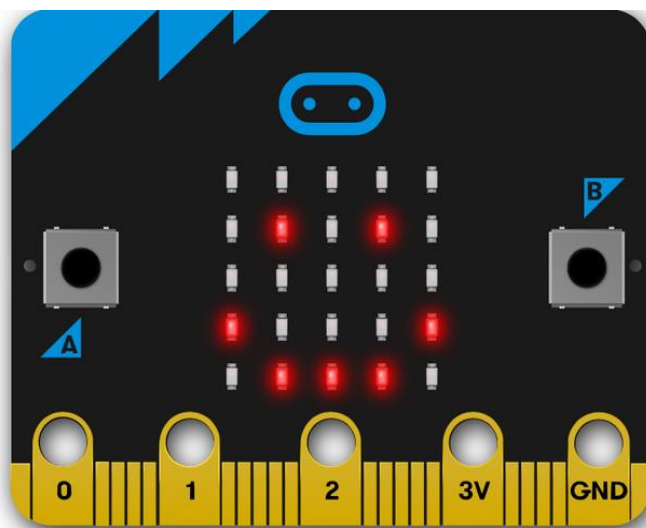


Рисунок 15. Micro:bit

## 7 класс.

Таблица 4. Тематический план 7 класса

Название модуля	Тема	Кол-во часов
Робототехника на основе Arduino	Введение в Arduino. Что такое микросхема и микроконтроллер. Установка программного обеспечения, подключение Arduino.	2
	Схемы электрические принципиальные. Правила построения схем. Линейка радиоинженера.	2
	Спецификация электро- радиоэлементов. Обозначение наименования	2
	Отладка программы	2
	Arduino и датчики.	2
	Практические работы со светодиодами.	4
	Практические работы со светодиодами.	4
	Практическая работа с пьезоэлементом и кнопками.	2
	Практическая работа с кнопками.	4
	Практическая работа с 7-сегментным индикатором и выходным сдвиговым регистром.	2
	Практическая работа с термистором.	4
	Практическая работа с инвертирующим триггером Шмитта.	2
	Arduino и сенсорная панель, датчики температуры и влажности.	2
	<b>34</b>	

### Цели обучения.

Модуль 1. Робототехника на основе Arduino.

- 1.1 Введение в Arduino. Что такое микросхема и микроконтроллер. Установка программного обеспечения, подключение Arduino - Сформировать понятия «микросхема» и «микроконтроллер». Ознакомить с элементами набора Arduino. Ознакомить со средой разработки программ.





Рис. Стандартный набор Ардуино

1.2 Схемы электрические принципиальные. Правила построения схем. Линейка радиоинженера - Ознакомить с элементами принципиальных электрических схем. Научить читать схемы и чертить посредством линейки радиоинженера. Последовательные и параллельные цепи. Закон Ома.

1.3 Спецификация электро - радиоэлементов. Обозначение наименования - Познакомить с резисторами, конденсаторами и их маркировкой. Познакомить с диодами, светодиодами, транзисторами

1.4 Отладка программы - Ознакомить с детальной процедурой отладки программы. Научить находить логические ошибки в коде и исправлять их.

Arduino и датчики - Ознакомить с основными функциями и назначением датчиков.

1.5 Практические работы со светодиодами - "Использовать функций `digitalWrite( )` и `digitalRead( )`. Использовать функции `analogWrite( )` и `analogRead()`. Отработать на практике алгоритм включения/выключения светодиода. Научить определять состояния кнопки (замкнута или разомкнута)."

1.6 Практические работы со светодиодами - "Закрепить на практике алгоритм включения/выключения светодиода. Использовать переменные и функции. Использовать управляющие конструкции `if( )` и `switch( )`."

1.7 Практическая работа с пьезоэлементом и кнопками - Отработать на практике свойства пьезоэлемента (пищалка) и фоторезистора. Закрепить на практике свойства пьезоэлемента (пищалка) и работу кнопок.

1.8 Практическая работа с кнопками - "Закрепить на практике работу кнопок и изменение яркости светодиода. Научить делать из тактовой кнопки триггер. Применить на практике полученные знания для создания игрушки на реакцию."

1.9 Практическая работа с 7-сегментным индикатором и выходным сдвиговым регистром - Отработать на практике работу с 7-сегментным индикатором. Отработать на практике работу с выходным сдвиговым регистром.



1.10 Практическая работа с термистором - Отработать на практике работу с термистором. Закрепить на практике работу со светодиодной шкалой

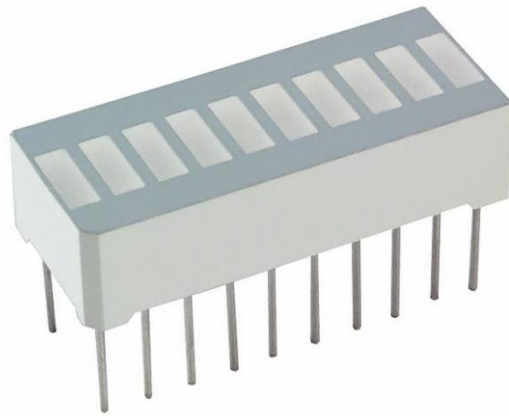


Рисунок 16. Светодиодная шкала

1.11 Практическая работа с инвертирующим триггером Шмитта - Закрепить на практике работу с тактовыми кнопками, пьезопищалкой. Отработать на практике работу с инвертирующим триггером Шмитта

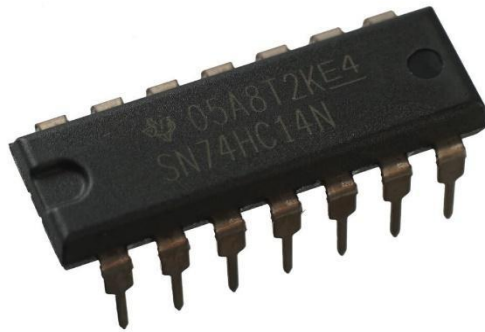


Рисунок 17. Инвертирующий триггер Шмитта

1.12 Arduino и сенсорная панель, датчики температуры и влажности - Ознакомить с основными функциями датчиков. Отработать на практике использование фоторезистора, датчиков температуры и влажности.

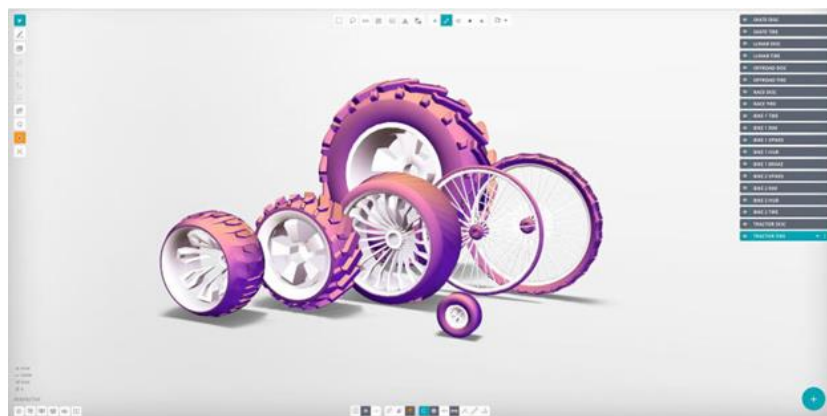


Рисунок 10. Интерфейс Vectary

## 9. SelfCAD

Бесплатная программа для сферы образования. При использовании в других целях, стоимость программы - около \$5,0 в месяц.

Страна разработчика – США.

SelfCAD фокусируется на том, чтобы быть лучшей браузерной бесплатной программой для 3D-моделирования для студентов по всему миру, которую не надо скачивать. Она популярна в американских школах, обучающих студентов 3D-дизайну (Рисунок 11). Простая и удобная в использовании, требующая короткий период обучения, SelfCAD имеет все необходимые инструменты для создания моделей. Также в ней есть инструменты нарезки для подготовки файлов STL или G-кода к 3D-печати.



Рисунок 11. Интерфейс SelfCAD

## 10. BlocksCAD

Программа предоставляется бесплатно. Есть платные учебные версии для школ.

Страна разработчика – США.

BlocksCAD ориентирована на обучение и была создана для учащихся по модели STEM с 3-го по 8-й классы. Это облачное бесплатное программное обеспечение для 3D-моделирования, которое способствует изучению математики, вычислительного мышления, концепций кодирования и проектирования моделей для 3D-печати.

В свободной галерее имеется широкий выбор проектов, которые можно использовать в личных и классных работах, начиная от снеговиков, ювелирных колец, рыб и даже печально известного кафетерия. Отлично взаимодействует с OpenSCAD, предназначена для простой, веселой и удобной работы для детей. 3D-модели можно создавать с помощью красочных, простых в управлении блоков, и экспортировать либо в виде STL-файлов, либо в виде файлов для открытия и редактирования в OpenSCAD (Рисунок 12). Для новичков BlocksCAD предлагает обширные учебные пособия по основам 3D-моделирования и использованию 3D-программного обеспечения.



Рисунок 12. Интерфейс BlocksCAD

## 11. OpenSCAD

OpenSCAD – бесплатная, загружаемая программа, являющаяся по своей сути, программной средой. Была создана в 2010 году Мариусом Кинтелем и Клиффордом Вольфом (Рисунок 13). Регулярно выходят новые обновления и патчи. OpenSCAD крайне популярен среди 3D-дизайнеров, которые предпочитают скриптовый, не художественный метод проектирования.

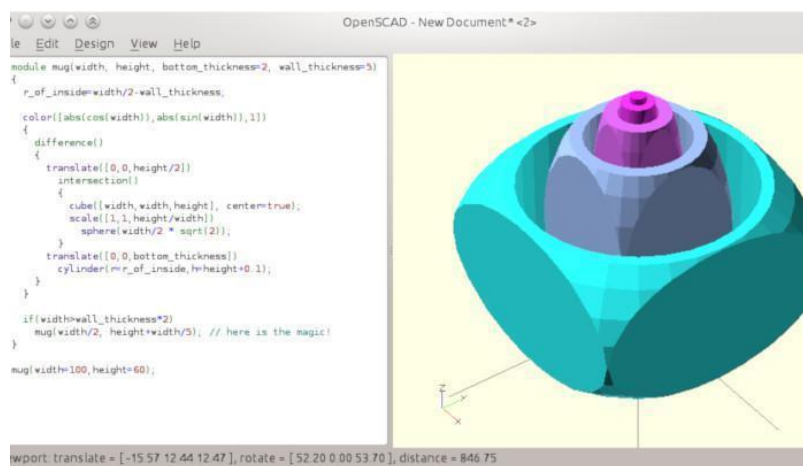


Рисунок 13. Интерфейс OpenSCAD

## 12. Wings 3D

Wings 3D - полностью открытое и бесплатное программное обеспечение для 3D-моделирования (Рисунок 14), которое активно разрабатывается и совершенствуется с 2001-года. Программа является относительно старой, но при этом удобной в использовании, как Vectary или TinkerCad, очень хорошо работает с персонажами, настольными моделями и другими проектами по созданию 3D моделей. Довольно легко работать с функциями по настройке моделей, например, лепкой, соединением, резкой, сгибанием - прост в изучении для начинающих.

Каждый пункт меню контекстный, поэтому, во время щелчка правой кнопкой мыши появляются различные опции в зависимости от того, что вам может понадобиться. Отзывчивый интерфейс программы экономит время и

делает её отличным инструментом для 3D-моделирования для начинающих и продвинутых.

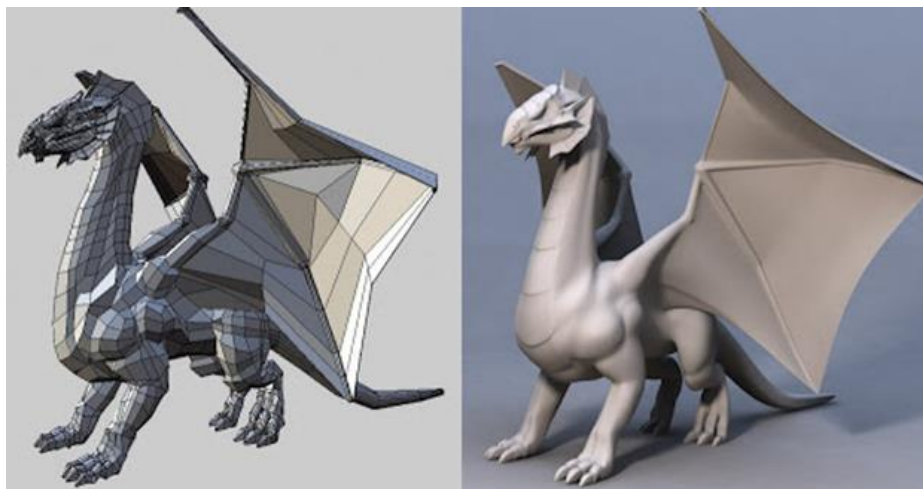


Рисунок 14.

### ***DesignSpark Mechanical***

#### ***Общая информация:***

- Стоимость: бесплатно;
- Тип работы: Твёрдотельное проектирование;
- Предназначен для обывателей и продвинутых пользователей.

***Примечательная особенность:*** Имеет обширную библиотеку 3D-моделей от промышленных поставщиков и возможность создания различных спецификаций для проектов (инструменты САПР для электротехники и печатных плат, и т.д.) (Рисунок 15).

Программное обеспечение САПР идеально подходит как для профессионалов, так и для продвинутых любителей. Пользовательский интерфейс относительно прост, программное обеспечение работает довольно быстро.. У пользователей также есть возможность создать спецификацию, в которой рассчитывается стоимость печати потенциальных проектов 3D-дизайна.

DesignSpark Mechanical позволяет пользователям использовать встроенную библиотеку для собственных чертежей. Еще одна функция, которая может оказаться полезной для новых пользователей, — это функция визуального экспортирования, которая позволяет пользователям создавать 3D-модели только с изображения поверхности объекта. Программа многофункциональна и довольно удобна для изучения на начальном этапе изучения 3D-моделирования.



Рисунок 15. Интерфейс DesignSpark Mechanical

### ***FreeCAD***

#### ***Общая информация:***

- Стоимость: бесплатно;
- Тип работы: Твёрдотельное проектирование;
- Предназначен для обывателей и продвинутых пользователей.

***Примечательная особенность:*** Модели являются полностью параметрическими и пересчитываются по запросу отмены/возврата стека. Другие функции включают роботизированное моделирование, архитектурные инструменты и модуль пути для АП (автоматизированное производство).

Инструмент параметрического 3D-моделирования с открытым исходным кодом, позволяющий проектировать реальные объекты любого размера. Параметрический компонент упрощает редактирование дизайна (Рисунок 16). Пользователь может видеть историю своей модели и менять по желанию любые параметры в истории изменений, тем самым меняя модель. Значимым преимуществом приложения является то, что она полностью бесплатна, поэтому пользователи могут настроить свои модели, пользуясь всеми инструментами среды.

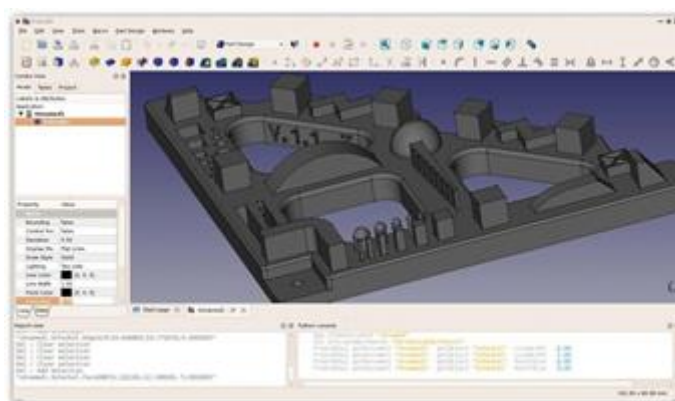


Рисунок 16. Интерфейс FreeCad

### ***OpenSCAD***

#### ***Общая информация:***

- Стоимость: бесплатно;
- Тип работы: Твёрдотельное проектирование;
- Предназначен для обывателей и продвинутых пользователей.



**Примечательная особенность:** Изначально разработан для программистов, модели генерируются и меняются с помощью набора скриптов (Рисунок 17).

OpenSCAD — это бесплатное программное обеспечение со множеством функций и уникальным способом создания моделей. Это программное обеспечение использует программный подход к 3D-моделированию, что делает его уникальным дополнением к этому списку программных инструментов для 3D-печати. Вместо традиционного интерактивного интерфейса моделирования, пользователи пишут код в файле сценария, описывающем параметры 3D-объекта. После того, как вы ввели свой код, вы можете просмотреть созданные вами формы, нажав кнопку «Compile (Компилировать)».

Еще одна замечательная функция OpenSCAD — это возможность импортировать 2D-чертежи и преобразовать их как трехмерные. Он использует профиль детали из чертежей, сделанных в стандартном программном обеспечении для создания эскизов, и использует для этого файлы формата SXF. Благодаря более сильному акценту на программировании, OpenSCAD крайне избирателен в пользователях. Тем не менее, это - один из популярнейших и высокоэффективных инструментов трехмерного моделирования на данный момент.

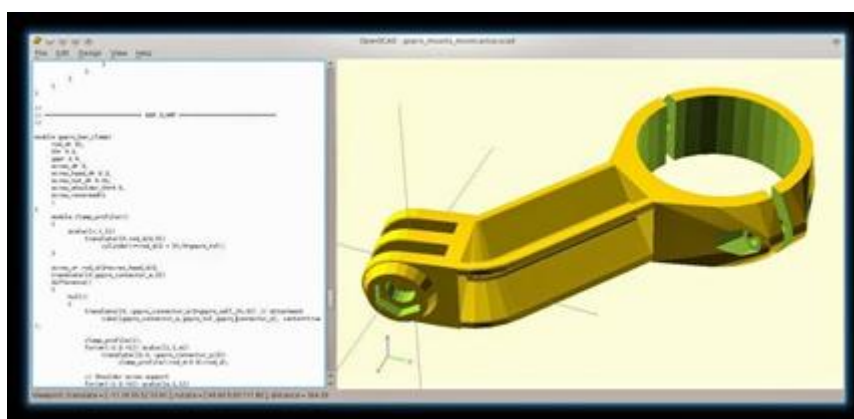


Рисунок 17. Интерфейс OpenSCAD

## 1.2. Программы для подготовки к печати: инструменты алгоритмизации процесса печати - слайсеры

Данный раздел содержит список программных инструментов для 3D-печати, которые помогают реализовать качественную трехмерную печать уже спроектированной модели. Использование *слайсеров* является самым простым способом преобразования 3D-модели в печатаемую модель, они “нарезают” модель САПР на слои и превращают модель в *G-код*, который представляет из себя алгоритмы печати объекта. Программное обеспечение слайсера также включает в себя настройки 3D-принтера, такие как температура, высота слоя, скорость печати и т. д. для G-кода. 3D-принтер может считывать этот G-код и создавать модель слой за слоем, следуя инструкциям, указанным в G-коде.

## *Ultimaker Cura*

Благодаря открытому исходному коду, программу Cura можно использовать при работе практически с любым 3D-принтером (Рисунок 18). Программа идеально подходит для начинающих, она интуитивно понятна и быстра, и довольно проста в использовании. Более продвинутые пользователи могут получить доступ к еще 200 настройкам для улучшения своих моделей.

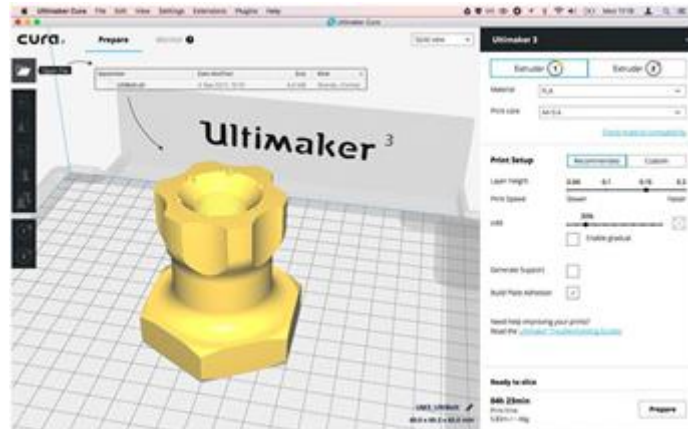


Рисунок 18. Интерфейс Ultimaker Cura

## *Simplify3D*

Simplify3D — чрезвычайно мощный инструмент для наиболее качественной нарезки (Рисунок 19), который помогает значительно улучшить качество 3D-печати. Simplify3D способен самостоятельно исправлять ряд проблем, связанных с распечаткой моделей и позволяет предварительно оценить конечный результат, помогая выявить множество проблем заранее. Является частично-платным.

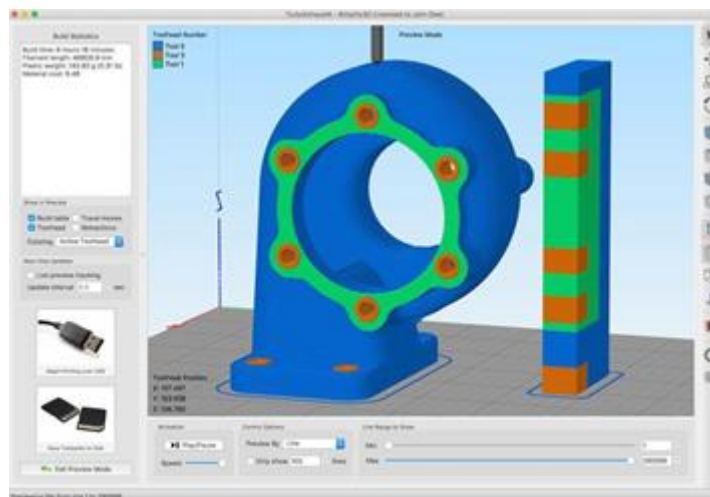


Рисунок 19. Интерфейс Simplify3D

## *Slic3r*

Данное программное обеспечение с открытым исходным кодом включает в себя поэтапную нарезку в режиме реального времени, предварительный просмотр в 3D и многое другое. Slic3r - один из наиболее широко

используемых программных инструментов для 3D-печати (Рисунок 20). Инкрементная нарезка режиме реального времени гарантирует, что при изменении параметра нарезку не нужно будет начинать сначала. G-код пересчитывается только для затронутых деталей. Конечным результатом является быстрая, гибкая и точная программа нарезки.

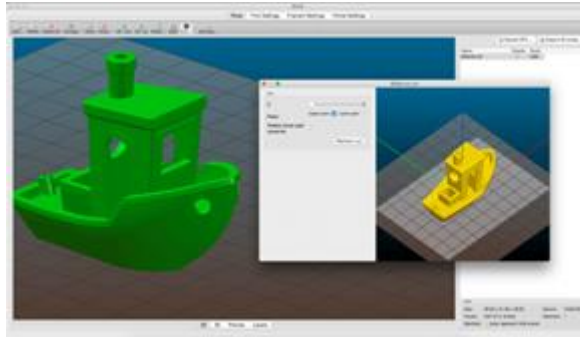


Рисунок 20. Интерфейс Slic3r

### ***Repetier***

Repetier - это программное обеспечение для слайсеров с открытым исходным кодом, поддерживающий три различных механизма нарезки: Slic3r, CuraEngine и Skeinforge (Рисунок 21). Repetier также может одновременно обрабатывать до 16 экструдеров с различными типами и цветами нитей, и предоставляет возможность визуализировать конечный результат перед печатью. Существует множество настроек и много работы, что делает Repetier идеальным для более продвинутых пользователей. Также в нем существует возможность удаленного доступа к своим принтерам с хостом Repetier.

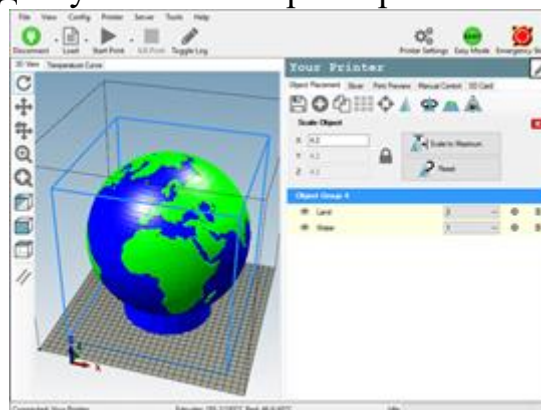


Рисунок 21. Интерфейс Repetier

### ***KISSlicer***

Это программное обеспечение для нарезки отличается довольно простым пользовательским интерфейсом (Рисунок 22). Тем не менее, в случае, когда нужен слайсер, с относительно простым интерфейсом и быстрым результатом, KISSlicer является довольно неплохим решением. Базовая бесплатная версия программы предназначена только для машин с одним экструдером.



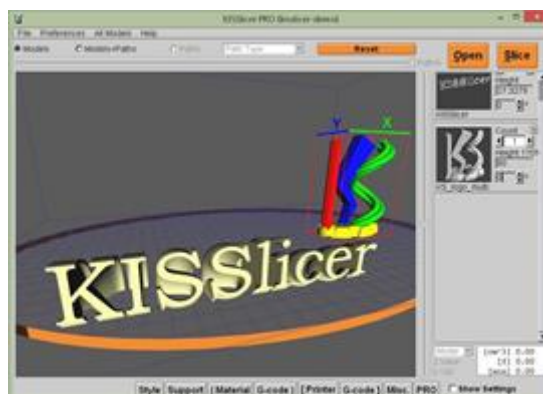


Рисунок 22. Интерфейс KISSlicer

### *ideaMaker*

Этот бесплатный слайсер является продуктом компании Raise3D и обеспечивает оптимальную быструю работу с большинством 3D-принтеров (Рисунок 23). Члены одной команды пользователей могут совместно использовать макеты печати, а также доступна функция автоматической или ручной правки изделий. Инструмент адаптивной высоты слоя позволяет программному обеспечению регулировать высоту слоя в зависимости от уровня детализации модели, максимально повышая качество печати и сводя к минимуму время печати. Также возможен удаленный мониторинг и управление с помощью Интернет.

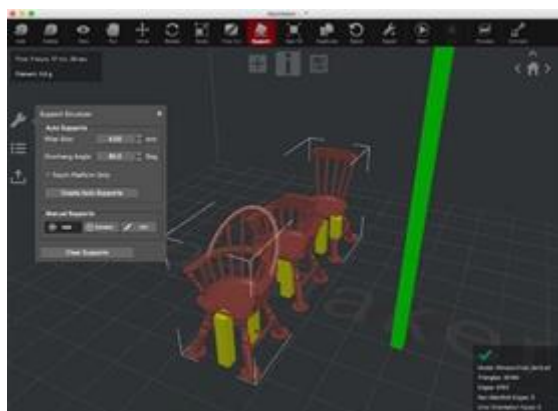


Рисунок 23. Интерфейс ideaMaker

### *OctoPrint*

Бесплатный веб-интерфейс с открытым исходным кодом, который позволяет удаленно управлять и контролировать 3D-принтеры (Рисунок 24). Он совместим с большинством 3D-принтеров и позволяет пользователям просматривать свои отпечатки с помощью встроенной веб-камеры. Печать можно запускать, приостанавливать и останавливать удаленно, а также доступны подключаемые модули для отслеживания статистики печати и отправки push-уведомлений о ходе выполнения задания.

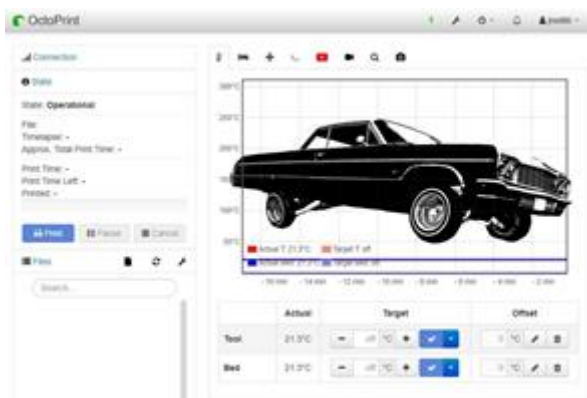


Рисунок 24. Интерфейс OctoPrint

### ***3DPrinterOS***

Платная программа, основная идея которой заключается в управлении всем процессом 3D-печати с помощью одной платформы (Рисунок 25). Пользователи могут редактировать и исправлять проекты, нарезать файлы STL из облака и даже отправлять файлы на печать из любой точки мира. Программное обеспечение также имеет возможность обмениваться файлами САПР.



Рисунок 25. Интерфейс 3DPrinterOS

## **2. Использование программ 3D-моделирования на практике**

В данной главе приведены практические примеры работ на наиболее популярных программах для 3D-моделирования и подробное описание процессов рендеринга и подготовки к печати (использование слайсеров и верстаков).

### **2.1. Применение на практике программы TinkerCad**

TinkerCad, вероятно, является одним из самых простых и интуитивно понятных инструментов для 3D-моделирования [4]. Она бесплатна, работает в веб-браузере, имеет понятный и логичный интерфейс, а простота использования делает ее популярной программой для многих неопытных пользователей, только начинающих свой путь изучения 3D-моделирования. В

этой главе мы рассмотрим основы проектирования в TinkerCad, создав модель простого гаечного ключа.

## **TinkerCad: Моделирование гаечного ключа**

### *Шаг 1: Начало работы*

Как в случае с любым веб-приложением, всё, что необходимо для начала работы с TinkerCad, — это зарегистрировать бесплатную учетную запись на веб-сайте <https://TinkerCad.com/>.

В конце регистрации пользователя встретит экран обучения TinkerCad. Пропустите это сейчас, закрыв его, а вместо этого перейдите на домашнюю страницу и нажмите «Создать новый дизайн».

Основной интерфейс программы показан на рисунке 26:

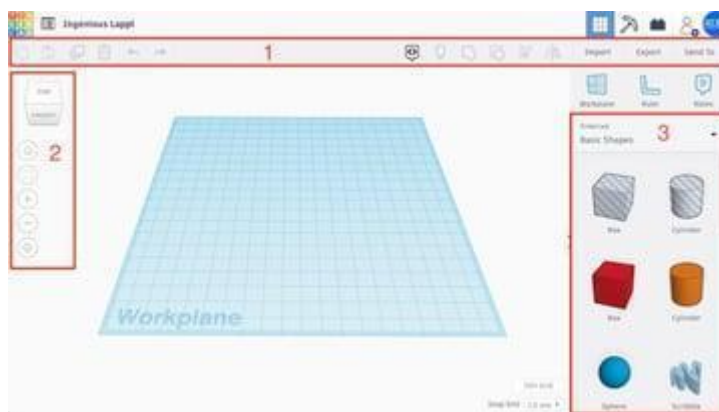


Рисунок 26. Интерфейс TinkerCad

Так выглядит основной интерфейс проекта. Центральным объектом является «Рабочая плоскость», на которой создаются модели и управляются инструментами, расположенных на боковом меню. Интерфейс также состоит из следующих элементов:

1. Верхняя панель инструментов с более общими инструментами, такими как «Копировать», «Вставить» и «Удалить» слева, и операциями проектирования, такими как «Группировать» и «Выровнять», справа.
2. Инструменты навигации для базовой ориентации, такие как увеличение и уменьшение масштаба. Обратите внимание, что эти специальные инструменты также можно использовать с помощью колесика мыши.
3. Панель Shape, содержащая все стандартные блоки для проектирования, включая базовые формы, такие как кубы, сферы и текст.

Начнем с того, что разделим ключ на три части



Рисунок 27. Разделим ключ на три части

Теперь, когда мы знакомы с интерфейсом TinkerCad, можем приступить непосредственно к дизайну.

«Разделяй и властвуй» — ключевой принцип 3D-моделирования. По сути, он говорит нам разбить часть на более мелкие формы и работать с ними по отдельности. Затем эти части будут собраны вместе, чтобы создать окончательный объект (Рисунок 27).

Этот метод особенно удобен при моделировании сложных объектов реального мира. Разделим нашу модель гаечного ключа на три основные части: кольцо, рукоятку и губку.

Эти блоки будут спроектированы в течение следующих трех шагов, а после этого они будут сшиты вместе. На шестом и последнем шаге мы добавим текст к гаечному ключу, чтобы можно было быстро определить его размер.

### *Шаг 2: Дизайн кольца*

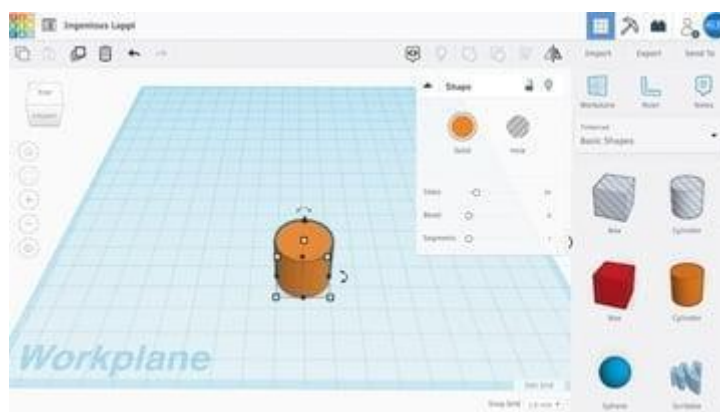


Рисунок 28.

Базовые фигуры можно перетаскивать на рабочую плоскость

Кольцо гаечного ключа можно представить в виде цельного диска с полым круглым участком посередине, что-то вроде очень толстой шайбы.

Чтобы создать эту фигуру, сначала перетащим цилиндр из панели «Основные фигуры» на рабочую плоскость. Щелкнув по фигуре, можно увидеть ее параметры в разделе «Форма» (Рисунок 28). Мы не будем вдаваться в них сейчас, но изучение таких функций, как «Bevel», может быть интересным и полезным.

При выборе вы заметите, что цилиндр имеет пять белых квадратов или «ручек», четыре из которых находятся в нижней плоскости фигуры, а один — над цилиндром. Настройка размеров приведена на рисунке 29.

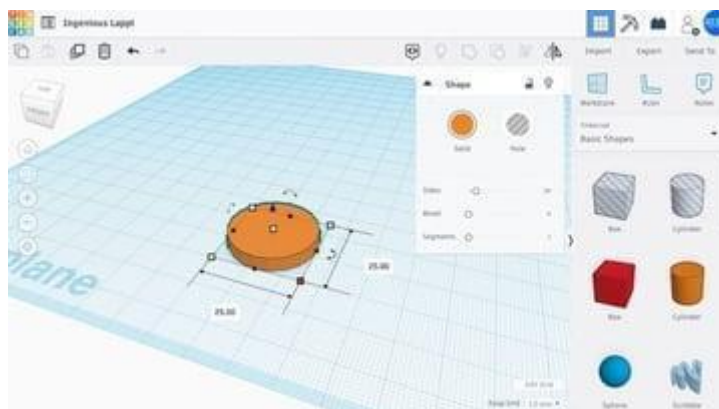


Рисунок 29. Настройка параметров фигуры

Четыре указателя в нижней плоскости можно использовать для регулировки длины и ширины цилиндра. Черные указатели контролируют длину, ширину и высоту. Мы хотим, чтобы наше кольцо имело диаметр 25 мм. После того, как вы нажмете на белый квадратный маркер, отобразится небольшое поле с текущими размерами.

Либо перетащите квадратные ручки, пока цилиндр не будет иметь длину и ширину 25 мм, либо введите значения, щелкнув поле размера и нажав Enter. (Tinkercad по умолчанию привязывает любое перетаскивание к любому значению, отображаемому в правом нижнем углу, например, «1 мм».)

Ручка над цилиндром регулирует его высоту. Чтобы наш ключ имел высоту 4 мм, отрегулируйте его так же, как длину и ширину.

#### *Булевы операции. Создание полых объектов*

Сплошной диск готов, теперь необходимо создать отверстие посередине объекта (Рисунок 30).

Tinkercad работает по принципу логического проектирования, который позволяет вычитать объекты для создания пустых областей и добавлять объекты для формирования новых форм. (Обратите внимание, что эти термины не используются явно.)

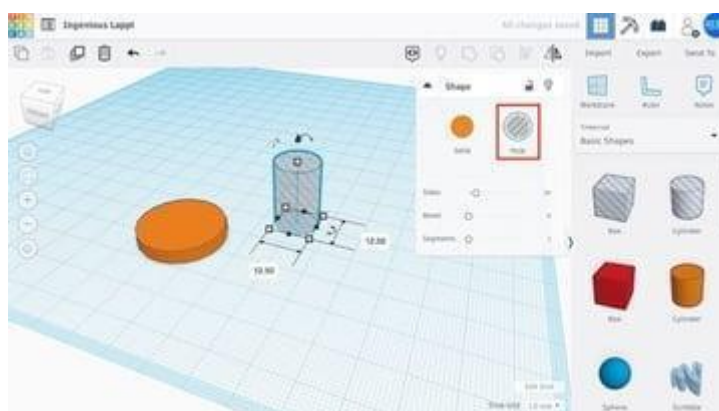


Рисунок 30. Форма «Hole (Отверстие)» удаляет материал из других твердых форм

На этом этапе мы будем использовать операцию вычитания, чтобы создать круглое отверстие в нашем кольце. Сначала, перетащите еще один цилиндр на рабочую плоскость и отрегулируйте его длину и ширину до 12,5 мм, что соответствует диаметру отверстия, которое нам нужно.

Выбрав новый цилиндр, нажмите «Отверстие» в правом верхнем окне. Это говорит TinkerCad рассматривать эту форму как нечто, что нужно вычесть.

### *Выравнивание (Align)*

Теперь, необходимо расположить отверстие в форме цилиндра в середине сплошного диска. Для этого, будем использовать инструмент выравнивания на верхней панели инструментов.

Выберем объекты, которые необходимо выровнять. Щелкните левой кнопкой мыши на сплошном диске и, удерживая клавишу Shift, щелкните левой кнопкой мыши отверстие цилиндра. В результате, оба объекта должны быть одновременно выделены и подсвечены (Рисунок 31).

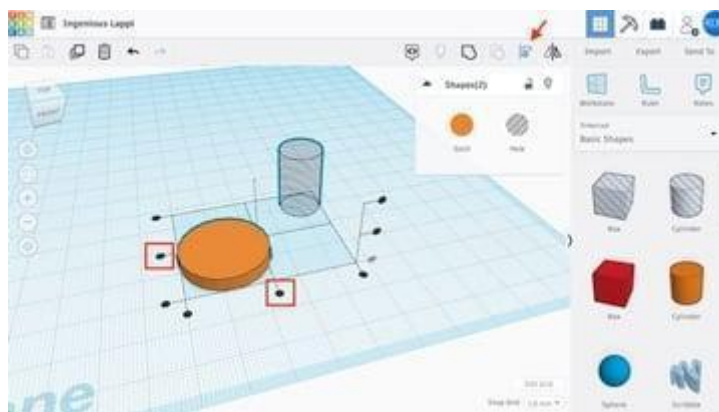


Рисунок 31. Нажмите на черные круглые маркеры, чтобы выровнять два объекта

Нажмите кнопку «Выровнять (Align)» на верхней панели инструментов. Вы заметите, что на рабочей плоскости появятся черные круглые указатели. Нажмите на нижние средние маркеры, как показано ниже, чтобы выровнять объекты по их центральным осям в направлениях X и Y.

### *Объединение фигур*

После того, как фигуры были выровнены, нажмите кнопку «Group (Объединение)» на верхней панели инструментов. В результате, на рабочей площади должна остаться только одна фигура (Рисунок 32).



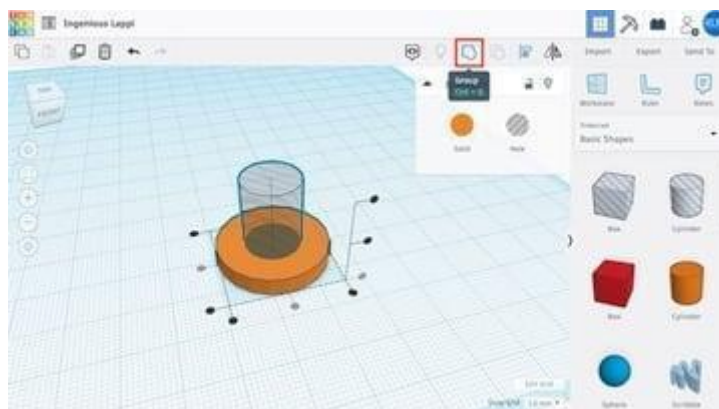


Рисунок 32. Команда «Group (Объединение)» объединяет выбранные фигуры

Этот инструмент, по сути, соединяет две фигуры вместе. В случае, если одна из выбранных фигур является отверстием, форма отверстия высекается из материала (Рисунок 33).

На рисунке ниже представлен результат данного шага - объемное кольцо, по сути своей, полый цилиндр.

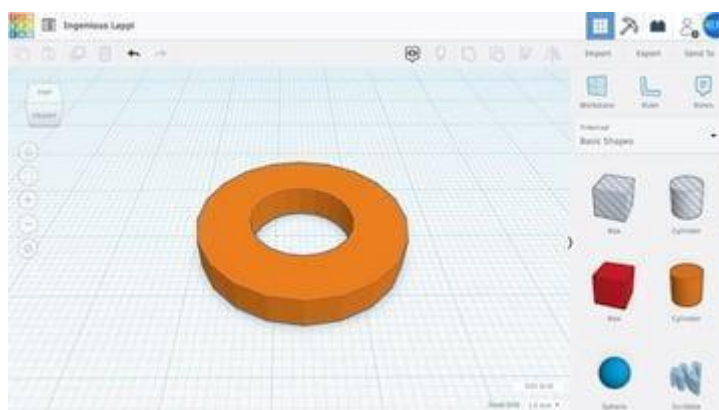


Рисунок 33. Полый цилиндр - кольцо

### *Шаг 3: Проектирование ручки*

Рукоятка, пожалуй, самая простая часть нашего ключа. В основном нам нужен прямоугольник с правильными размерами (Рисунок 34).

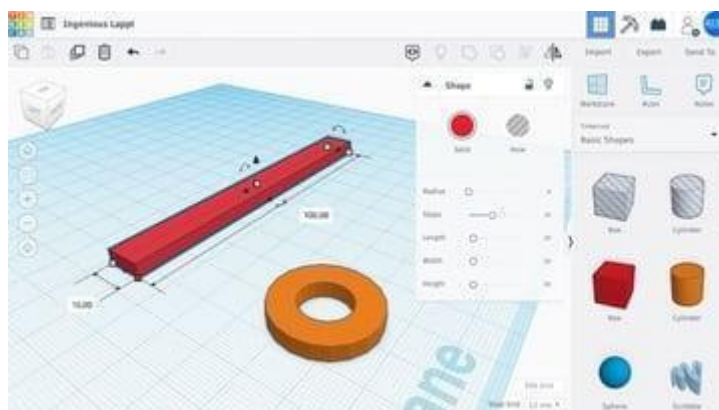


Рисунок 34. Проектирование прямоугольной формы

Начнем с установки прямоугольника с панели основных фигур на рабочую плоскость. Установите длину фигуры на 100 мм и ширину на 10 мм, используя любую из двух процедур, описанных ранее. Далее, установите высоту 4 мм, чтобы она соответствовала кольцу.

#### *Шаг 4: Проектирование рожек ключа*

Как и в случае с кольцом, начнем с диска (Рисунок 35). Проектирование рожек изначально очень схоже с проектированием кольца.

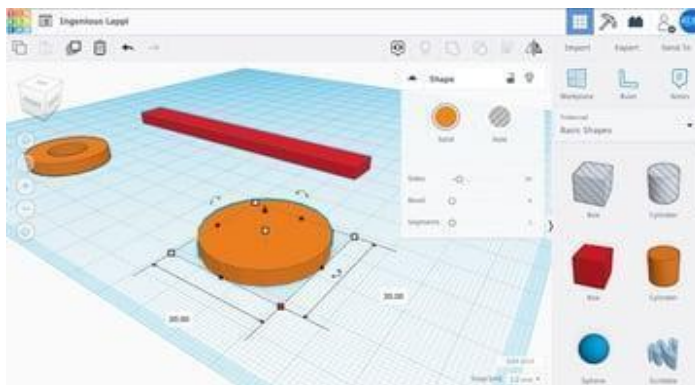


Рисунок 35. Проектирование рожек

Рожки также представляют собой круглый диск, но на этот раз с шестиугольным отверстием, вырезанным под углом. Установите диаметр вашего диска на 30 мм и его высоту на 4 мм.

#### *Многоугольники. Шестиугольник*

Чтобы создать форму, похожую на рожки ключа, можно использовать базовую форму «Многоугольник (Polygon)» (Рисунок 36). Перетащив многоугольник на рабочую плоскость, вы увидите, что по умолчанию, он имеет форму шестиугольника. В случае необходимости, можно изменить количество сторон многоугольника, настроив соответствующий параметр в окне Shape.

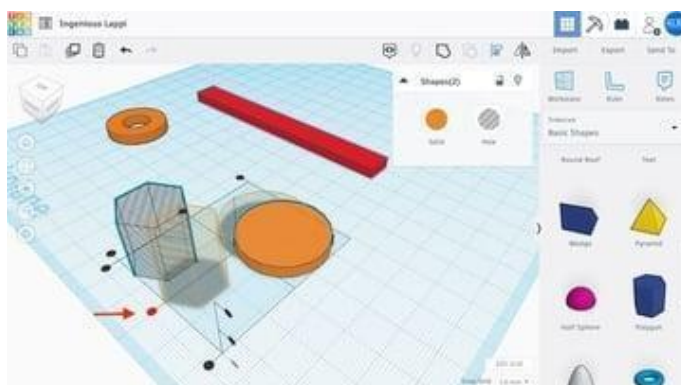


Рисунок 36. Использование многоугольник

Используйте инструмент «Выровнять (Align)» еще раз



Теперь нужно превратить шестиугольную модель в отверстие соответствующей формы.

Снова воспользуемся инструментом выравнивания, чтобы расположить шестиугольник вдоль одной из центральных осей диска. (Другими словами, отцентрируйте его в одном, а не в двух направлениях.) Выберите обе фигуры, нажмите кнопку «Выровнять» и щелкните черную круглую ручку, как показано выше.

Обе фигуры должны быть разделены, поэтому нам нужно поместить шестиугольник внутрь диска. Чтобы сохранить только что сделанное выравнивание, удерживайте Shift и перетащите шестиугольник так, чтобы он частично оказался внутри диска. Шестиугольник должен двигаться по прямой линии.

Теперь, когда обе фигуры расположены правильно, можно установить размеры шестиугольника. Данный гаечный ключ имеет размерность в 15 мм. Обратите внимание, что размер остальной части ключа может быть неподходящим для значений, сильно отличных от 15 мм. Теперь, выберите шестигранное отверстие и диск, и нажмите кнопку «Группировать» в верхнем меню. Результат показан на рисунке 37.

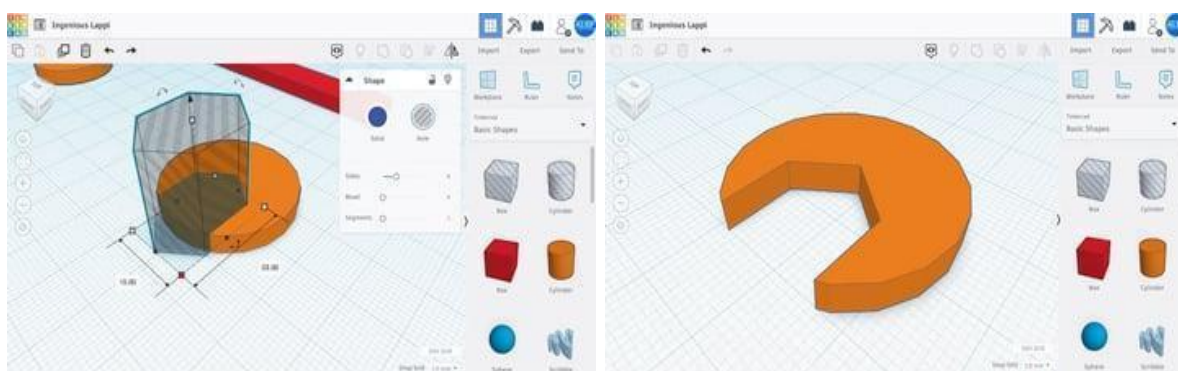


Рисунок 37. Объединяем формы для получения отверстия

В зависимости от типа проектируемого ключа, его ручки могут иметь самую разнообразную форму. Теперь, когда спроектированы кольцо, ручка и челюсть, необходимо сгруппировать эти элементы..

#### *Шаг 5: Объединение отдельных частей*

Объединение элементов начнем с ручки и кольца. Необходимо выравнивать обе фигуры так, как было показано в предыдущем шаге, вдоль длины ручки. Затем, зажав кнопку Shift, переместите кольцо по прямой линии к рукоятке. Убедившись в том, что две фигуры перекрываются друг-другом (Рисунок 38), сгруппируйте их.

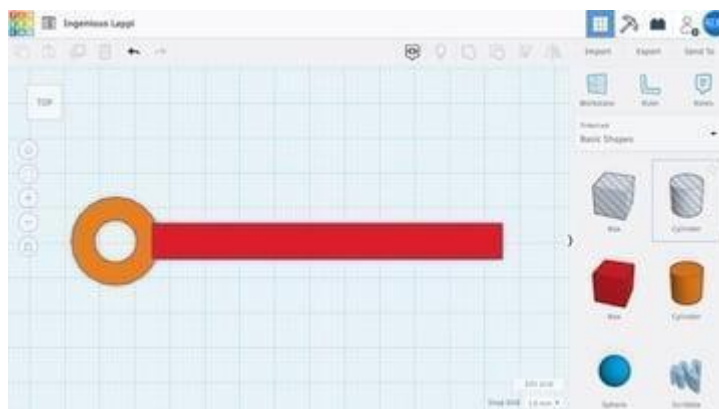


Рисунок 38. Объединение ручки и кольца

После объединения, соединенные части должны слиться в единый цвет. Теперь прикрепим ручки к корпусу гаечного ключа. Повернув ручки в правильном направлении, готовим их к объединению (Рисунок 39).

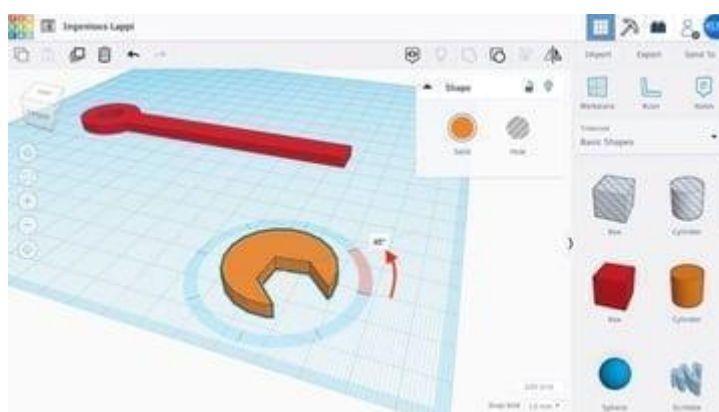


Рисунок 39. Используйте ручку или укажите угол для поворота

Теперь выберите ручки, щелкнув по ней левой кнопкой мыши. Помимо белых ручек, обратите внимание на три изогнутые ручки с двойными стрелками. При наведении указателя мыши на нижнюю (XY) плоскость появится всплывающее значение угла.

Чтобы повернуть челюсть, щелкните и перетащите ручку вокруг фигуры или введите значение вручную. Здесь нам нужен поворот на 45 градусов против часовой стрелки. (Совет: удерживание Shift ограничит количество поворотов, и сделает поворот более удобным)

Все, что осталось сделать на этом шаге, это объединить челюсть. Процедура более или менее такая же, как и при соединении кольца и ручки. Необходимо выровнять обе фигуры, а затем перетащить ручки, зажав клавишу Shift, в сторону гаечного ключа. Расположите их так, чтобы они немного перекрывали друг друга (Рисунок 40).

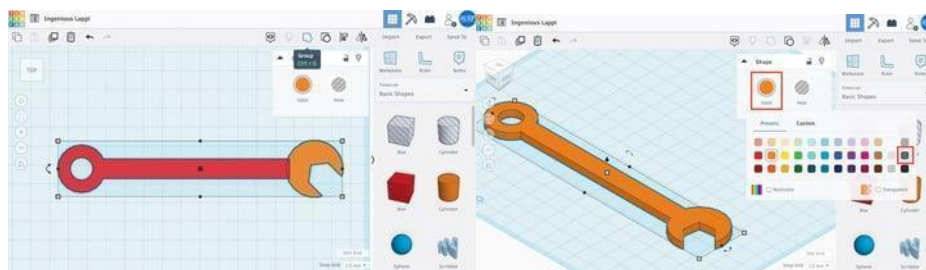


Рисунок 40. Объединение всех частей гаечного ключа

Для придания гаечному ключу естественного цвета, выберите готовый гаечный ключ и нажмите кнопку «Solid» в окне Shape. Будет показан широкий набор цветов. В нашем примере будет выбран темно-серый. Измените цвет модели в окне «Форма».

#### *Добавление текстовых гравировок в модель.*

На этом этапе мы добавим информацию о размере гаечного ключа непосредственно на ручку.

Для этого найдите «Текст» на панели основных фигур и перетащите его на рабочую плоскость. Текст работает как любая другая фигура, и пользователь может изменять его содержимое. Поскольку нам необходимо высечь текст из ручки, сделаем в форме отверстие (Рисунок 41).

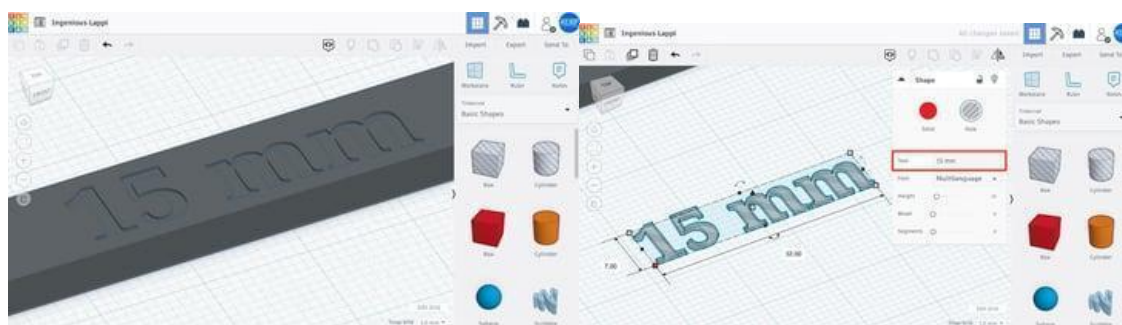


Рисунок 41. Моделирование текста на теле ключа

Работа с текстом аналогична работе с фигурами. Для начала необходимо редактировать текстовое содержимое блока. В окне «Форма» щелкните поле «Текст» и замените его на «15 мм», что соответствует размеру нашего гаечного ключа. Форма изменится автоматически.

Теперь осталось только изменить размер текста и объединить его с помощью гаечного ключа. Отрегулируйте его длину и ширину до 32 и 7 мм соответственно (Рисунок 42).

Чтобы расположить текст над маркером, давайте еще раз воспользуемся инструментом выравнивания. Выберите обе фигуры и нажмите «Выровнять». Поскольку мы хотим, чтобы текст «врезался» в верхнюю часть гаечного ключа, мы выберем верхнюю круглую ручку по вертикали, как показано ниже.

Это сделает верхние поверхности обеих фигур заподлицо. Завершите размещение текста, щелкнув средние круглые маркеры в плоскости XY.

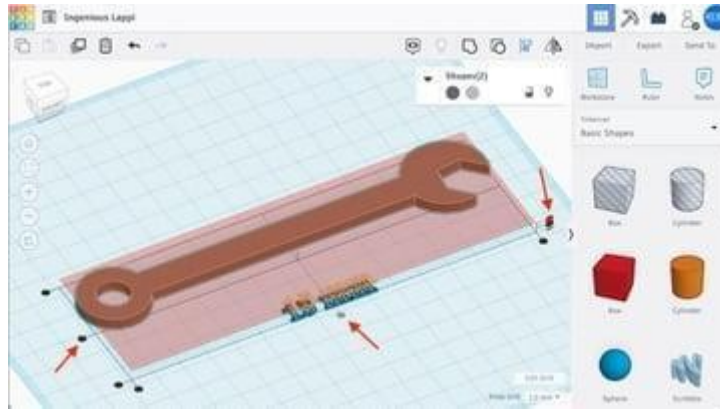


Рисунок 42. Выравнивание высоты текста над уровнем корпуса

Если необходимо расположить текст так, чтобы он выпирал наружу из ручки, вам нужно, чтобы нижняя часть текста была на одном уровне с верхней частью ручки. Ниже представлен один из способов выравнивания объектов таким образом:

1. Временно переместите активную зону рабочей плоскости, нажав «Рабочая плоскость» (или «W» на клавиатуре) и щелкнув соответствующую поверхность. В данном случае это будет верхняя часть ключа.
2. Выберите другой объект — в нашем случае текст — и нажмите «D». Это положит его на временную рабочую плоскость.
3. Отмените временную рабочую плоскость, нажав «Рабочая плоскость» и щелкнув вдали от любых объектов. Все объекты сохранят свои позиции.

*Дополнительно: 3D-печать полученной модели*

Нажмите «Экспорт» и выберите формат файла STL [5] или OBJ[6]. Появится окно загрузки моделей для печати (Рисунок 43))



Рисунок 43. Экспорт объекта в формат STL или OBJ

Выберите тип материала [7] и установите высокий процент заполнения [8]. Пример результата показан на рисунке 44.





Рисунок 44. Готовый гаечный ключ

## 2.2. Применение на практике программы 3DS Max

3DS Max— полнофункциональная программная система для создания и редактирования трёхмерной графики и анимации [9] [10]. Содержит множество современных средств для художников и специалистов в области мультимедиа, а также позволяет создать анимационный ролик или статическое изображение [11].

Из основных направлений использования 3DS Max можно выделить следующие:

- визуализация архитектурных проектов и дизайнов интерьеров;
- компьютерная анимация;
- компьютерные игры;
- спецэффекты.

Богатый набор инструментов дает возможность реализовать в программе любую задумку, относительно несложна в освоении, что делает 3DS Max популярной среди пользователей. Учащиеся могут разрабатывать 3D детали, печатать, тестировать и оценивать их. Возможность материализации выполненных на компьютере объектов с помощью 3D принтера вызывает особый интерес к 3D моделированию у учащихся. Это даёт толчок к познавательной деятельности и способствует развитию технического творчества школьников [12].

В результате работы с программой 3DS Max, возможно выполнение анимации или статического изображения с высокой реалистичностью. Для получения 2D - изображение трехмерного объекта, необходимо выполнить в программе его объемную модель.

Рабочее пространство, в котором работает пользователь 3DS Max, называется *трехмерной сценой*. Создание трехмерных объектов в программе моделируется на основе простейших примитивов - куба, сферы, тора и др.

Для получения, 2D-изображения объекта, спроектированного в 3DS Max, трехмерную сцену, которая содержит созданные объекты, необходимо визуализировать. В окне проекции трехмерная сцена визуализируется

напрямую из заданного окна проекции, либо же через объектив виртуальной камеры. Виртуальная камера - это вспомогательный объект, который обозначает в сцене угол зрения, под которым возможно произвести визуализацию проекта [13].

Реалистичность и качество полученного в результате визуализации изображения во многом зависит от освещения сцены. Программное обеспечение 3DS Max позволяет устанавливать освещение в различных точках трехмерной сцены, с помощью виртуальных источников света. Источники света такие же вспомогательные объекты, как виртуальные камеры. С ними возможно выполнять различные операции такие как: анимировать, менять положение в пространстве, изменять цвет и яркость освещения. Одной из важных деталей являются отбрасываемые тени, благодаря им источники света придают сцене большую реалистичность [14].

### 3DS Max: Элементы интерфейса

Для начала работы в программном обеспечении необходимо выполнить запуск программы 3DS Max, далее на экране монитора ПК появляется рабочее пространство - графический пользовательский интерфейс (Рисунок 45). В рабочем пространстве находятся инструменты управления, навигации, настройки пользовательской рабочей среды. Несмотря на то что данное программное обеспечение имеет сложную структуру, пользовательский интерфейс расположен в определенном логическом порядке, упрощающем дальнейшее применение инструментов.

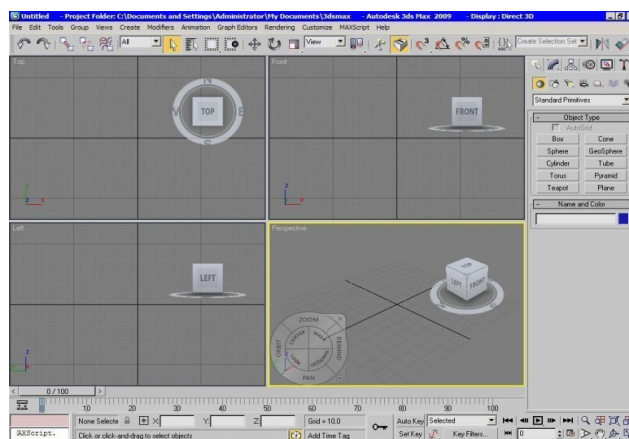


Рисунок 45. Интерфейс программы

В окне проекций отображается сцена, активная в текущий момент. По умолчанию, рабочая сцена в 3DS Max отображается в четырех окнах проекций - вид слева, сверху, спереди и в перспективе (рис. 3.1). В трех окнах объекты показаны схематически, а в окне перспективы - в трехмерном представлении. Одно из окон проекции, в котором осуществляется работа, выделяется желтым цветом и называется *активным*. Вид из каждого окна проекции можно изменять [15]. Изменяя вид в активном окне проекции, необходимо щелкнуть на названии необходимого окна правой кнопкой мыши и выбрать отображение

сцены в меню Views (Вид). Местоположение видовых окон меняются в диалоговом окне Viewport Configuration (Конфигурация окон) [16].

В правом нижнем углу рабочего пространства программы размещены кнопки управления окнами проекций. С их помощью регулируются параметры изображения сцены в активном окне путем смещения точки ее наблюдения (Рисунок 6).

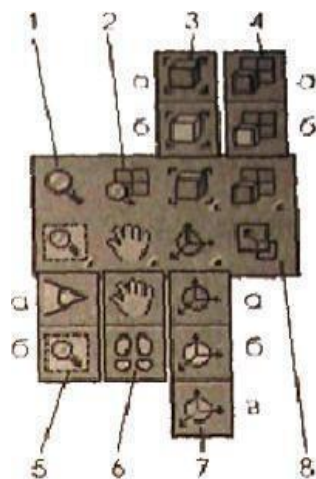


Рисунок 46. Кнопки управления

1) Zoom (Масштаб) — подключение режима изменения масштаба с помощью мыши, изображения, находящегося в активном окне проекции;

Zoom All (Масштаб) — подключение режима изменения масштаба изображений во всех окнах проекций;

Zoom Extents (Содержимое целиком в окне) — автоматическое изменение масштаба активного окна проекции для полного отображения в нем всех объектов сцены;

2) Zoom Extents Selected (Выделенное целиком в окне) — кнопка автоматического изменения масштаба активного окна для полного отображения в нем выделенных объектов сцены;

3) Zoom All Extents (Содержимое целиком во всех окнах) — масштабирование всех окон проекций для полного отображения в них всех объектов сцены;

Field-of-View (Поле зрения) — регулирование мышью поля зрения в активном окне центральной проекции;

4) Zoom Region (Масштаб области) — режим отображения с максимальным масштабом заданной прямоугольной области выбранного окна ортогографической проекции;

5) Pan View (Прокрутка вида) — подключение режима перемещения мышью содержимого активного окна проекции в его плоскости;

6) Walk Through (Проход сквозь) — подключение режима перемещения в пространстве в окне центральной проекции с помощью клавиш-стрелок при нажатой кнопке мыши;

7) Arc Rotate (Повернуть) — подключение режима поворота в пространстве плоскости проекции активного окна относительно центра текущей системы координат, для чего используется трекбол вращения (окружность желтого цвета с четырьмя маркерами)

8) Arc Rotate Selected (Повернуть относительно выделенного) — подключение режима поворота в пространстве выбранной плоскости активной проекции видового окна, относительно центра выделенной группы объектов.

Атрибутами, составляющие интерфейс 3DS Max, являются:

- Кнопка – элемент управления, вызов команды или диалога; Флажок – управляет в форма квадратной выемки;
- Переключатель – осуществляет управление в форме круглой рельефной выемки, состоит из двух и более пунктов, взаимоисключающих друг друга, выбранный отмечается точкой внутри;
- Строка – используется для ввода символьных данных;
- Список – несколько строк, объединенных по смысловому признаку; Группа параметров – логически объединенные элементы управления, расположенные в меню панели и в рельефной рамке;
- Кнопка-флажок – рельефная кнопка, при нажатии на которую происходит переход в другой режим работы, который показывается на кнопке с подсветкой. Для отмены действия нажатой кнопки-флажка необходимо повторно выполнить левый щелчок на ней;
- Модальная кнопка – аналогичный кнопке-флажку элемент управления, для отключения действия которого необходимо выбрать другую модальную кнопку;
- Список кнопок – кнопка, при нажатии и удерживании которой происходит выпадение панели с набором других кнопок, образующих список. Выбор любого из элементов списка необходимо производить, перемещая мышью и отпуская ее нажатую клавишу на нужной кнопке списка;
- Свиток – раскрывающаяся панель с набором элементов управления, объединенных в группы и расположенных одиночно. Свиток в свернутом состоянии выглядит как узкая рельефная горизонтальная кнопка со значком «+» в ее начале.
- Панель команд – совокупность наборов инструментов создания и редактирования объектов, настроек параметров анимации и служебных средств среды 3DS Max. Эта панель представлена в виде закладок по видам действий. Она содержит шесть вкладок: Создание, Изменение, Иерархия, Движение, Отображение и Утилиты.

В нижней части окна расположена шкала анимации, под ней – координаты преобразований, строка состояния, а также кнопки управления анимацией и видом в окнах проекций [17].

Ниже описаны важные настройки интерфейса 3DS Max, включающие в себя выбор единиц измерения, установку режимов Привязок и настройку параметров Сетки Координат.



Для выбора Единиц Измерения следует вызвать пункт Units Setup (настройка единиц измерения) из меню Customize (настроить). В появившемся окне диалога есть набор переключателей для выбора одной из систем: Metric (Метрические), US Standard (Американские), Custom (Пользовательские) и используемые по умолчанию Generic Units (Общие) [18].

Рабочее поле видовых окон для удобства ориентации и отсчетов размеров при создании и редактировании объектов может быть размечена Сеткой Координат – взаимно перпендикулярными служебными линиями. Линии сетки, которые подразделяются на основные и вспомогательные, отстоят друг от друга на расстоянии шага сетки, задаваемого на вкладке Home Grid (Сетка) диалогового окна Grid Snap Setting (применение сетки и привязок), вызывается через пункт меню Customize (Настроить) [19].

Главным назначением сетки состоит в использовании ее линий и их пересечений для применения режима Привязки.

Snaps (Привязки) - это индивидуальный режим создания и редактирования сцены; при данном режиме произвольное перемещение курсора заменяется привязкой - определенные элементы сцены могут быть использованы как точные ориентиры для указания.

### *Примитивы 3DS Max*

Трехмерная графика рассматривает любой предмет как набор простейших объектов. Самые простые объекты в 3DS Max создаются с помощью панели инструментов, при помощи нескольких щелчков мыши.

Для этого удобно пользоваться Командной панелью. Активные кнопки для создания простых объектов находятся на вкладке Create (Создать) [20].

3DS Max содержит множество необходимых объектов, которые сгруппированы по категориям. Примитивы расположены в категории Geometry (Геометрия). Одна категория может содержать несколько групп объектов, переключение которых можно выполнить с помощью списка. По умолчанию первой открывается группа простых примитивов (Standard Primitives) - геометрические объекты самой простой формы. Раскрыв список свиток, вы увидите другие группы объектов.

Для создания объекта, необходимо нажать кнопку с названием примитива на командной панели, щелкнуть в активном окне проекции и, удерживать нажатой кнопку мыши, перемещать указатель в сторону, при этом объект будет менять свои параметры. Если объект достигнет желаемого размера, отпустите кнопку мыши. Объекты можно создавать и путем ввода параметров (размеров) объекта в свитке Keyboardy Entry (Ввод с клавиатуры). Для этого после нажатия кнопки с названием примитива перейдите в появившийся ниже свиток, введите параметры объекта, координаты точки расположения и нажмите кнопку Create (Создать)(Рисунок 47).



Рисунок 47. Создание примитивов

Существует еще около десяти разных режимов отображения объектов, объект может отображаться как в виде полигональной сетки, так и с наложением цветной заливки с отбрасыванием теней [21].

Переключение между этими режимами осуществляется с помощью контекстного меню окна проекции. Вызвать его можно щелкнув на названии окна правой кнопкой мыши (Рисунок 48). Напротив, названия того варианта отображения, который используется в данный момент, установлен флажок. На верхнем уровне меню есть все доступные режимы отображения, так как некоторые находятся в меню Other (Другие).

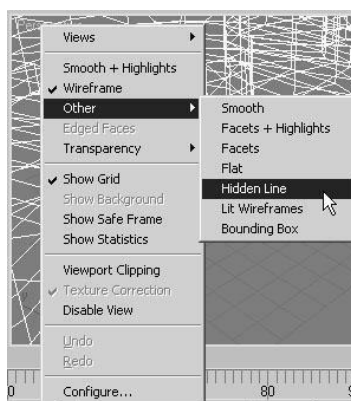


Рисунок 48. Контекстное меню окна проекции

По умолчанию стоит режим Smooth + Highlights (Сглаживание и блики), но чаще используются варианты отображения Wireframe (Каркас),

Hidden Line (Скрытые линии), Bounding Box (Габаритный контейнер), а также вспомогательный режим Edged Faces (Контурные ребер) [22].

Режимы Wireframe (Каркас), Hidden Line (Скрытые линии) и Bounding Box (Габаритный контейнер) применяются приоритетно, когда рабочая сцена содержит большое количество объектов, и воспроизведение их в окнах проекций занимает много времени. Вспомогательными режимами являются Edged Faces (Контурные ребер) часто используется вместе с режимом Smooth + Highlights (Сглаживание и блики) и с некоторыми другими. При его активности, отображаются контуры ребер, благодаря этому можно видеть полигональную структуру модели оценить качество построения [23].

Вид отображения объектов возможно выбирать отдельно для каждого окна проекции.

*Настройка объектов.* Каждый отдельный объект имеет свои размерные параметры, которые определяют форму. Данные параметры находятся в нижней части командной панели на вкладке Create (Рисунок 49). Сняв выделение с объекта, то параметры отображаться не будут. Всегда сохраняется возможность из изменения, для этого следует выбрать объект и перейти на вкладку Modify (Изменение) командной панели.

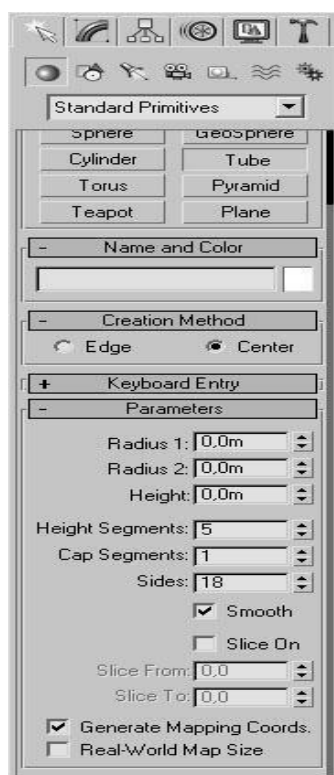


Рисунок 49. Командная панель

*Настройки объекта на вкладке Create.* Прimitives имеют свои режимные настройки. Например, основные параметры цилиндра - это высота и радиус, примитив Tube (Трубка) имеет два радиуса и высоту и т. д.

Почти все примитивы имеют параметры Segments (Количество сегментов) и Sides (Количество сторон). Они отвечают за детальность построения модели. Чем больше их количество, тем более точной и качественной будет модель, но и тем больше аппаратных ресурсов ПК потребуется на работу с ней. При создании объекта в окне проекции, ему автоматически присваивается имя. Если в сцене содержится много объектов, есть возможность редактирования имен, например, Пол, Окно, Перегородка и т. д. Это помогает проще ориентироваться в рабочем пространстве [24].

При выделении объекта, в верхней части панели Modify (Изменение) отображается его имя. Чтобы переименовать объект, нужно установить курсор в поле, где написано название, и введите новое имя (Рисунок 50).

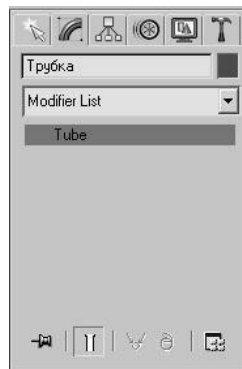


Рисунок 50. Отображение имени объекта на вкладке Modify (Изменение)

При создании каждого отдельного объекта 3DS Max автоматически выбирает для него цвет. По умолчанию используются все цвета используются случайно, белый не используется так как на объекте белого цвета плохо заметно выделение.

Цвет объекта можно изменить. Для этого нужно выбрать поле с изображением цвета, выбрать нужный цвет в появившемся окне Object Color (Цвет объекта) (Рисунок 51), после чего подтвердить выбор, нажав на кнопку ОК [25].

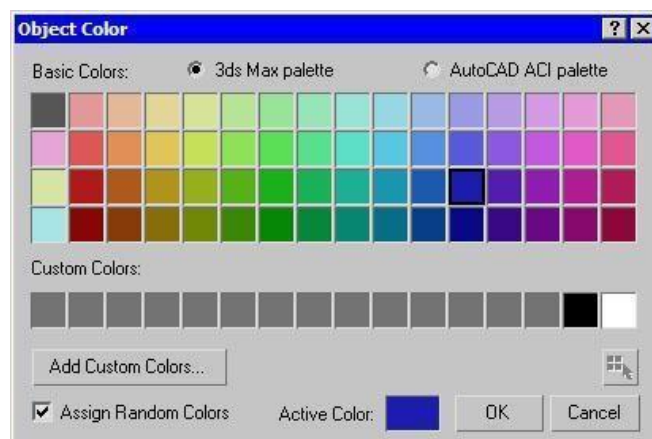


Рисунок 51. Окно Object Color

Объекты 3DS Max можно найти на вкладке Create командной панели. Объекты разделены по категориям, а категорий разделены по группам. Всего доступно семь категорий: Geometry (Геометрия), Shapes (Формы), Lights (Источники света), Cameras (Камеры), Helpers (Вспомогательные объекты), Space Warps (Объемные деформации), Systems (Дополнительные инструменты).

Одни объекты служат для моделирования трехмерных сцен, а другие используются в качестве вспомогательных инструментов. Объекты первой категории представляют собой простейшие трехмерные геометрические фигуры: Sphere (Сфера), Box (Параллелепипед), Cone (Конус), Cylinder (Цилиндр), Torus (Тор), Plane (Плоскость) и др. Прimitives делятся на две группы: Standard Primitives (Простые примитивы) и Extended Primitives (Усложненные примитивы) [26].

### 3DS MAX: Практическое применение примитивов

Рабочее пространство содержит четыре окна проекции (Рисунок 52). Активное окно выделено желтым цветом.

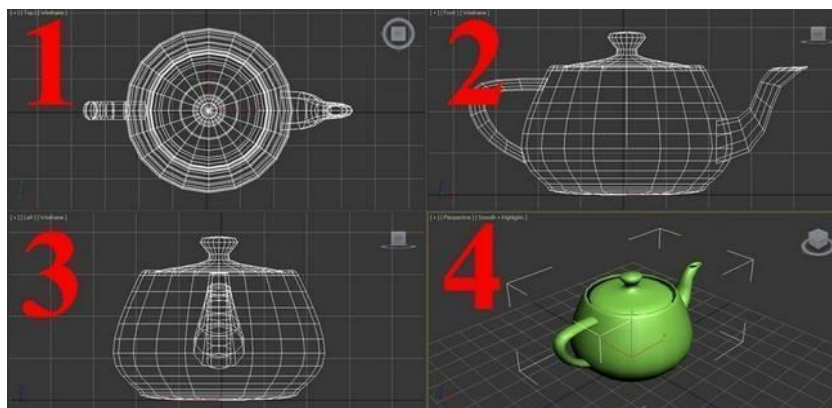


Рисунок 52. Рабочее пространство программы: 1 - Top (вид сверху), 2 - Front (фронтальный вид или спереди), 3-Left (вид слева), 4- Perspective (перспектива).

Вверху экрана находится командная панель, на панели располагаются командные клавиши:

Create ( Создать) —создание различных объектов;

Modify (Изменить) —изменение параметров объекта и применения модификаций;

Hierarchy (Иерархия) —управление связями;

Motion (Движение) — Настройки контроллеров анимации и траекторий движения;

Display (Отображение) — управление отображением объектов сцены в окнах проекций;

Utilities (Утилиты) — содержит вспомогательные программы, которые являются подключаемыми плагинами.

Первая вкладка Create содержит несколько подкладок:

Geometry (Геометрия) — объемные тела (кубы, шары, цилиндры и др.);

Shapes (Формы) —создание линий;

Lights (Источники света) — объекты для освещения сцены;

Cameras (Камеры) - камеры, выставляются в сцене, могут быть анимированы;

Helpers (Вспомогательные объекты) —значительно

упрощает конструирование сложных сцен и настройку анимации;

Space Warps (Объемные деформации) —отвечает за различные виды искажений окружающего пространства;

Systems (Системы) — объединяет объекты.

Сначала создайте несколько примитивных объектов. Для этого на правой панели выберите:

Create - Geometry – Standard Primitives – Box, далее рисуем в окне перспективы бокс.



Рисунок 53. Панель управления

После ввода параметров нажимаем Create и создается бокс с заданными параметрами в указанных координатах [27] (Рисунок 53).

Изменить параметры заданного бокса можно во вкладке Parameters. Также возможно изменить количество сегментов по ширине, высоте и длине (Length Segs, Width Segs, Height Segs).

Создайте фигуры произвольных размеров и измените количество сегментов (Рисунок 54, 55). Если визуально не наблюдаете изменений, попробуйте нажать F4 – эта клавиша включает\отключает отображение ребер во вьюпорте [28].

Создайте следующие объекты: Box, Sphere и Cylinder. Должно получиться как на картинке:

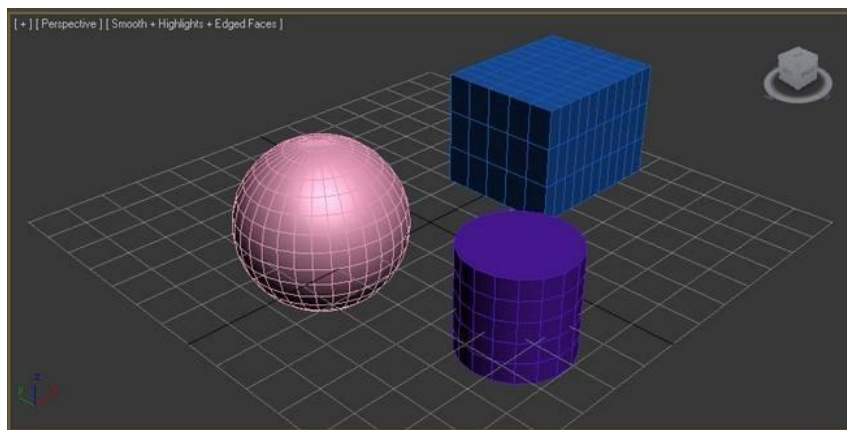




Рисунок 54. Примитивы

В нижнем правом углу располагается меню навигации:  - Zoom (приближение\отдаление)

 - Перемещение (вверх\вниз\вправо\влево)

 - Вращение

В результате работы с примитивами получаем следующие объекты:



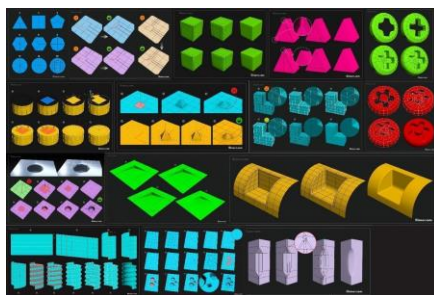


Рисунок 55. Редактированные примитивы

*Урок 2. Создание объектов со сложной полигональностью с применением сплайнов (Рисунок 56).*

В ходе этого урока вы познакомитесь с понятием- сплайн, для чего они применяются и как с ними работать, а также с редактированием сложной полигональной системой. Сплайн – это редактируемая линия. Сплайны не отображаются при рендере, служат вспомогательными средствами. Создать сплайн возможно, перейдя во вкладку Create - Shapes – Splines [29].

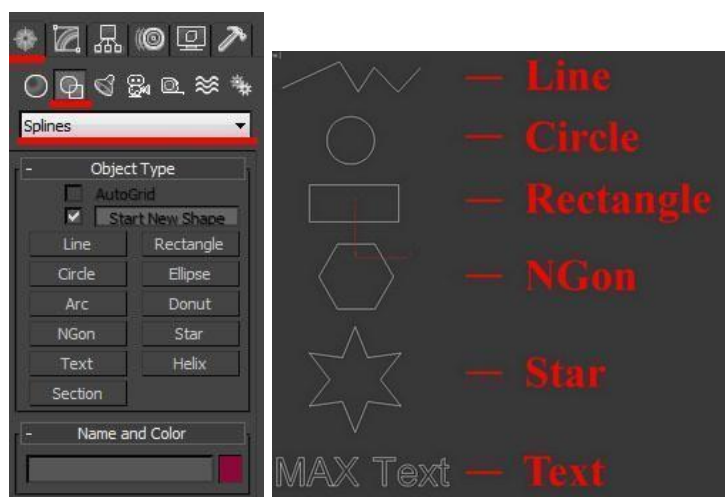


Рисунок 56. Вкладка Create - Shapes - Splines. Виды сплайнов

При создании любого типа сплайнов, после изменения параметров и для дальнейшего редактирования, необходимо выбрать его правой кнопкой и нажать Convert To - Convert To Editable Spline, или выбрать модификатор Edit Spline из списка модификаторов [30].

Для удобства редактирования сплайны состоят из точек (vertex) и сегментов (segment) (Рисунок 57):

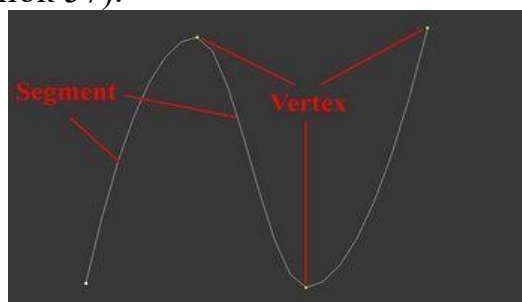


Рисунок 57. Конфигурация сплайнов



Точки бывают четырех типов (Рисунок 58):

1. Corner - точка образует ломаную линию;
2. Smooth - автоматически сглаживает линию, образуя кривую;
3. Bezier - образует сглаженную линию, формой изгиба возможно манипулировать с помощью специальных маркеров;
4. Bezier corner - маркеры несимметричностей

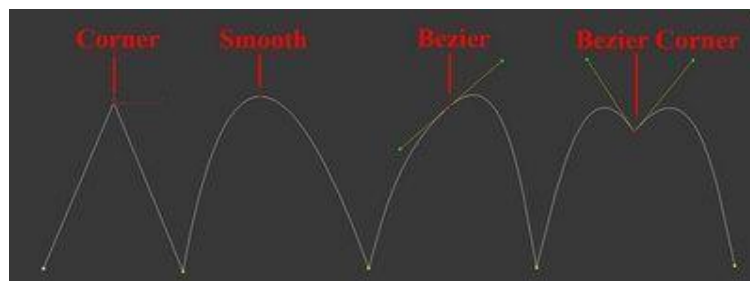


Рисунок 58. Виды точек

Для изменения типа точки нужно выделить одну или несколько точек и нажать правую кнопку мыши. В контекстном меню выбрать нужный тип. В разделе Modify сплайны имеют несколько закладок: Rendering, Interpolation, Selection, Soft Selection и Geometry [31].

Редактировать сплайны так же возможно во вкладке Rendering:

- Enable in rendering - включает отображение сплайна при рендере
- Enable in Viewport - включает отображение сплайна в окнах вьюпорта
- Radial - имеет круг в сечении
- Thickness - ширина или диаметр круга в сечении. Sides – кол-во сторон круга
- Angle - угол поворота сечения.
- Rectangular - сплайн будет иметь прямоугольник в сечении Length - длина прямоугольника в сечении
- Width - ширина прямоугольника в сечении
- Aspect - соотношение длины и ширины друг к другу Auto Smooth - автоматическое сглаживание
- Вкладка Interpolation:
- Steps - количество сегментов между каждыми двумя точками сплайна
- Optimize - убирает сегменты, которые не влияют на форму сплайна
- Adaptive - подбирает количества сегментов для идеально гладкой формы
- Вкладка Geometry:
- New vertex type - выбор тип вновь созданных точек. Create line - создать дополнительную линию в сплайне Attach - присоединить линию к сплайну
- Refine - добавить точку в любое место сплайна
- Weld - слить точки в одну
- Connect - соединить две точки линии не сливая их Insert - продолжить линию
- Fillet - скругление углов Chamfer - создание фаски

- Divide - разделяет выбранный сегмент на указанное количество точек.

**Применение модификатора Shell на примере создания объемного текста.**  
Создаем сплайн **Text** (Рисунок 59):

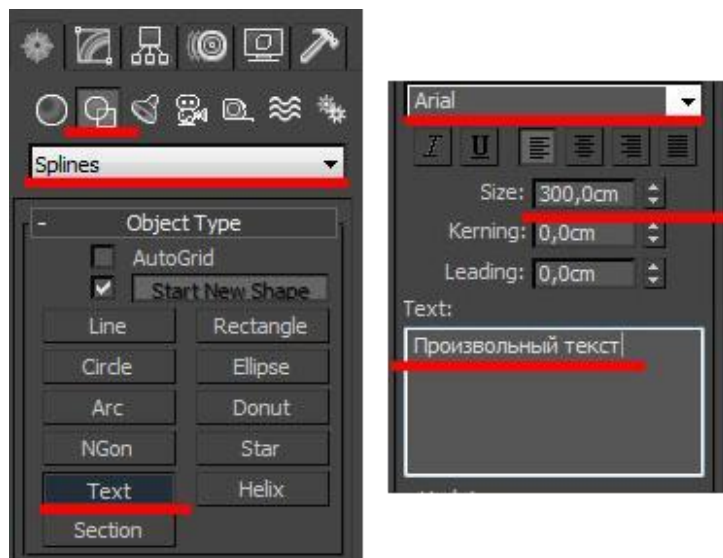


Рисунок 59. Создание текста

В поле **Text** меняем стандартный текст на необходимый и изменяем его размер (Size) (Рисунок 60). Выбираем и стека модификаторов Shell. Далее задаем нужную степень выдавливания (Inner Amount или Outer Amount) [32].

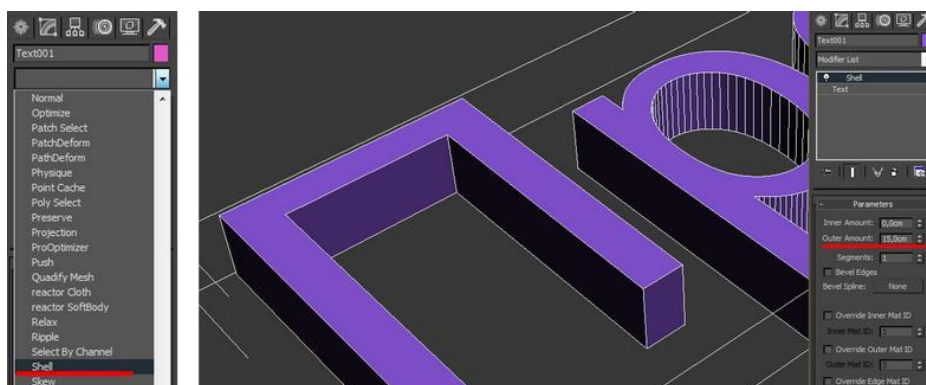


Рисунок 60. Применение модификатора

Модификатор Shell (Рисунок 61) возможно применять только к замкнутым сплайнам (не имеющим пробелы)

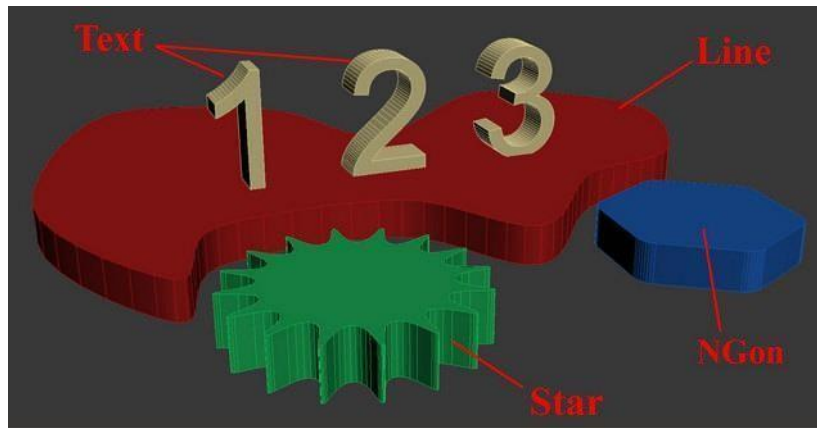


Рисунок 61. Пример использования модификатора Shell на сплайны Text, Line, NGon и Star:

### Применение модификатора Sweep.

Создаем любую линию, например, **Line** с прямыми углами (создается с нажатой клавишей Shift).

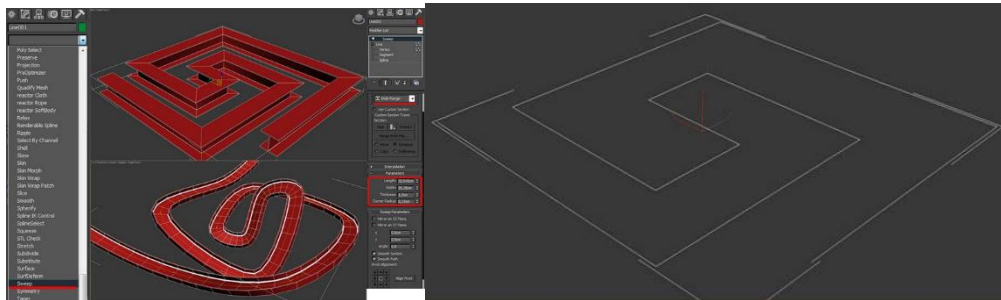


Рисунок 62. Создание линии

Из списка модификаторов выбираем Sweep.

### Применение модификатора Lathe.

Модификатор **Lathe** используется для создания объектов, которые имеют симметрию при вращении (бокалы, стаканы, бутылки, колонны). Для этого необходимо нарисовать вертикальную линию в проекции Front и выбрать из списка модификаторов Lathe:

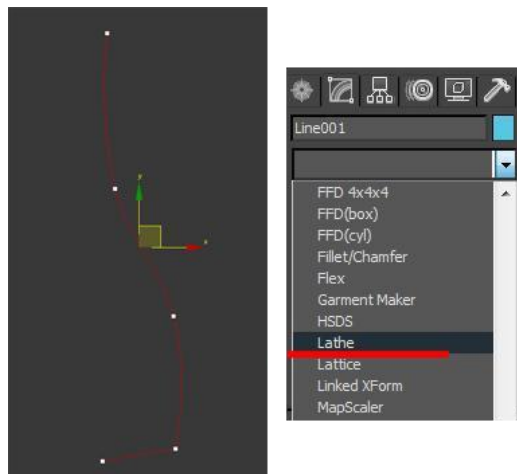


Рисунок 63. Создание профиля будущей модели

Необходимо переместить ось симметрии, для этого нажимаем на плюс (+), расположенный справа от названия модификатора, и выбираем Axis. Перемещаем Axis по оси X:

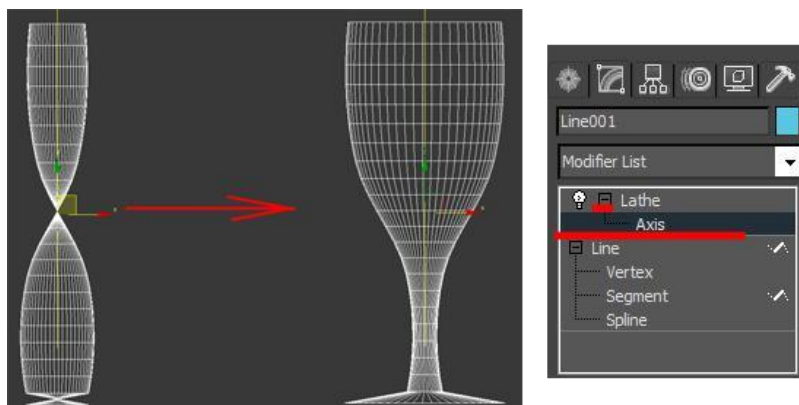


Рисунок 64. Применение модификатора Sweep

Далее необходимо проверить включена ли кнопка Show end result on/off toggle, которая находится снизу от списка модификаторов (Рисунок 65). Ее нужно включить для того, чтобы видеть эффект всех модификаторов, назначенных объекту:

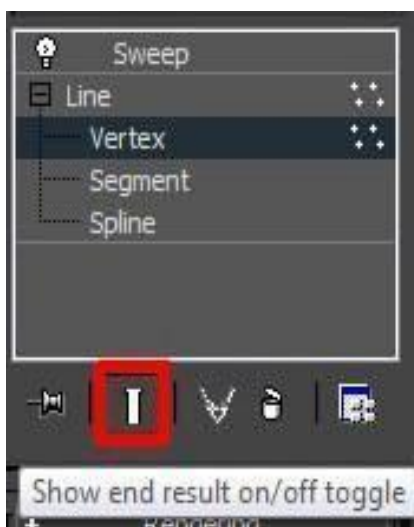


Рисунок 65. Окно редактирования сплайна

Далее переходим к редактированию вершин. Делаем активной кнопку Vertex, добиваемся желаемого результата. Также возможно добавлять\удалять точки, менять их тип [33].

При знании вышеизложенного материала возможно создание следующих моделей, которые были предложены ученикам на контрольном уроке №2:

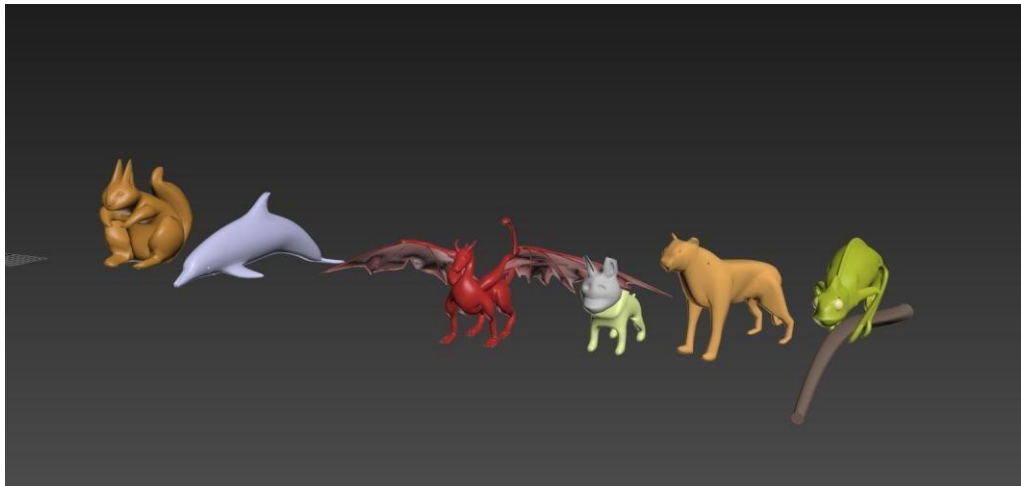


Рисунок 66. Создание моделей животных

### Изучение основных модификаторов в 3D MAX

1. Bend. Модификатор изгибает поверхность на заданный угол. Можно перемещать центр изгиба (Рисунок 67).

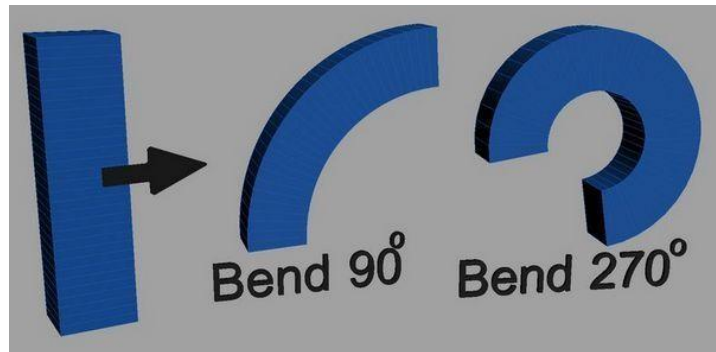


Рисунок 67. Применение модификатора Bend

2. Shell. Данный модификатор выдавливает, придает объем, можно использовать на сплайны и любые другие объекты (Рисунок 68).

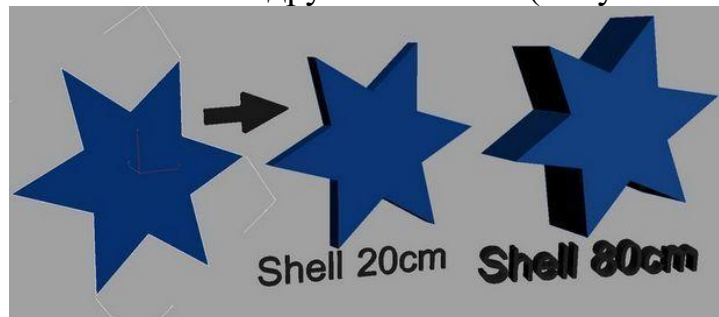


Рисунок 68. Применение модификатора Shell

3. Twist. Скручивает поверхность по заданной оси координат, возможно изменение центра скручивания.

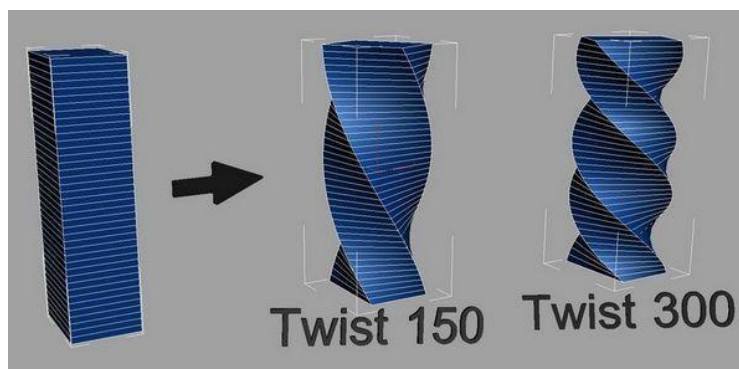


Рисунок 69. Применение модификатора Twist

4. Symmetry. Создает плоскость отражения. Плоскость передвигается в любом направлении (Рисунок 70).

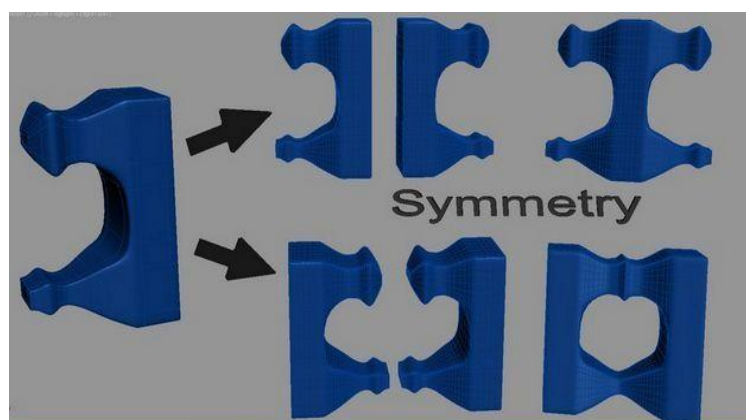


Рисунок 70. Применение модификатора Symmetry 5.TurboSmooth.

Сглаживание поверхности с заданной силой (Рисунок 71).

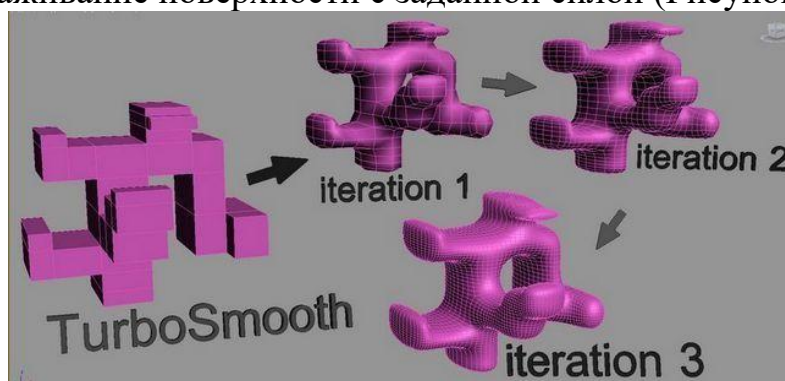


Рисунок 71. Применение модификатора TurboSmooth

6.Relax. Работает аналогично TurboSmooth и Spherify. Сглаживает острые углы (Рисунок 72).



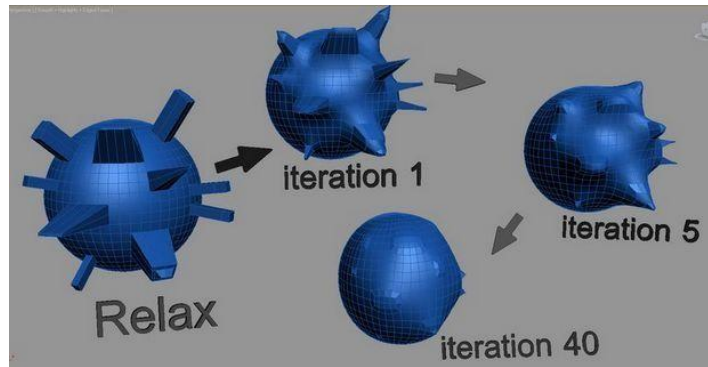


Рисунок 72. Применение модификатора Relax 7.Spherify.

Этот модификатор превращает любой объект в сферу, при достаточном количестве полигонов (Рисунок 73). Должен задаваться определенный процент интенсивности [34].

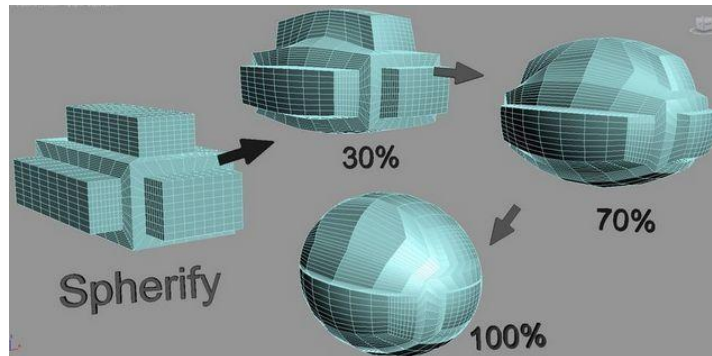


Рисунок 73. Применение модификатора Spherify

8. Noise. Создает шумы или неровности. Задается размер и деформация по осям (Рисунок 74).

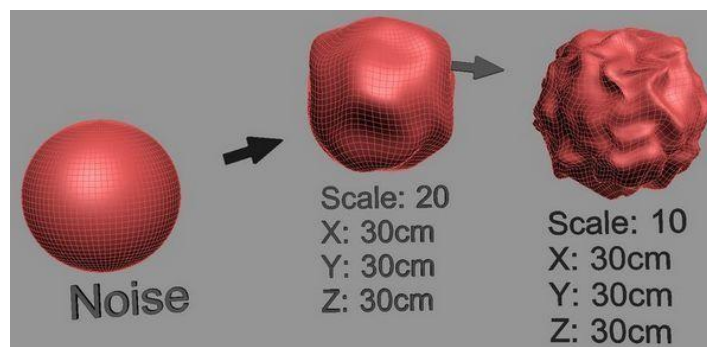


Рисунок 74. Применение модификатора Noise

9. FFD. Создает редактируемую сетку вокруг объекта. Возможно задавать сетку с различным количеством точек (Рисунок 75).



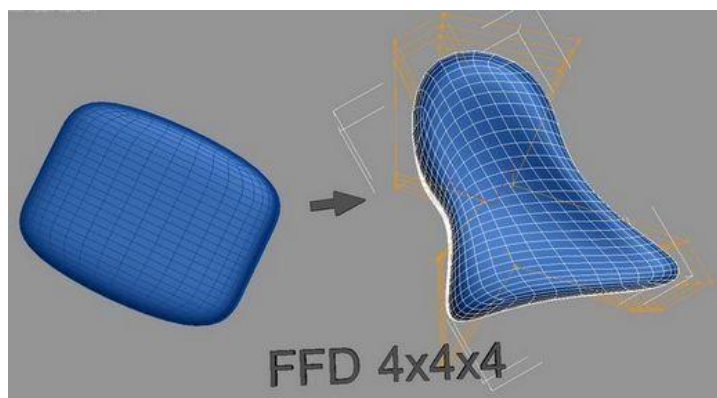


Рисунок 75. Применение модификатора FFD

10. HSDS. Сглаживание по аналогии с TurboSmooth, но с разнообразным количеством функций. Позволяет создавать слои сглаживания для отдельных полигональных групп.

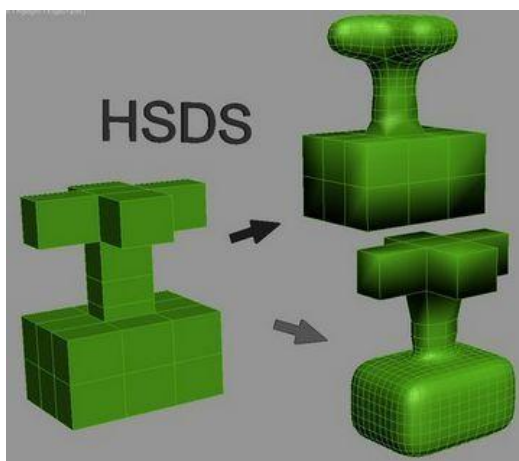


Рисунок 76. Применение модификатора HSDS

11. Lattice. Создает сетку из ребер поверхности.

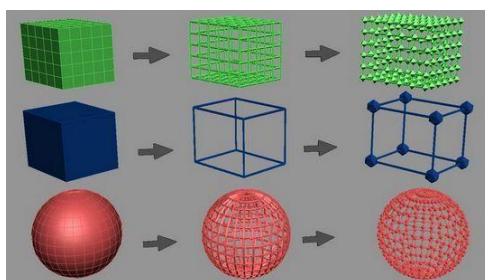


Рисунок 77. Применение модификатора Lattice

12. Lathe. Проворачивает выбранный сплайн вокруг своей оси, образует поверхности с симметрией вращения.

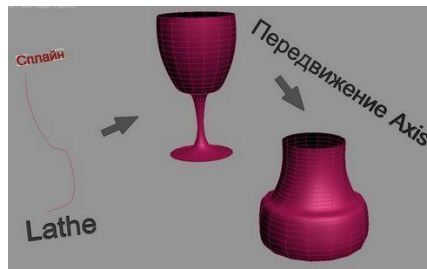


Рисунок 78. Применение модификатора Lathe

13. Sweep. Протягивает одну из предложенных в модификаторе форм вдоль сплайна и образует поверхность, возможно изменять параметры формы и угол сечения.

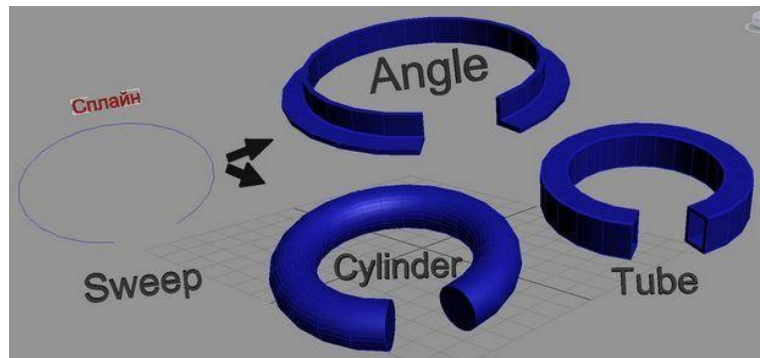


Рисунок 79. Применение модификатора Sweep

14. Melt. Имитирует "плавление", возможность выбрать материал плавления (лед, стекло, желе) и степень "расслабленности"

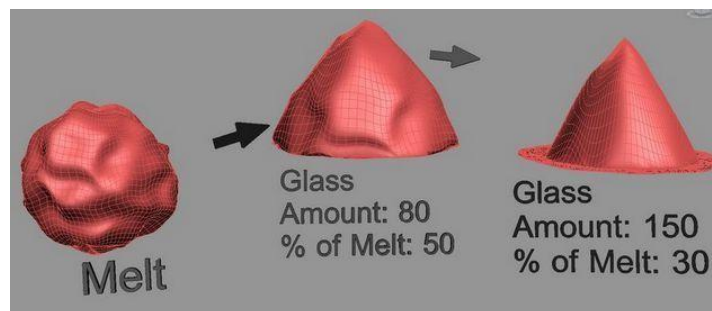


Рисунок 80. Применение модификатора Melt

15. Wave. задается амплитуда "волны" по горизонтали и вертикали. Объект деформируется:

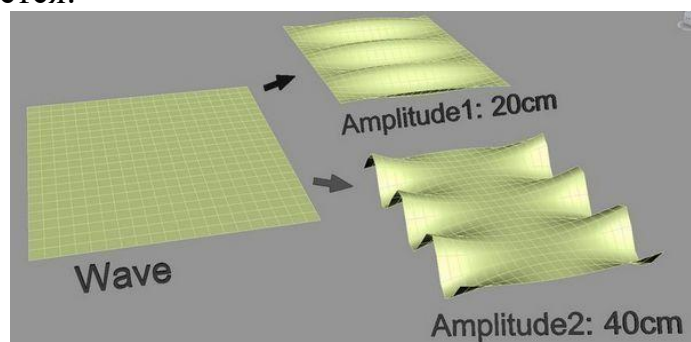


Рисунок 81. Применение модификатора Wave

16. Stretch. Простой модификатор, объект сужается как бы по "талиии" и вытягивается в высоту.

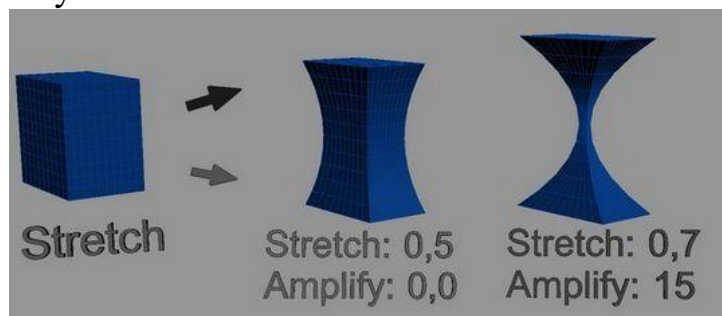


Рисунок 82. Применение модификатора Stretch

17. Taper. Так же простой модификатор, редко его использую, делает собственно следующее:

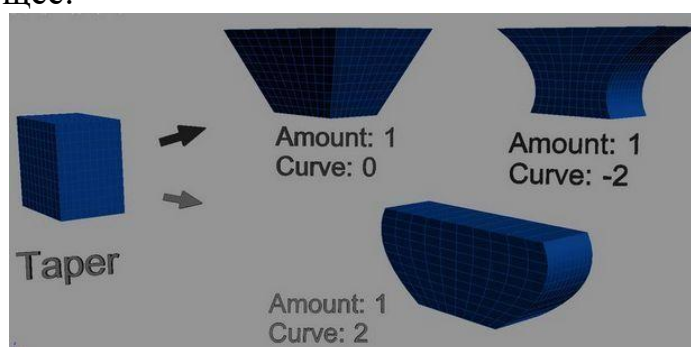


Рисунок 83. Применение модификатора Taper

В результате пройденного материала на контрольном занятии №3, ученикам было предложено выполнение модели по собственному эскизу:

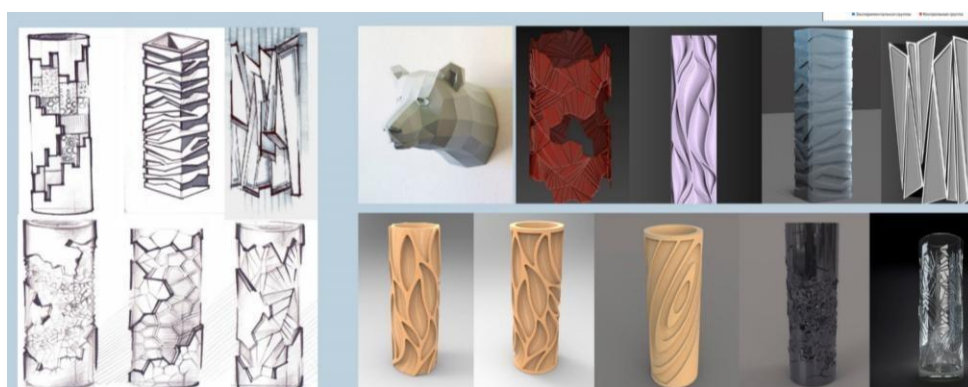


Рисунок 84. Модели, созданные по эскизам учеников

**3DS MAX:** Механизмы дальнейшей обработки для получения готового проекта

Для получения готового проекта, будь то 2D-графика или трехмерная модель, существуют различные операции и программы для последующей обработки полученных "сырых" макетов.

Программа 3DS MAX предоставляет возможность 3D-визуализации (рендеринг), которая превращает трехмерную модель в изображение высокого

качества и с максимальной реалистичностью. Для того, чтобы объект был максимально реалистичным, необходимо *наложение материалов* на него.

Материалы служат для самых разных целей и могут быть использованы для отображения различных видов поверхностей. Также, материалы позволяют придавать тот или иной возраст объекту. Материалы можно создавать в соответствии со стилем, выбранным для конкретных изображений или фрагментов анимации. При создании материала необходимо принимать во внимание его внешний вид при разных условиях освещения, а также характер взаимодействия света с материалом. Каждый материал имеет большое количество свойств (рельеф поверхности, зеркальность, рисунок, размер блика, фактура, текстура и т. д.). Для описания характеристик материала используются числовые значения параметров [35].

Программа 3DS Max содержит специальный модуль для работы с материалами - Material Editor. С помощью данного модуля можно управлять различными свойствами объектов (цвет, фактура, яркость, прозрачность и др.)

Окно Material Editor (Редактор материалов) вызывается при помощи

- а) команды Rendering-Material Editor (Визуализация-Редактор материалов);
- б) горячей клавишей M;
- в) щелчком на кнопке Material Editor, расположенной на основной панели инструментов.

В окне Material Editor отображаются основные типы материалов, которые наиболее часто востребованы в архитектуре и дизайне (Рисунок 85). Каждый материал размещен в отдельной ячейке. Активировав нужную ячейку, можно увидеть настройки материала и при необходимости изменить их [36].

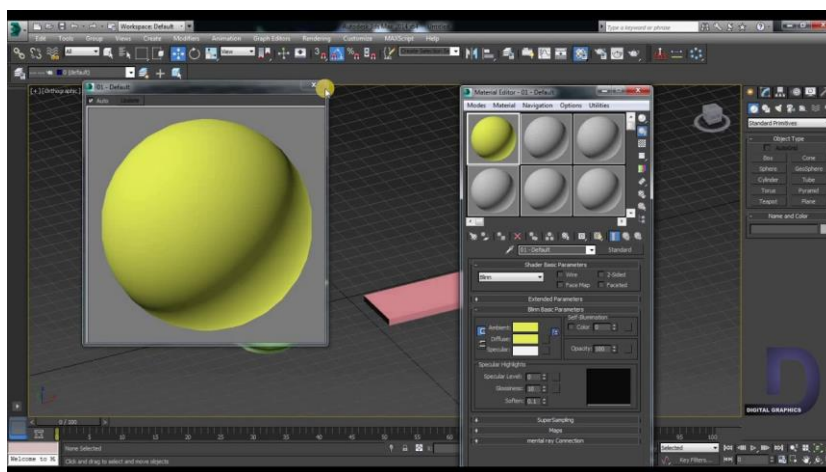


Рисунок 85. Окно активности материалов

При помощи набора материалов, который имеет общее название Pro Materials, можно создать реалистичный металл, керамику, бетон, стекло, пластик, камень и другие типы поверхностей. Каждый из материалов имеет множество достаточно простых настроек. К примеру, настраивая пластик,

возможно выбрать один из вариантов поверхности: глянцевая, блестящая или матовая [37].

Настройки каждого материала содержатся в свитках под ячейками материалов. Выбранная ячейка выделяется цветом. Работа ведется именно с материалом выделенной ячейки, и все параметры, расположенные ниже, относятся к ней.

Внизу под ячейками расположена панель инструментов для редактирования материалов.

Существуют сложные материалы состоящие из двух и более материалов, используется для текстурирования сложных объектов (Рисунок 86).



Рисунок 86. Пример сложного двухкомпонентного материала

По умолчанию к объекту применяется тип материала Standard (Стандартный). Чтобы изменить стандартный материал на необходимый, нужно нажать кнопку Get Material (Установить материал) и выбрать требуемый в окне Material/Map Browser (Окно выбора материалов и карт) (Рисунок 87).

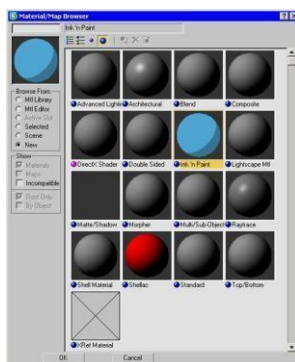


Рисунок 87. Окно выбора материалов и карт

Применить к объекту материал возможно двумя способами:

а) перетащить необходимый материал из окна Material Editor (Редактор материалов) на объект в окне проекции (Рисунок 88);



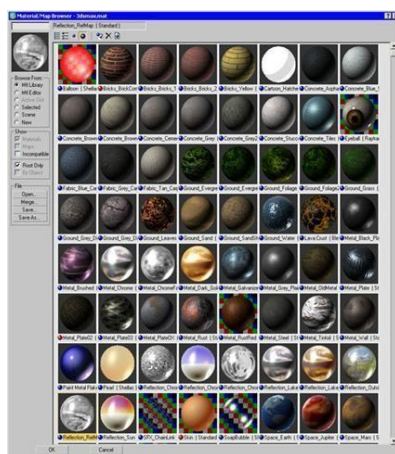


Рисунок 88. Редактор материалов

б) выделить объект или набор объектов в окне проекции; выбрать необходимый материал в окне Material Editor (Редактор материалов) и щелкнуть *Assign Material to Selection* (Назначить материал выделенным объектам) на панели инструментов окна Material Editor (Редактор материалов) [45].

Для определения, применен ли материал к объекту сцены, обратитесь внимание на закладку материала в окне Material Editor (Редактор материалов).

К дальнейшим функциям постобработки можно отнести визуализацию созданного объекта. Необходимые настройки визуализации устанавливаются в окне Render Setup (Настройка визуализации (рендеринга)) (Рисунок 89). Для его вызова необходимо выполнить команду Rendering-Render Setup (Визуализация-Настройка визуализации), нажать кнопку Render Setup (Настройка визуализации) на основной панели инструментов или воспользоваться горячей клавишей F10 [39].

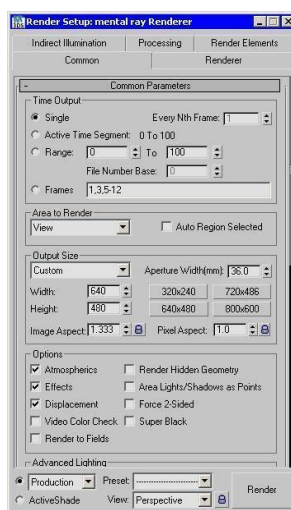


Рисунок 89. Настройка визуализации

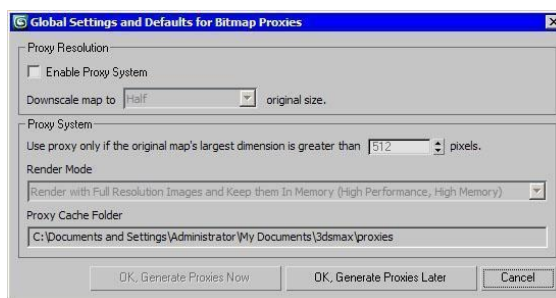


Рисунок 90. Режим визуализации

Параметры режимов визуализации редактируются в списке Render Mode (Режим визуализации). Возможно визуализировать объект, используя следующие режимы:

1. Render with Proxies (High Performance, Low Memory) (Визуализировать, - этот режим необходим для тестовых визуализаций, с возможностью оценки картинке при небольшой затрате времени [40].
2. Render with Full Resolution Images and Keep them In Memory (High Performance, High Memory) - этот режим подходит для использования мощных компьютеров с большим объемом оперативной памяти..
3. Render with Full Resolution Images and Free them from Memory (Low Performance, Low Memory) - визуализирует изображение с высоким качеством текстур, но при этом мощности компьютера недостаточно для постоянного хранения информации в оперативной памяти.

Следить за процессом визуализации можно при помощи окна Virtual Frame Buffer (Виртуальный буфер кадра). В нем активируется изображение сцены по мере того, как она визуализируется (Рисунок 90).

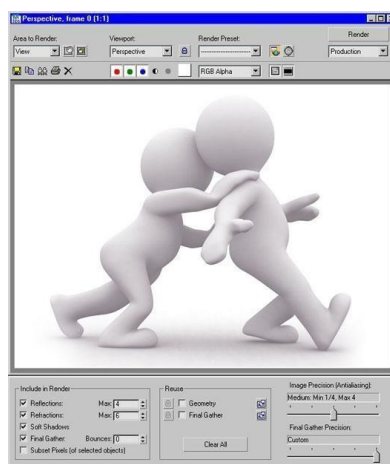


Рисунок 91. Виртуальный буфер кадра



### 2.3. Применение на практике программы Blender

**Blender** – расширенная, бесплатная, небольшая по размеру программа для работы с трёхмерной компьютерной графикой (Рисунок 92). Является одним из наиболее популярных программных обеспечений для 3D-дизайна. Имеет огромное активное сообщество, которое делится своими 3D-моделями и STL-файлами (файлы для распечатки моделей), а также информацией в интернете. Быстрый поиск Google и YouTube выдаст тысячи ссылок, где пользователи демонстрируют свои 3D-проекты и обмениваются опытом работы в Blender 3D. Такая популярность обусловлена прежде всего тем, что программа на 100% бесплатная и с открытым исходным кодом.

Благодаря своему набору инструментов, Blender является универсальной программой для 3D-моделирования. Она используется в различных областях, начиная от создания визуальных эффектов для фильмов, видеоигр, дизайна 3D-моделей, заканчивая 3D-печатью, инструментами для создания разнообразных моделей, анимации, видеороликов, интерактивных игр. В свободном доступе располагается множество подключаемых модулей для расширения функционала и возможностей программы. Результаты работы в приложении можно сохранять для трёхмерной печати [41].



Рисунок 92. 3D-моделирование на Blender

В качестве примеров работы на Blender приведем два проекта, описывающие алгоритмы создания моделей в Blender. [42] Ниже представлен полный пошаговый алгоритм работы на Blender для разработки 3D-модели кольца с драгоценным камнем.

*Blender: 3D-моделирование кольца с драгоценным камнем*

Рассмотрим создание 3D-модели кольца [43]. Полученный результат можно распечатать на 3D принтере или отфрезеровать восковку для дальнейшего литья в золоте или серебре (Рисунок 93). Ниже представлен пошаговый алгоритм моделирования кольца.



Рисунок 93. Модель кольца, созданной на Blender

1. Готовим рабочее пространство на Blender:

1. Создать новый файл на среде Blender и удалим модель куба, предлагаемой средой: *DELETE* -> *ENTER*;
  2. Изменить обзор *NUMPAD-5*;
  3. Добавить окружность *SHIFT-A* -> *Circle*;
  4. Перейти в режим *Edit Mode TAB*;
- Результат указан ниже:

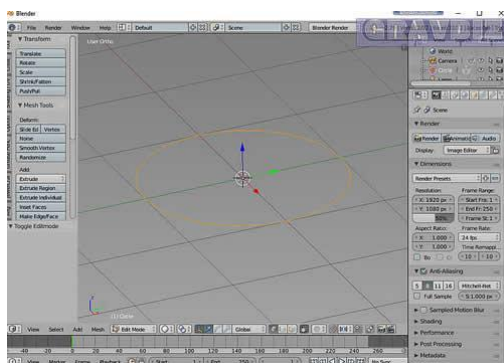


Рисунок 94. Создаем рабочее поле

2. Моделируем контур кольца:

1. Дважды нажать клавишу *A* (*Select All*), затем нажать *E* (*Extrude*);
  2. Нажать *S* для изменения толщины шинки, далее нажать *ENTER*;
  3. Таким же образом подобрать ширину внешней шинки и нажать *ENTER*.
- Должны получить такой результат:

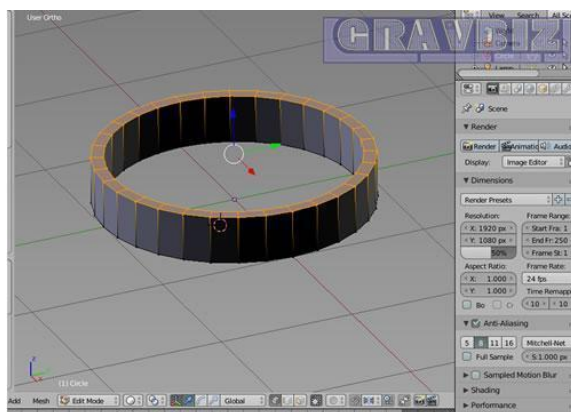


Рисунок 95. Контур кольца

### 3. Меняем контур кольца:

1. Нажать комбинацию клавиш *CTRL-R* и установить розовую линию по периметру внутри кольца, и кликнув *LMB* (левая кнопка мыши), передвинуть линию ниже верхнего края курсора мыши;
2. Нажать комбинацию клавиш *CTRL-R* и установить розовую линию по периметру внутри кольца, и кликнув *LMB*, передвинуть линию чуть выше нижнего края курсора мыши;
3. Нажать комбинацию клавиш *CTRL-R* установить розовую линию по периметру в центре снаружи кольца и кликнуть *LMB*. Должны получить такой результат

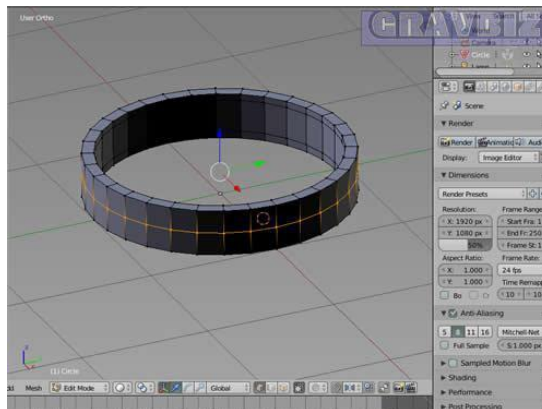


Рисунок 96. Изменения в контур

4. Нажать кнопку *S* и придать объем контуру кольца;
  5. Переключиться в режим *Object Mode* -> *TAB*.
- Результат показан ниже:

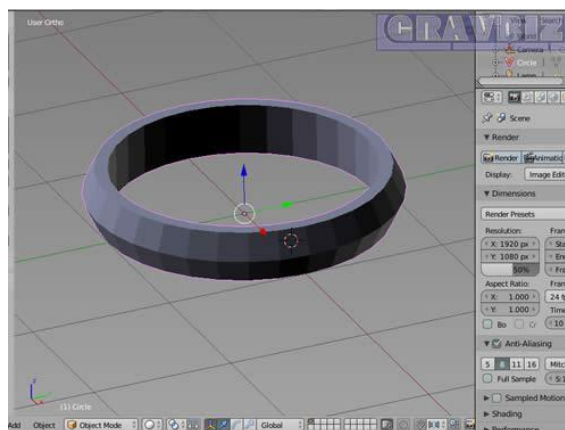


Рисунок 97. Подборка толщины

6. Добавляем элемент из списка модификаторов: *Modifier* -> *Subdivision Surface*

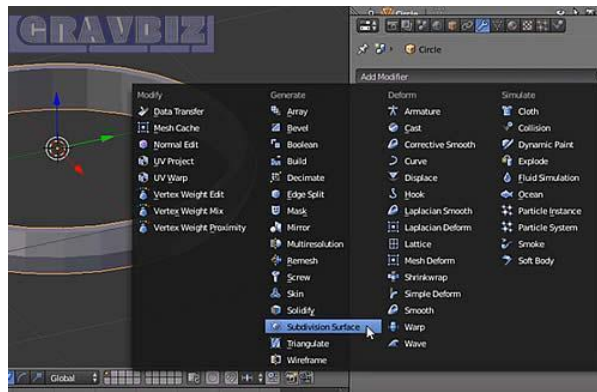


Рисунок 98. Модификаторы Blender

- Устанавливаем для дальнейшего моделирования следующие параметры : *Subdivisions - View - 3, Subdivisions - Render - 3*. В левом меню выбираем *Tools -> Shading -> Smooth*. Получаем следующее:

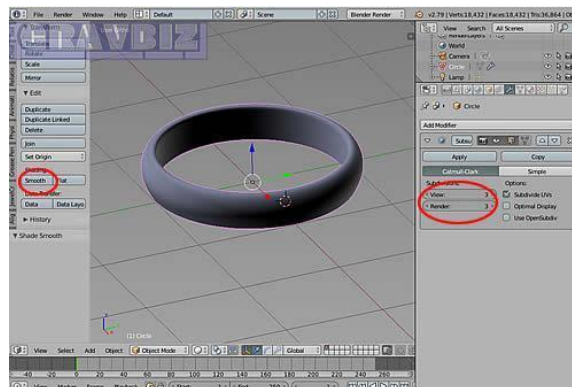


Рисунок 99. Параметризация модели

- Сдвигаем полученное кольцо по оси *X* :

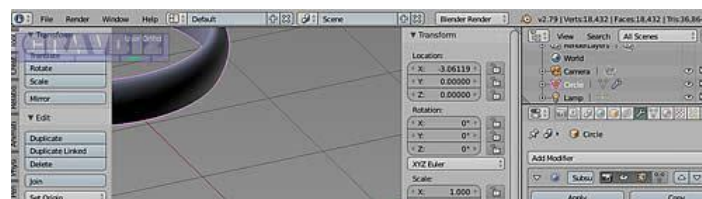


Рисунок 100. Движение по оси *X*

- Добавить окружность *SHIFT-A -> Circle*;
  - Повернуть окружность по оси *X* на 90 град (клавиша *R* или *Rotation*);
  - Нажать *S -> 0.5 -> Enter* Уменьшаем масштаб в 2х раза;
  - Перейти в режим *Edit Mode TAB*;
  - Дважды нажать клавишу *A*;
- Получаем промежуточный результат:

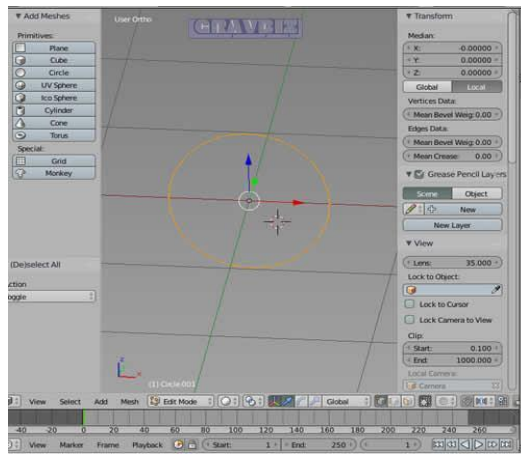


Рисунок 101. Изменения Edit Mode

5. Вытягиваем кольцо по оси  $Y$ .  $E \rightarrow Y \rightarrow 0.05 \rightarrow ENTER$ ;
6. Далее  $S \rightarrow 1.1$  .Получаем следующее:



Рисунок 102. Вытягиваем контур кольца

7. Выделяем внутреннюю грань  $ALT - RMB$  (правая кнопка мыши);
  8. Далее  $E \rightarrow Y \rightarrow 0.01 \rightarrow ENTER$ ;
  9. Далее  $S \rightarrow 0.97 \rightarrow ENTER$ ;
- Результат указан ниже:

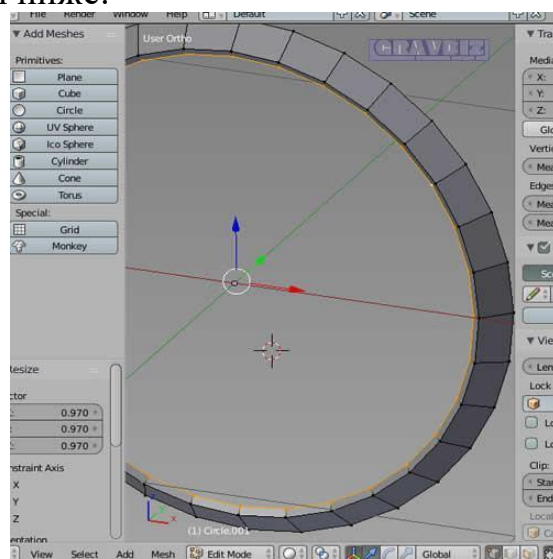


Рисунок 103. Настройка параметров внутренней рамки



10. *ALT - RMB - E* сдвинуть по оси *Y* -> *S* -> *1.3* -> *ENTER*. Получаем результат (Рисунок 104(1));
11. Вращаем внешнюю окружность *R* -> *Y* -> *10* -> *ENTER*;
12. Вытягиваем внешнюю окружность *E* -> *Y* -> *0.1* -> *ENTER*;  
Результат (Рисунок 104(2))

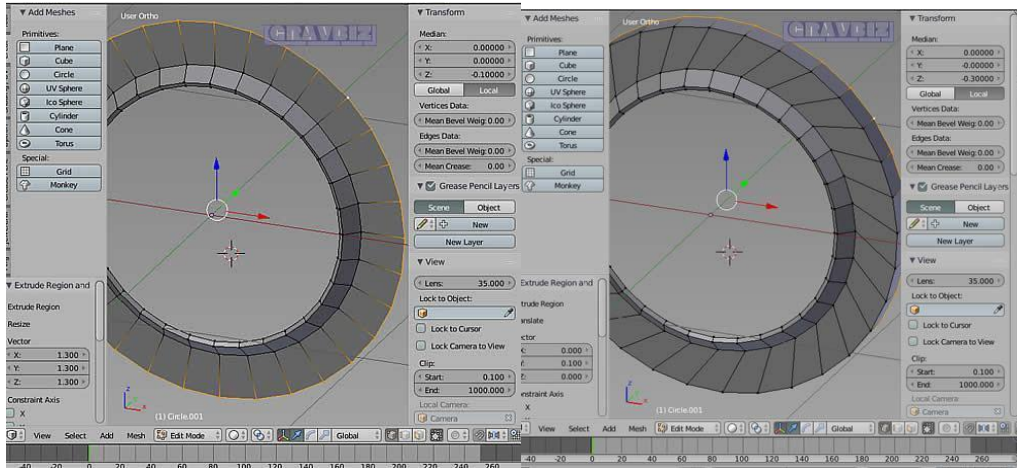


Рисунок 104. Изменение контура

13. Поворачиваем к обратной стороне (1) и нажимаем *F* - добавление "крышки" (2):

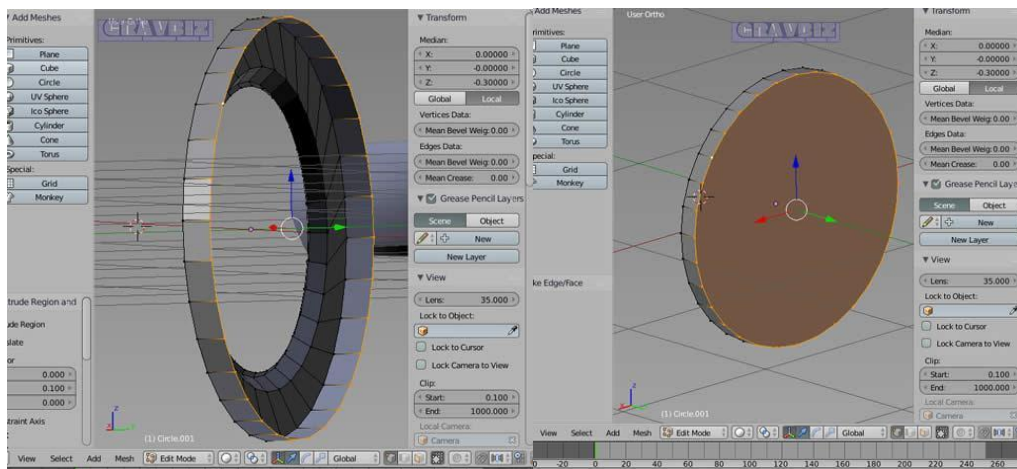


Рисунок 105. Поворот кольца

14. Развернуть модель лицом. Перейти в режим *Face Select* (кубики внизу) и выделить область *ALT - RMB* как на картинке (1);
15. Вытянуть сторону. *Tools* -> *Add* -> *Extrude* -> *Individual Faces* -> *-0.08* -> *ENTER*;
16. *TAB* (возврат в Object Mode) ;
17. Добавить Modifier. Вкладка *Modifiers* -> *Add Modifier* -> *Subdivision Surface* Задаем параметры *View* -3, *Render* -3, *Shading* - *Smooth* Получаем (2):



Рисунок 106. Модификация кольца

18. Изменить форму.  $S \rightarrow X \rightarrow 0.7 \rightarrow ENTER$ ;
19. Сохранить промежуточный вариант  $CTRL S$ ;
20. Отодвинуть заготовку из центра, чтоб не мешала делать камень.;
21. Делаем камень.  $SHIFT A \rightarrow UV Sphere \rightarrow TAB$ ;
- Результат (1)
22. Нажать  $A$  2х раза;
23. Вращать сферу  $R \rightarrow X \rightarrow 90 \rightarrow ENTER$ ;
24. Изменить обзор  $NUMPAD 3$  и нажать  $A$  ;
25. Перейти в режим *Vertex Select* (маленькие кубики внизу) (2)

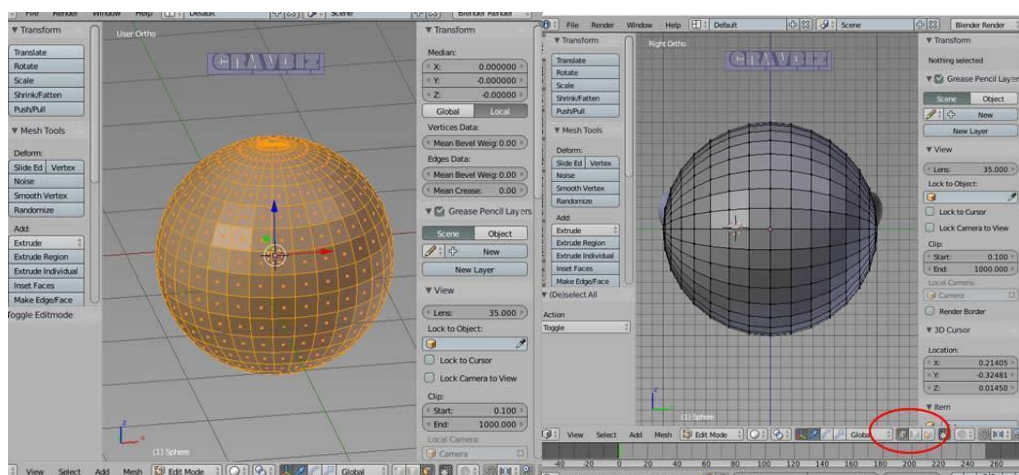


Рисунок 107. Подбор внешнего вида

26. Перейти в режим *Wireframe*;
27. Кнопка  $B$  и выделить половину сферы и удалить кнопка  $X \rightarrow DELETE \rightarrow VERTICES$  (1);
28. Перейти в режим *Object Mode* ( $TAB$ ) и *Viewport Shading* -> *Solid*
- Результат на рисунке (2);



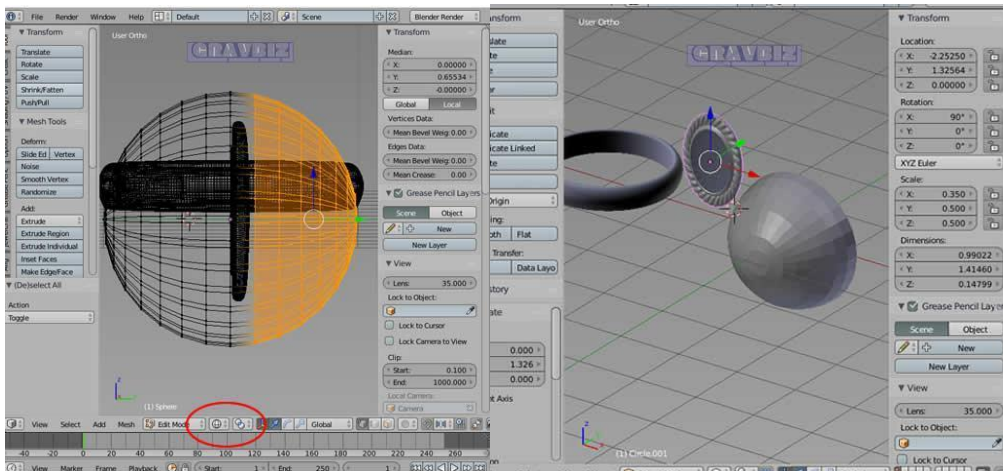


Рисунок 108. Работа с Wireframe

29. Увеличить сферу  $S \rightarrow Y \rightarrow 0.5 \rightarrow ENTER$ ;
30. Превратить в овал  $S \rightarrow X \rightarrow 0.7 \rightarrow ENTER$ ;
31. Уменьшить камень  $S \rightarrow 0.5 \rightarrow ENTER$ ;
32. Добавить Modifier. Вкладка *Modifiers*  $\rightarrow$  *Add Modifier*  $\rightarrow$  *Subdivision Surface*. Задаем параметры *View* -3, *Render* -3, *Shading* - Smooth;
33. Переместите кабошон в центр. Должно получиться так (1);
34. Передвинуть резинку к центру капошона;
35. Немного утопите камень;

Результат на рисунке (2)

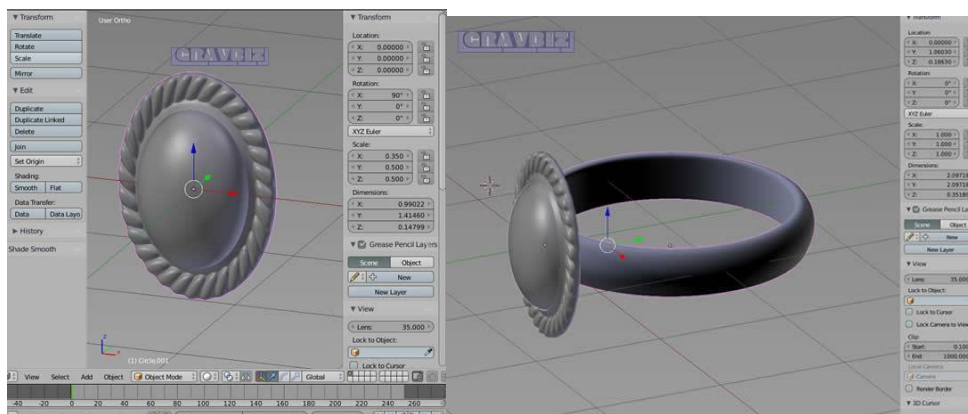


Рисунок 109. Соединяем кольцо с камнем

## 2. Раскраска объекта:

1. Изменить обзор *Numpad-1*;
2. Выделить RMB камень;
3. Открыть вкладку *Material*;
4. Добавить новый материал - кликнуть на *New Material*;
5. Включить *Cycles Render* в верхней части экрана:



6. Вкладка *Material* изменит вид следующим образом:

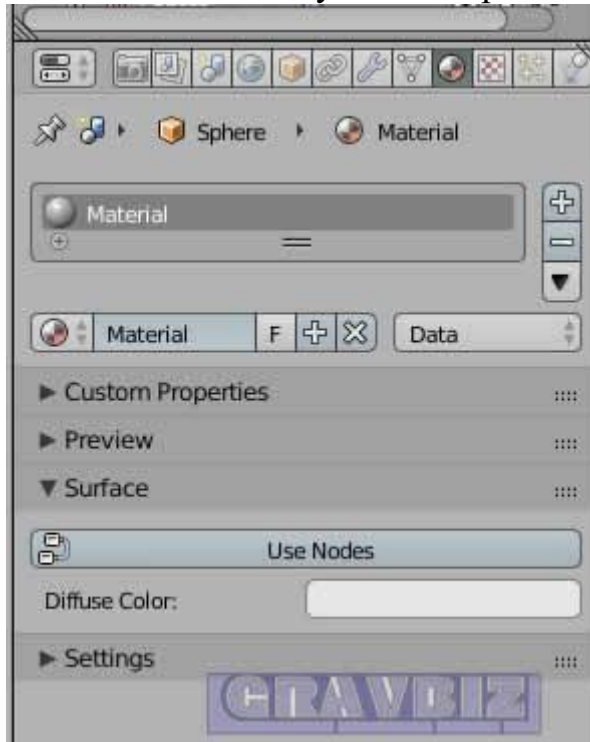


Рисунок 110. Вкладка *Material*

7. Нажать кнопку *USE Nodes* и переименовать *Material* в *Stone* (дважды кликнуть на названии *Material* и ввести новое название)
8. Выделить капюшон *RMB* и добавить новый материал - кликнуть на *New Material*;
9. Изменить параметр *Surface* на *Glossy BSDF*, кликнуть на поле *Color* и указать шестнадцатеричный код цвета (кнопка *HEX*) *C3C3D1* ->*ENTER*;
10. Параметр *Roughness* указать *0.05* -> *ENTER*;
11. Переименовать материал во *Front*;
12. Выделить шинку *RMB* и слева от *New Material* кликнуть на круг и выбрать *Front* (Рисунок 111 (1));
13. Добавить новый материал. Кликнуть кнопку *+* справа от названия материала (Рисунок 111 (2));

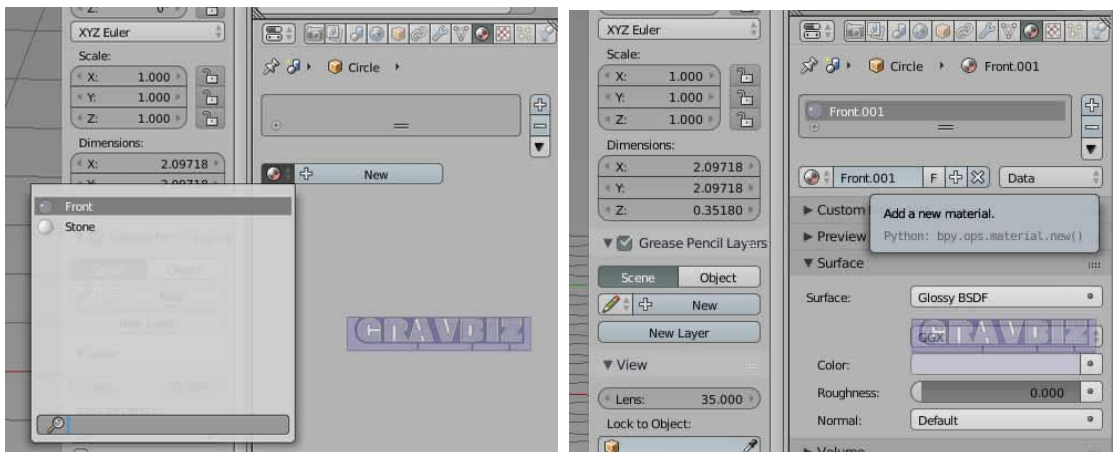


Рисунок 111. Вкладка *USE Nodes*

14. Переименовать новый материал в *Band*;
15. Объединить *Stone Front* и *Band* - выделить по очереди три объекта при нажатой *SHIFT* и нажать слева *Tools* -> *Edit* -> *Join*;
16. Повернуть кольцо по оси *Y*: Выделить кольцо *RMB* -> *R* -> *Y* -> *-7* -> *ENTER*;
17. Изменить обзор *Numpad-3*, повернуть кольцо по оси *X*: Выделить кольцо *RMB* -> *R* -> *X* -> *-15* -> *ENTER*

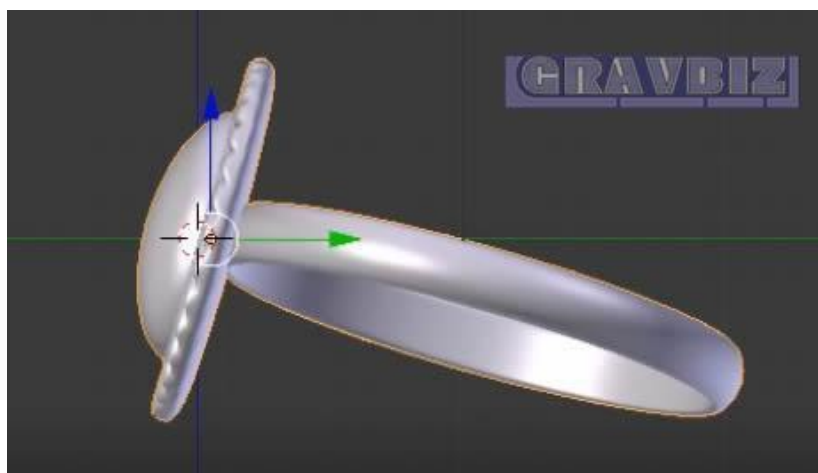


Рисунок 112. Обзор кольца в *Numpad-3*

18. Добавить основание для кольца: *SHIFT A* -> *Mesh* -> *Plane* и изменить разме *S* -> *10* -> *ENTER*. Опустить получившуюся плоскость под кольцо;
19. Добавить материал: Вкладка *Material* -> *New Material (Diffuse BSDF)*;
20. Добавить светоотражатели: *SHIFT A* -> *Mesh* -> *Plane* -> *Numpad 3*;
21. Поднять светоотражатель над кольцом, сместить перед кольцом, наклонить на 45 град и сместить влево (по оси *X*)

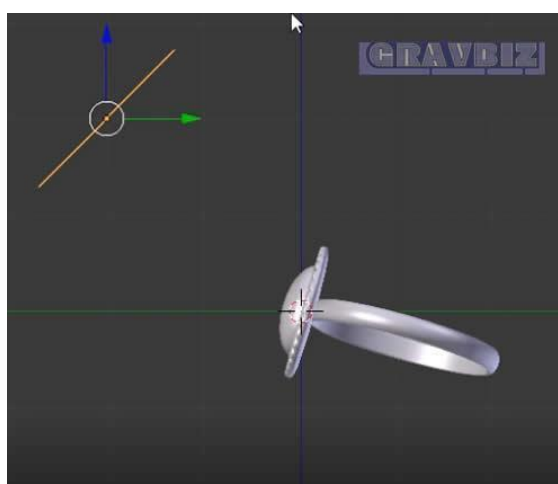


Рисунок 113. Установка светоотражателя

22. Создать копию светоотражателя *SHIFT D* и переместить его вправо от кольца;

23. Нажать *Numpad 7* (вид сверху), повернуть отражатели на кольцо: выделить отражатель кнопка *R* повернуть мышкой *Enter*:

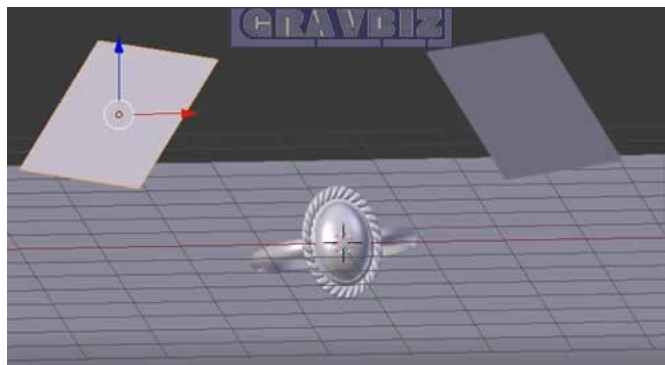


Рисунок 114. Выбор отражателя

24. Добавить материал к светоотражателям. Выделить *Левый отражатель*, нажать *New Material -> Surface -> Emission*. Задать цвет светло-желтый *Color HEX E7E0C2. Strength 20*;

25. Выделить *правый отражатель*. Нажать *New Material -> Surface -> Emission*. Задать цвет светло-голубой *Color HEX C1D5E7. Strength 15*

26. Сохранить файл - *Ctrl+S*;

27. Установить камеру. Нажать *Numpad 0*;  
Получаем вид из камеры:

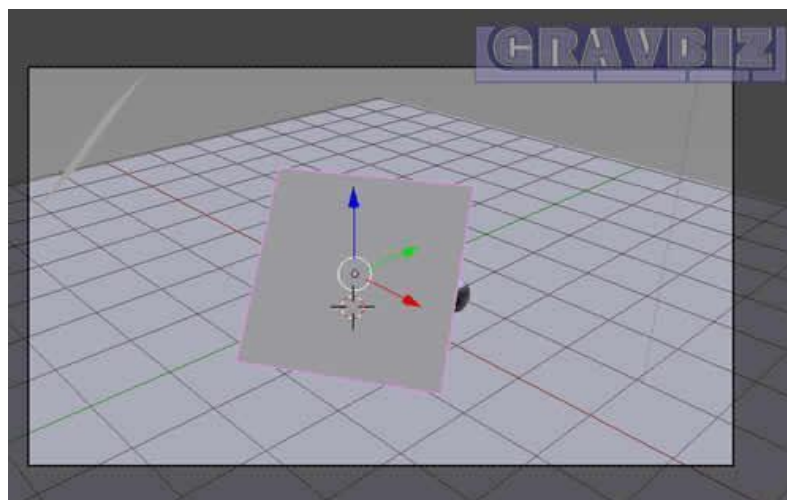


Рисунок 115. Вид из камеры

28. Заблокировать вид из камеры. Вкладка *Transform -> View -> Lock Camera to View*;

29. Изменить фокальное расстояние камеры. Выделить *LMB* камеру в списке *Scene*. Для параметра *Focal Length* указать 100.

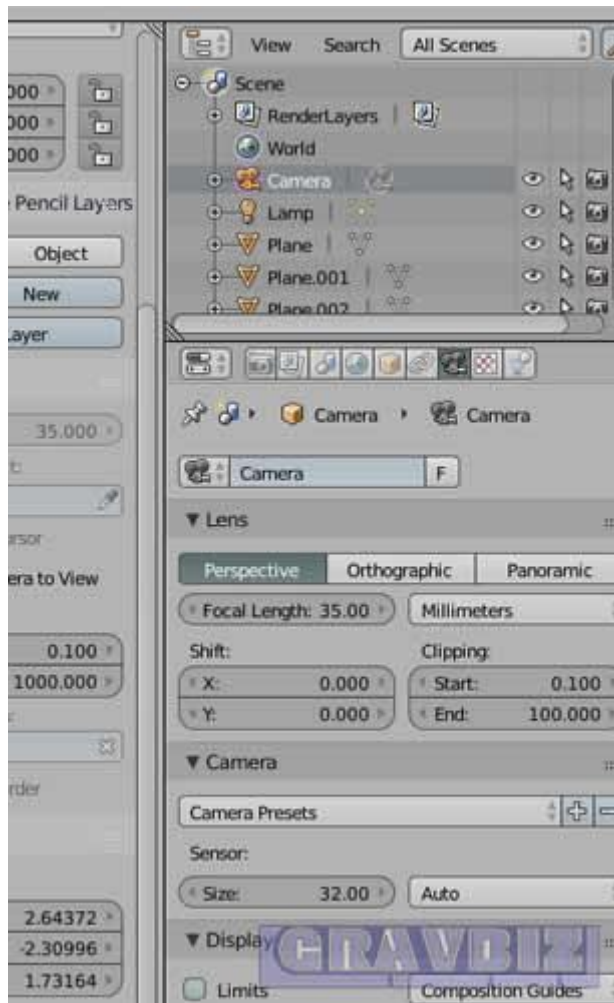


Рисунок 116. Настройка Camera to View

30. Пересчет узлов кольца. Выделить кольцо *RMB* -> *TAB* -> *A2* *Раза* -> *Вкладка слева Shading / UVs* -> *Normals* -> *Recalculate* -> *Tab*

31. Установка связей освещения/цвета. Включить режим *Compositing*

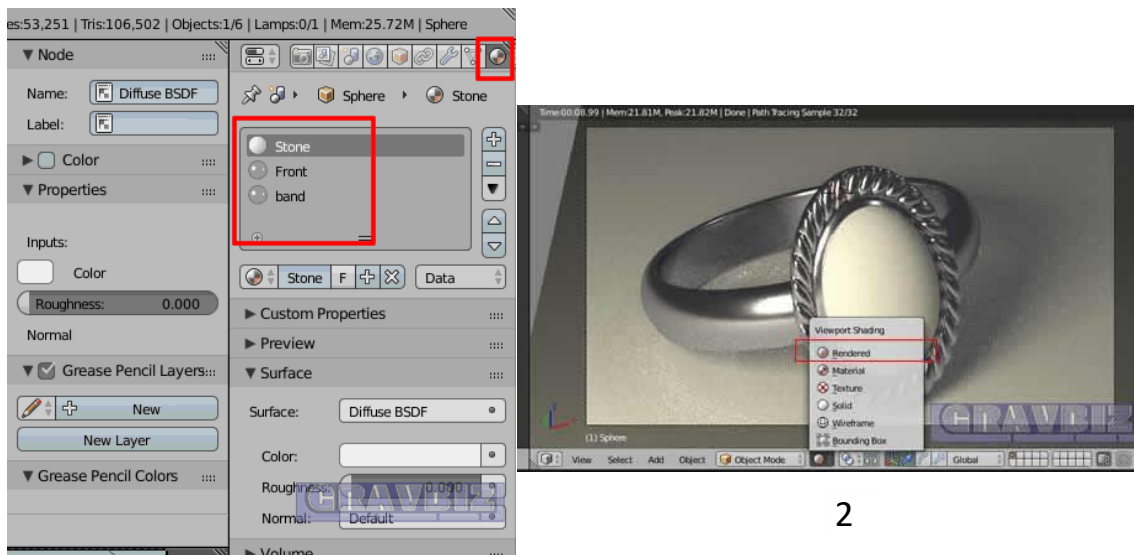


Рисунок 117. Режим Compositing

32. Нажать кнопку справа *Material*. Всего имеется 3 материала. (1);

33. Включить способ отображения *Rendered*. Кольцо изменит вид (2);





1

2

Рисунок 118. Рендеринг кольца

34. В окне *Material* выбрать материал *Front*;
35. Перейти в окно *Compositing* -> **SHIFT A** -> *Shader* -> *Mix Shader*;
36. Перетащить новую вкладку *Mix Shader* между *Class BSDF* и *Material Output*/ После того как отпустите кнопку мыши, установятся новые связи

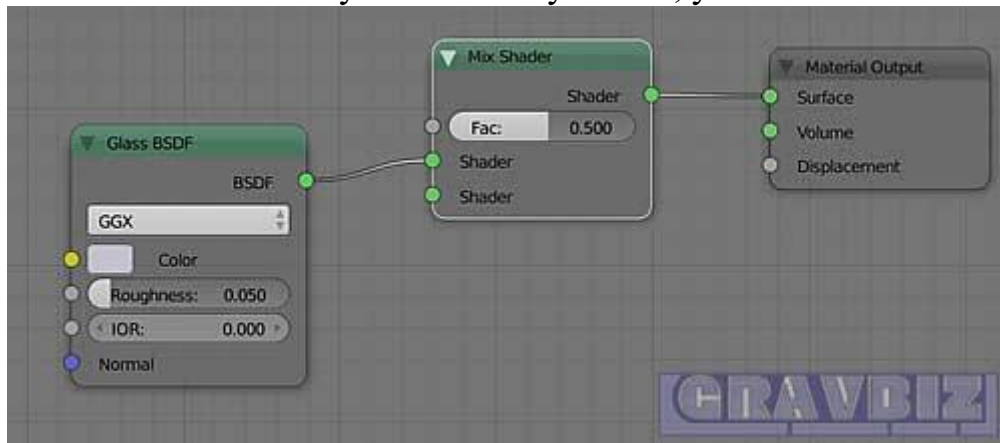


Рисунок 119. Связь между между Class BSDF и Material Output

37. Переместить связь В окне *Max Shader* на нижнюю зеленую точку
38. Добавить окно *Glossy BSDF*: **SHIFT A** -> *Shader* -> *Glossy BSDF*.  
Установить черный цвет;
39. Установить связь с окном *Mix Shader*, как на рисунке:

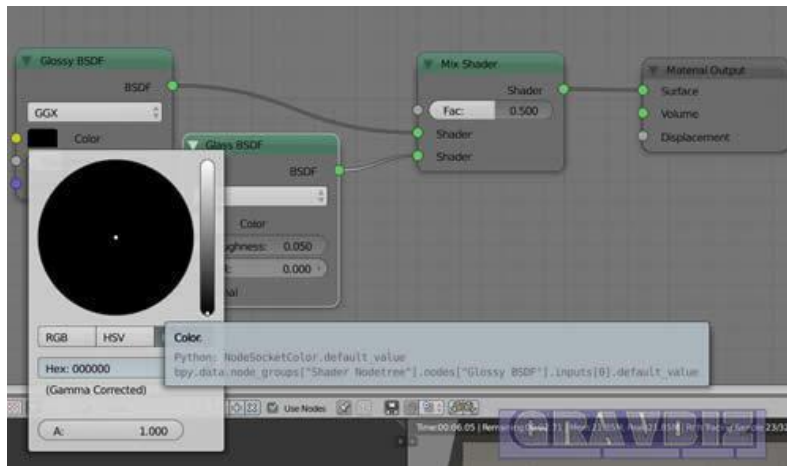


Рисунок 120. Связь с окном Mix Shader,

40. Добавить окно *Geometry: Shift A -> Input -> Geometry*;
41. Добавить окно *Math (1): Shift A -> Converter -> Math*;
42. В окне Math задать параметр *Value (верхнее): Multiple*, параметр *Value (средний) - 50*;
43. Добавить еще одно окно *Math (2): Shift A -> Converter -> Math*. Параметр *Value (верхнее): Subtract*. Параметр *Value (средний) - 25*;
44. Установить связи, как на рисунке;
- 45.

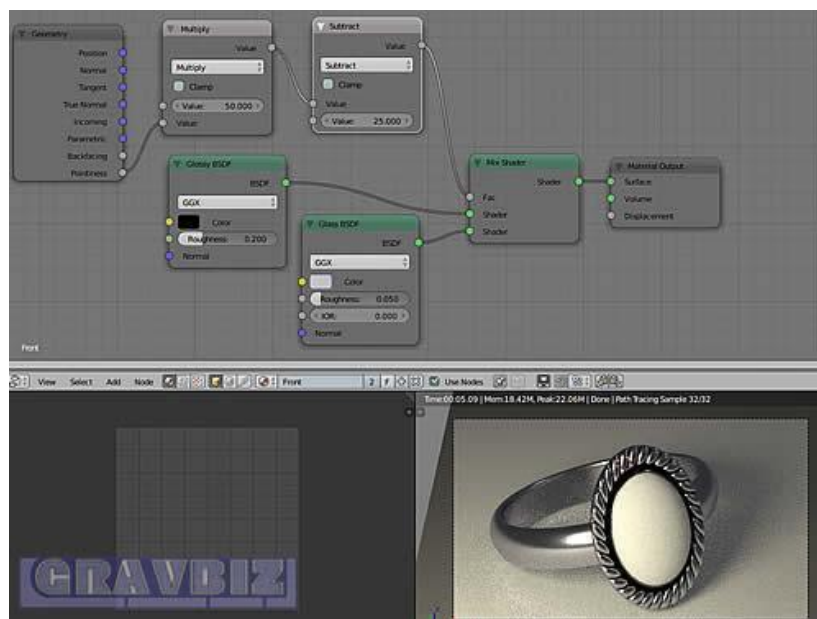


Рисунок 121. Связи окон параметров

46. При увеличении в окне *Math (1)* параметра *Value*, объем черного цвета на кольце уменьшится. Значение в данном примере - 50.6.

### Подбор материала для камня. Настройка внешнего вида

1. В окне Material выберите Stone;
2. В вкладке *Compositing -> Diffuse BSDF -> Color (HEX 15bbe2) -> ENTER*. Получаем небесно-голубой камень (Рисунок 122)



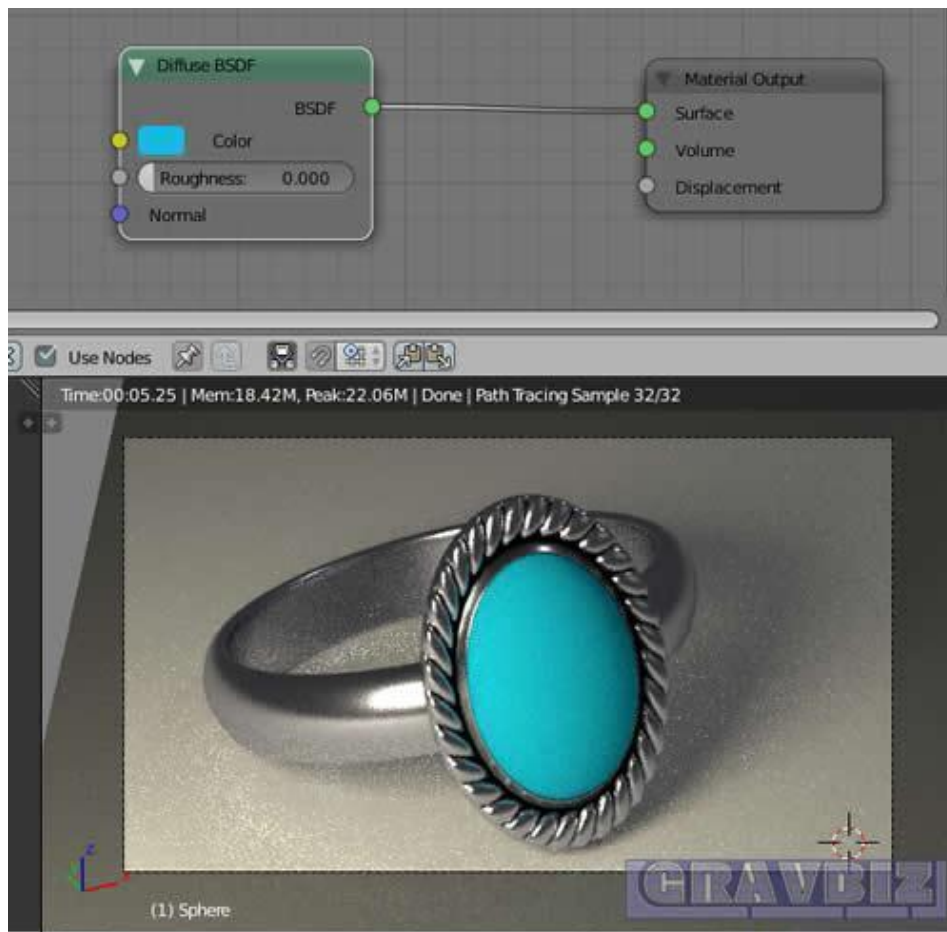


Рисунок 122. Подбор цвета камня

3. Создадим гладкую поверхность на камне. Во вкладке *Compositing* выбираем *Shift A -> Shader -> Mix Shader*
4. Добавим окно *Glossy BSDF*: *Compositing -> Shift A -> Shader -> Glossy BSDF*. Параметру *Color* задать значение белый, параметру *Roughness* - 0.02.
5. Установить связи, как на рисунке:

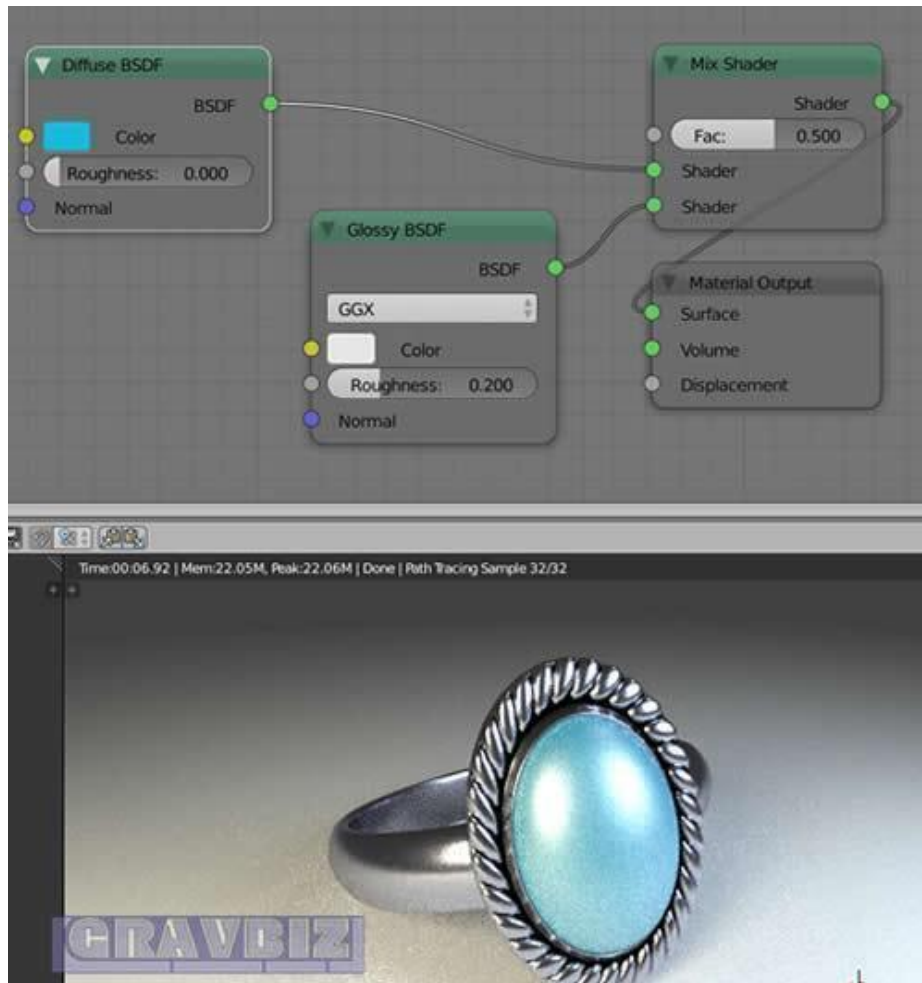


Рисунок 123. Редактируем внешний вид

6. В вкладке *Compositing* добавить окно - *Layer Weight: Shift A -> Input -> Layer Weight*. Параметру *Blend* установить значение 0.15;
7. В вкладке *Compositing* выбираем *Shift A -> Shader -> Mix Shader*;
8. В вкладке *Compositing -> Shift A -> Shader -> Diffuse BSDF*. Параметр *Color* задать значение *HEX 213010 -> ENTER*;
9. Разместить окна и связи, как на рисунке:

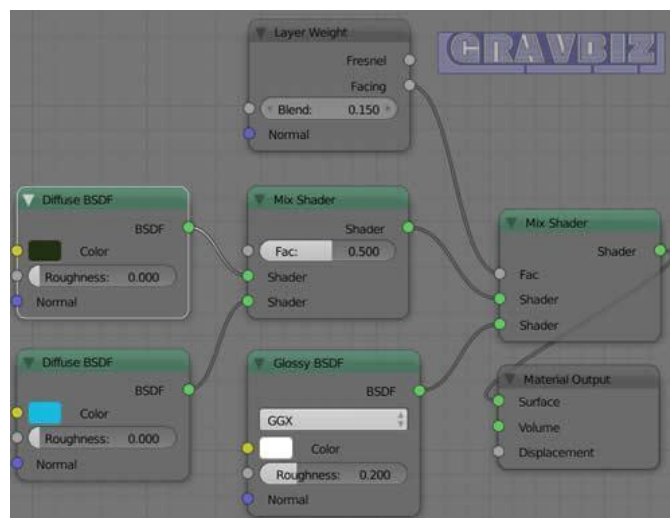


Рисунок 124. Установка связей между окнами

10. Добавим текстуру камню. Вкладка *Compositing* -> *Shift A* -> *Texture* -> *Noise Texture*
11. Добавить окно *Math* (1). Вкладка *Compositing* -> *Shift A* -> *Converter* -> *Math*. Параметр *Value* (верхний) выбрать *Multiply*. Задать *Value* (нижний): 5;
12. Добавить окно *Math* (2). Вкладка *Compositing* -> *Shift A* -> *Converter* -> *Math*. Параметр *Value* (верхний) выбрать *Subtract*. Задать *Value* (нижний): 2.5;
13. Добавить окно *Math* (3). Вкладка *Compositing* -> *Shift A* -> *Converter* -> *Math*. Параметр *Value* (верхний) выбрать *Subtract*. Задать *Value* (средний): 2.5;
14. Установить связи, как на рисунке

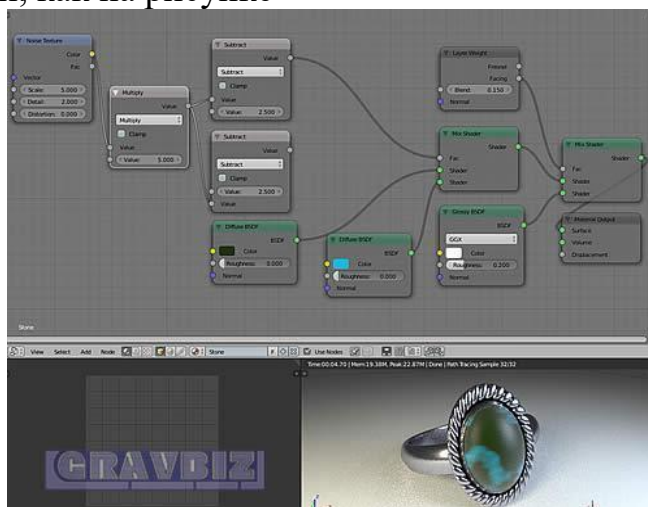


Рисунок 125. Связи окон параметров

15. Добавить окно *Math* (4). Вкладка *Compositing* -> *Shift A* -> *Converter* -> *Math*. Параметр *Value* (верхний) выбрать *Maximum*.
16. Добавить окно *Math* (5). Вкладка *Compositing* -> *Shift A* -> *Converter* -> *Math*. Параметр *Value* (верхний) выбрать *Power*. Задать *Value* (нижний): 0.3;
17. Установить связи, как на рисунке

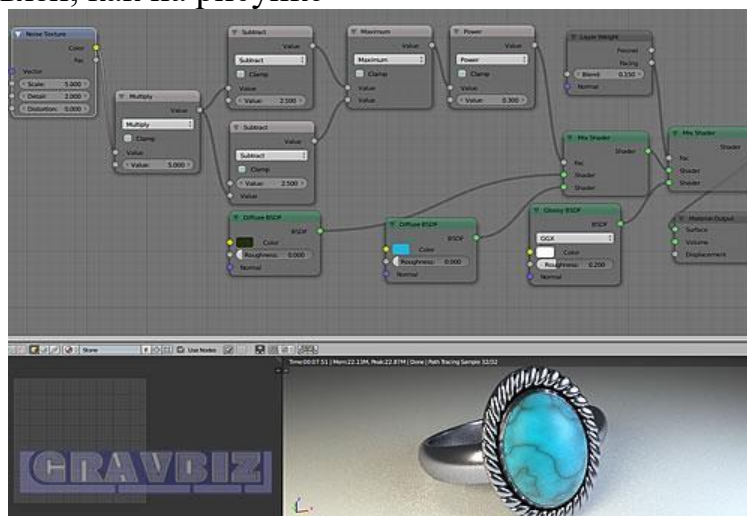


Рисунок 126. Связи окон параметров

18. В окне *Noise Texture* измените параметр *Scale*: 3.2, параметр *Detail*: 16;  
 19. В верхнем окне *Subtract* измените *Value* до 2.52:

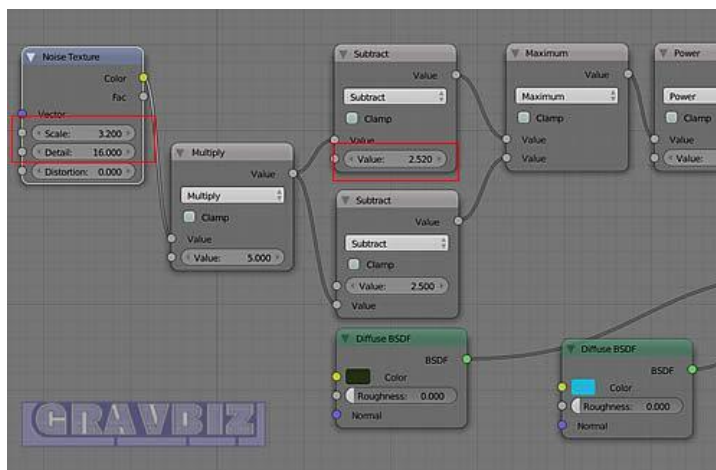


Рисунок 127. Меняем значение Subtract

20. Добавить тень к кольцу. Переключить режим просмотра *Compositing* в режим *Default*.  
 21. Добавить *Plain Axis*: *Shift A* -> *Empty* -> *Plain Axis*  
 22. Задать для созданного объекта *Plane Axis* параметры *Location*:  $Z=0.232$ ,  $Y=-0.116$

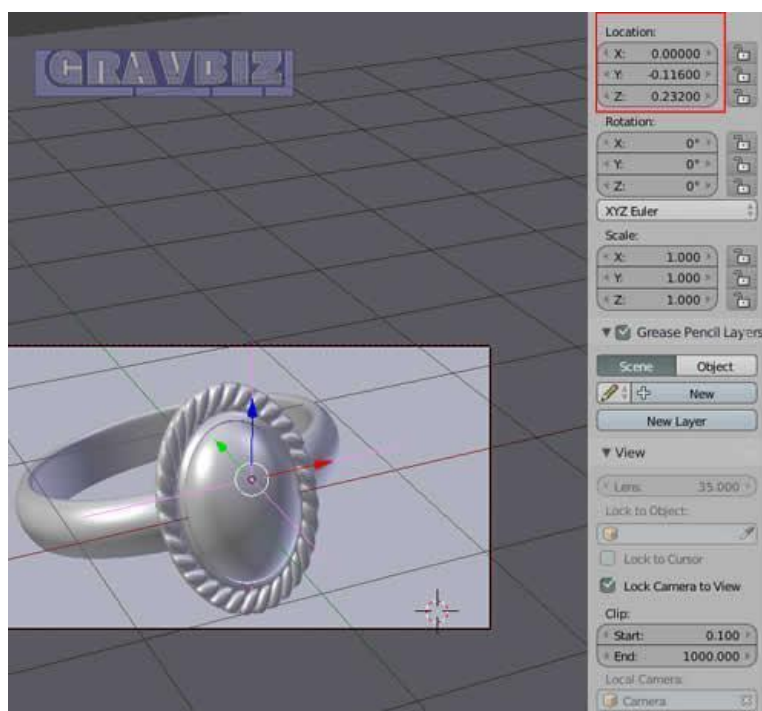


Рисунок 128. Редактируем режим просмотра

23. Нажать *Numpad 0*, включить режим *Rendered*;  
 24. Получится кольцо с песчаной рябью:



Рисунок 129. Песчаная рябь

- Исправим рябь, с помощью настройки камеры и детализации. Для этого:
25. Выделить справа в вкладке *View All Scenes* Камеру (*Camera*).
  26. В открывшейся вкладке выбираем *Camera* -> *Depth of Field* -> *Focus* -> *Empty*;
  27. В вкладке *Camera* выбираем *Depth of Field* -> *Aperture Type* -> *F-stop*. *Value* для параметра *F-stop* задать 1.
- Настройки указаны на рисунке (Рисунок 130):



Рисунок 130. Настройки камеры



Рендеринг конечного изображения:

28. Включить вкладку Render (значек фотоаппарата) -> Sampling -> Samples -> Render задать 1000, и сохранить файл (Ctrl S);

29. Нажимаем *Render*. Запустится процесс рендеринга, процесс может занять длительное время.

Итоговый вид кольца после рендеринга:



Рисунок 131. Кольцо после рендеринга

*Blender: Печать готовых моделей.*

Для начала работы, убедитесь в том, что на вашем компьютере установлена версия программы Blender не ниже 3.0., и ознакомьтесь с Пользовательскими сочетаниями клавиш для Blender .

Ниже представлены некоторые из наиболее важных комбинаций клавиш на Blender 3.0:

- **Tab**: Нажмите эту клавишу, если хотите войти в режим редактирования.
- **G**: позволяет перемещать заголовки или форму.
- **Shift + A**: позволяет заполнить пространство между выбранными вершинами.
- **S** или же **C + X** или же **Y** или же **Z**: Масштабирование объекта по осям X, Y или Z в зависимости от того, что вы хотите сделать.

Чтобы включить пользовательское расширение 3D Print Toolbox, выполните следующие действия:

- Перейти к **Правка -> Настройки**.
- Щелкните вкладку **Дополнения**.
- Найдите набор инструментов для 3D-печати.

Этот инструмент нужен для подготовки вашей модели к более плавному и успешному процессу 3D-печати.

## 1. Включите настройку 3D Print Toolbox (Рисунок 132).

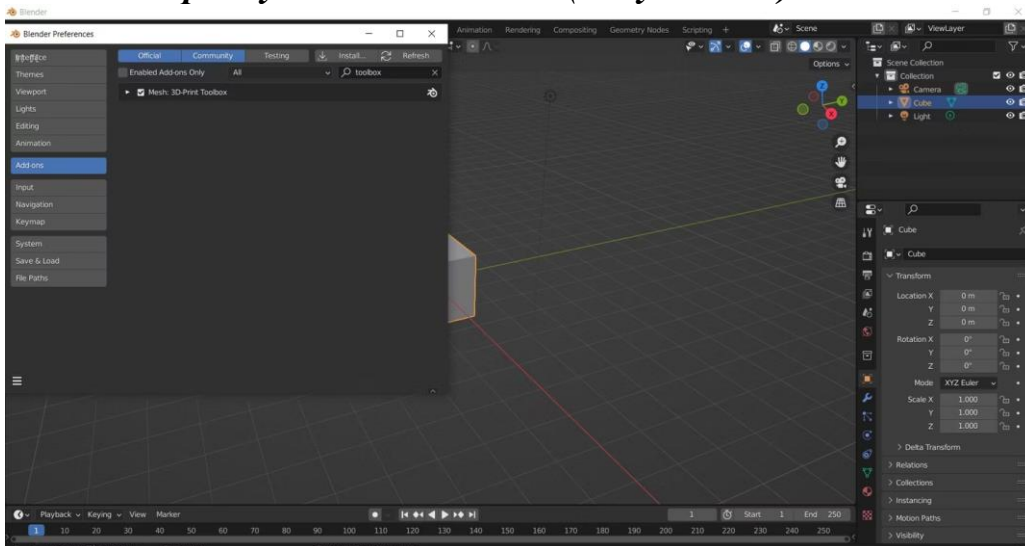


Рисунок 132. Настройка 3D Print Toolbox

## 2. Подготовьте свою 3D-модель

Для печати можете использовать любую из имеющихся шаблонов (Рисунок 133). Если готового шаблона у вас нет, то существует множество бесплатных моделей, доступных на таких сайтах, как Thingiverse или же Cults 3D. Однако, следует помнить, что не все из них предназначены для 3D-печати.



Рисунок 133. Подготовьте модель для печати

## 3. Оптимизация 3D-модели

После получения готового шаблона, важно помнить о следующем: ваша модель должна быть сплошным заполненным объектом. Это означает, что в вашей модели не должно быть отсутствующих деталей или лишних полостей (Рисунок 134).



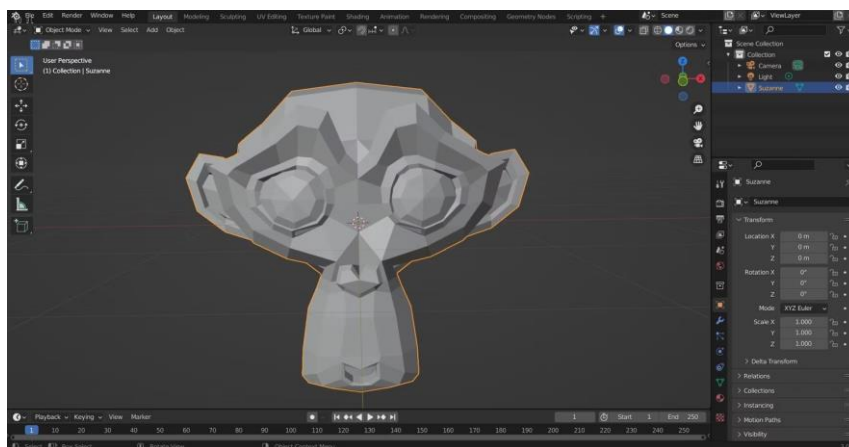


Рисунок 134. Оптимизация 3D-модели

Другими словами, эта трехмерная модель должна представлять из себя *полигональное многообразие*. Для этого, она должна быть окружена областью, которая могла бы существовать в реальной жизни (существующая в евклидовой геометрии).



Рисунок 135. Проверка многообразности

В случае, когда ваш объект является сложным и многосоставным, существует вероятность, что он может быть нецелым. Проверить это можно следующим образом:

- Выберите объект, который хотите проверить.
- нажмите клавишу **N** чтобы показать панель инструментов.
- выберите раздел **Набор инструментов для 3D-печати**.
- Нажмите кнопку «Проверить все».
- Посмотрите на результаты и убедитесь, что на нелакированном краю отображается 0.

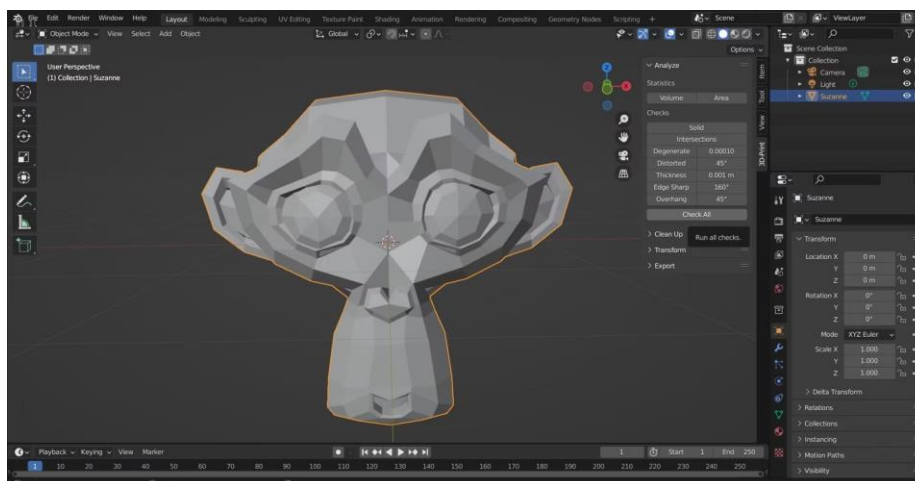


Рисунок 136. Проверка целостности объекта

В случае обнаружения значения выше нуля, объект нуждается в ремонте немногобразной (либо неразветвленной, англ: non-manifold) формы. Разберем пошаговую инструкцию ремонта.

*Ремонт неразветвленной формы: Способ 1.*

Начнем с самого простого способа исправления неразветвленной модели. Для этого можем использовать встроенный инструмент очистки Blender. Вы можете найти его рядом с кнопкой **Check All**. Выполните следующие действия:

- Выберите свою модель.
- Откройте раздел очистки.
- Кликните **Create a Collector**.
- 



Рисунок 137. Ремонт неразветвленной формы. Способ 1.

Данный способ также может работать и в старых версиях Blender. Однако этого подхода есть существенный недостаток: иногда он может испортить вашу 3D-модель. Если данный способ ремонта оказался неподходящим, следует воспользоваться методом, описанным ниже.

*Ремонт неразветвленной формы: Способ 2.*

Вместо использования Blender, можно попробовать внешнее приложение *3D Builder* от Майкрософт. Данное приложение доступно в версиях Windows 11 и 10. После загрузки и установки *3D Builder* откройте приложение и импортируйте свою модель. Затем, выполните следующие действия:

- Перейти к **Open -> Open Object...**
- Перейдите к своей 3D-модели, где бы она ни была сохранена на вашем компьютере.
- Нажмите **Import Model**.

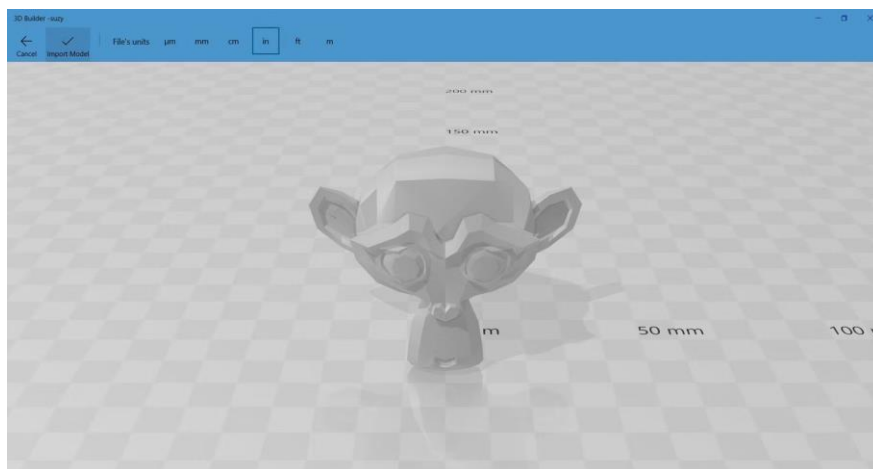


Рисунок 138. Ремонт неразветвленной формы. Способ 2.

3D Builder сообщит вам о наличии ошибок, с помощью всплывающих уведомлений, предлагающими пути их исправления. После выполнения необходимых шагов, 3D-модель должна быть починена.

*Примечание:* при использовании 3D Builder, приложение может изменить форму вашей модели, чтобы сделать ее целостной. Данные изменения сделают объект более удобным для печати.

#### ***4. Импортируйте свою модель в слайсер.***

На последнем этапе необходимо экспортировать модель из Blender и импортировать ее в приложение Slicing. Slicing — это приложение, преобразующее полученную модель в инструкцию по печати на 3D-принтере. Данный набор инструкций носит название G-code.

Другими словами, это - язык, сообщающий машине траекторию перемещения экструдера во время печати. Эти инструкции также могут содержать выбираемые пользователем параметры, такие как скорость печати и плотность наполнения модели.

Прежде чем импортировать модель в приложение для нарезки, необходимо экспортировать ее в виде файла в формате STL. STL, Standard Mosaic Language или Standard Triangle Language, описывает геометрию поверхности 3D-объекта.

Для этого, экспортируем объект как файл STL, перейдя в **Файл -> Экспорт -> Stl (.stl)** в Blender.

В данном примере, в качестве слайсера для полученной модели, используем Cura. Модель, перемещенная в Cura, показана на рисунке 139.

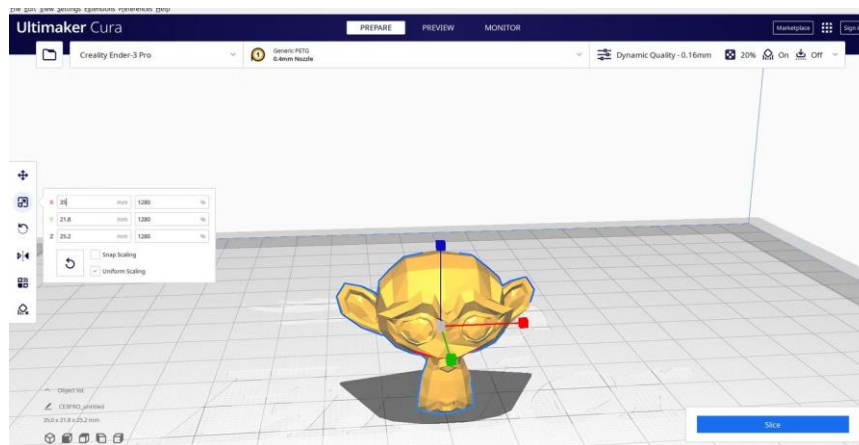


Рисунок 139. Модель, перемещенная в Cura

После того, как модель будет загружена на Cura, можно приступить к печати. Полное руководство по печати с использованием дополнительных плагинов представлено в источнике [44].

#### 2.4. Применение на практике программы FreeCAD

Данная глава описывает алгоритм работы с базовыми фигурами на среде 3D-моделирования FreeCAD [45]. Для начала, запустите FreeCAD и создайте новый файл (Рисунок 140).

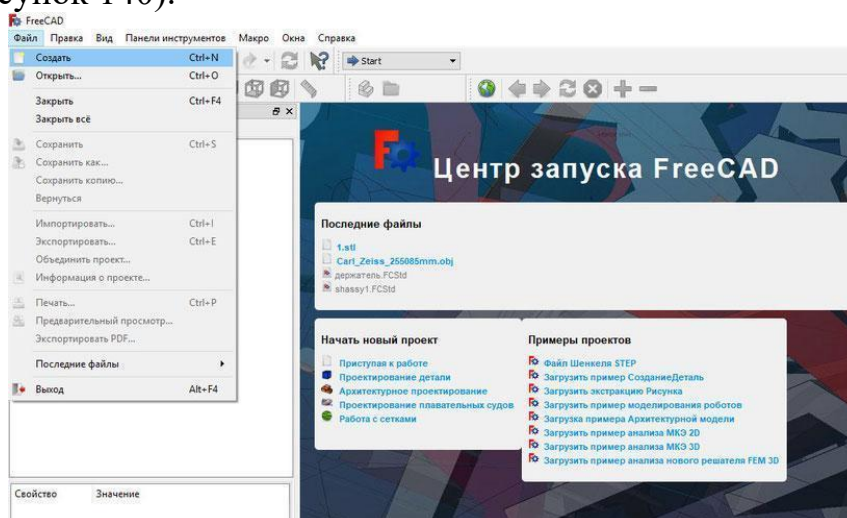


Рисунок 140. Создание нового файла

- Выбираем рабочий стол Part (в редакторе предусмотрено много различных рабочих столов, с назначением которых мы знакомимся по мере необходимости их применения)

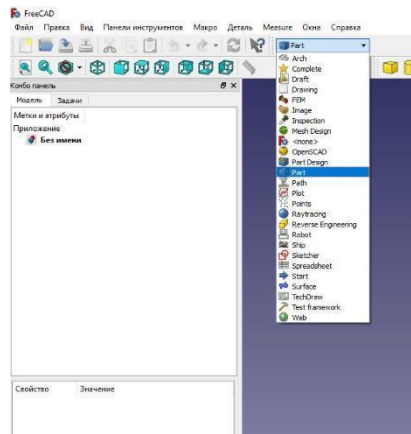


Рисунок 141. Рабочий стол Part

В качестве примера-базы нашей будущей модели рассмотрим создание квадратной пластины с заданными **параметрами**.

- Создадим куб, выбрав соответствующий инструмент на панели инструментов.

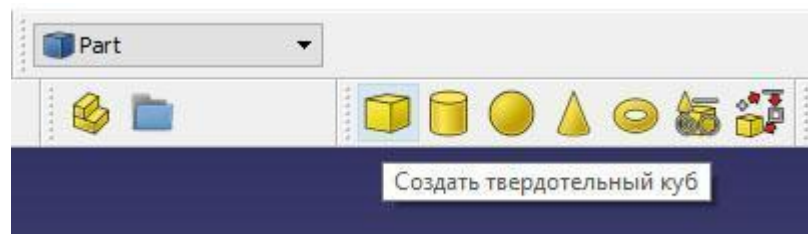


Рисунок 142. Панель инструментов

- Выделим слово куб в левой панели. Изображение станет светло зеленым, это означает, что фигура выбрана.

Различный вид фигуры можно выбирать в соответствующей панели инструментов Вид. Удобнее использовать аксонометрический вид. Для произвольного вращения фигуры можно использовать мышь, с зажатыми правой и центральной клавишей (нажатым колесиком мыши). Изменение масштаба - покрутить колесо мыши.

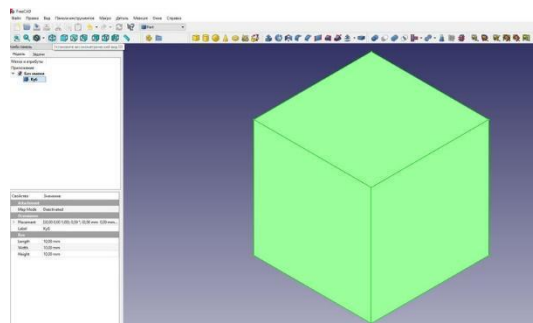


Рисунок 143. Создаем куб

- Зададим параметры куба (длина ребра, ширина, высота). Для экономии пластика высоту (толщину пластины) зададим 2 мм, ребра - по 10 мм.

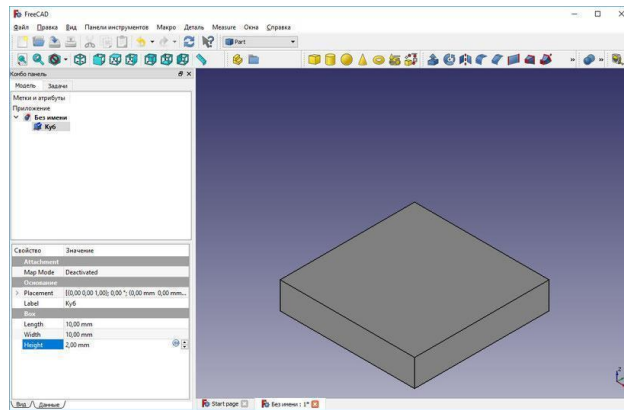


Рисунок 144. Задаем параметры куба

Для того, чтобы получившиеся пазлы можно было использовать, как брелоки, а также для экономии пластика, создадим в центре пластины большое отверстие. Для этого, создадим еще одну фигуру – Цилиндр

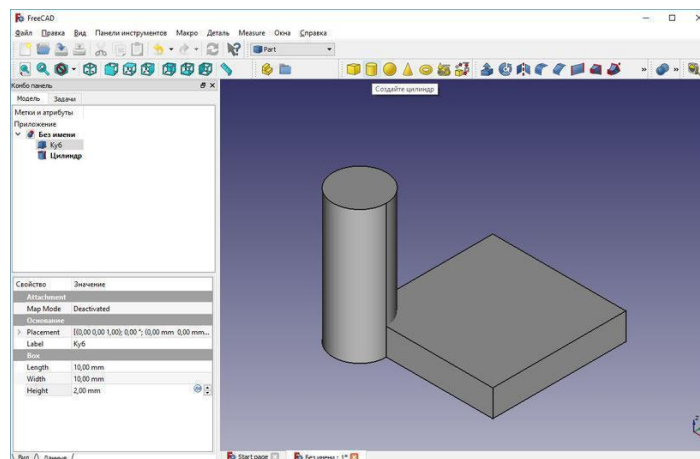


Рисунок 145. Создание цилиндра

Зададим параметры цилиндра, точнее один параметр - радиус. Здесь можно обсудить, какой именно. Исходя из желания оставить по краям базовой фигуры хотя бы по 1.5 мм пластика, радиус цилиндра может быть 3,5мм. Высота нас устраивает, ведь нам предстоит вычитать цилиндр из куба, но об этом позже.

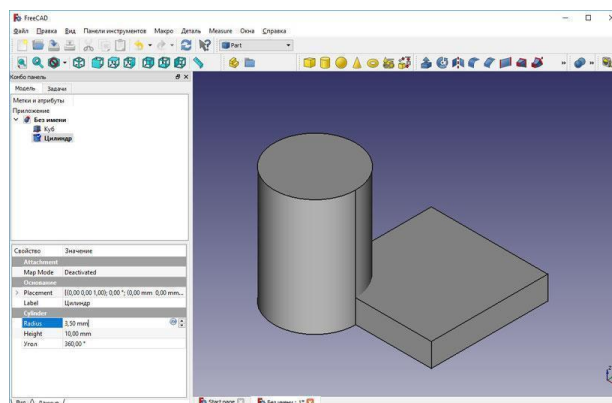


Рисунок 146. Редактируем параметры цилиндра



Теперь нам надо передвинуть цилиндр, изменив его координаты.

- Выделим цилиндр в левой панели.
- В окне Свойств и Значений зайдём в Положение (Placement) - слева три точки, там скрыты настройки.

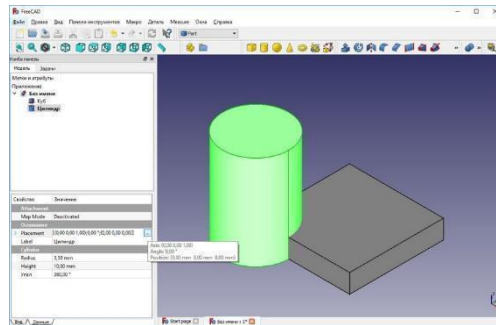


Рисунок 147. Перемещение цилиндра

Справа в углу видны координатные оси. Выставляем новые координаты центра цилиндра

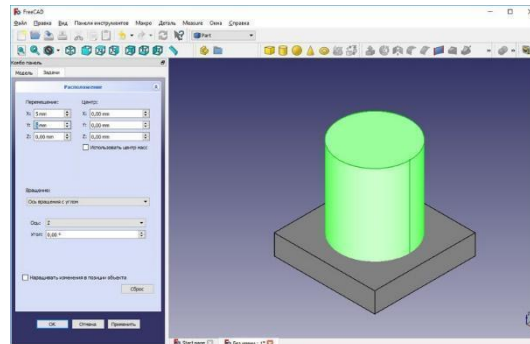


Рисунок 148. Выставляем новые координаты центра цилиндра

Далее мы хотим получить отверстие в базовой фигуре. Для этого надо вычесть цилиндр из куба)))

Выделяем слева в окне объектов сначала куб, затем, при нажатой клавише Ctrl - цилиндр. Последовательность важна.

Убеждаемся, что выделены обе фигуры (цвет в правом окне).

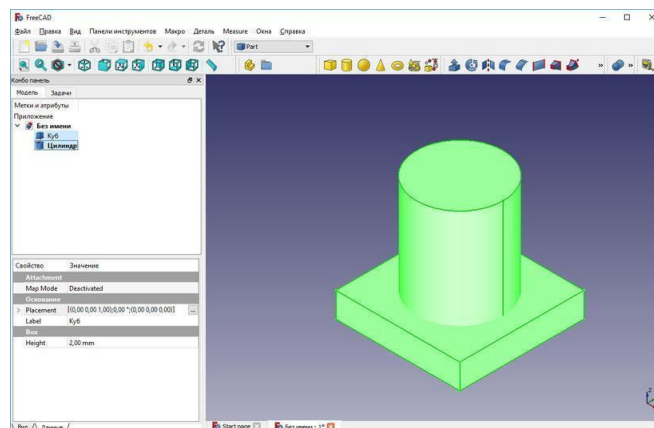


Рисунок 149. Выделение фигур



Применим операцию вычитания (в нашем случае - из куба вычтем цилиндр), выбрав нужную из панели (еще одно слово страшное) булевых операций. В новом переводе она значится как "Обрезать две фигуры".

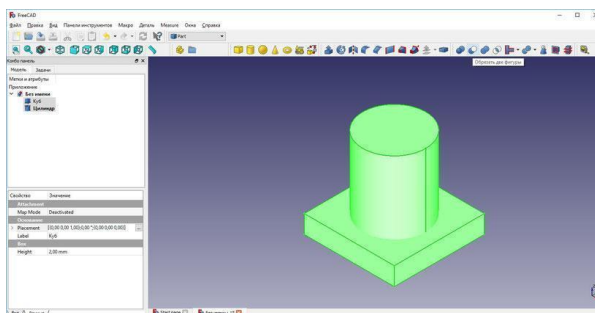


Рисунок 150. Операция “Вычитание”

Результат указан ниже:

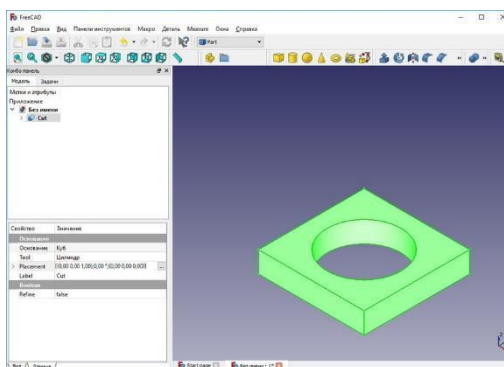


Рисунок 151. Результат вычитания

Базовый элемент нашего пазла готов. Но стоит показать еще пару операций для копирования объекта.

В результате нашей операции вместо отдельных фигур в левой части окна объектов теперь один объект - результат операции вычитания Cut.

Выделяем его и копируем (например, нажав комбинацию клавиш Ctrl+C). Появляется окно с предупреждением, что будут сделаны копии всех исходных объектов, соглашаемся.

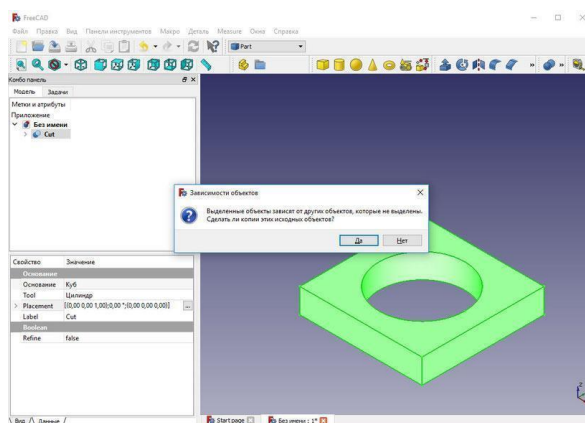


Рисунок 152. Копируем фигуру

Проводим операцию Вставить (например, комбинация Ctrl+V), в окне появляется новый объект, справа можно видеть его подсветку.

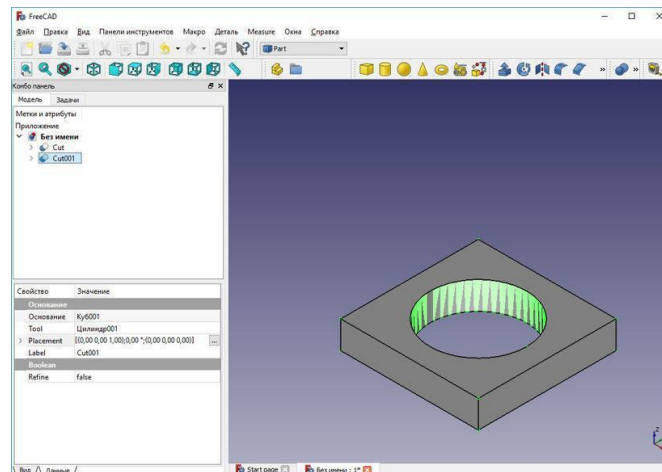


Рисунок 153. Вставка объекта

Выделяем новый объект Cut 001

Передвигаем объект в нужную сторону, войдя в настройки его положения (это мы уже проходили с Цилиндром). Сдвигаем на 10 мм в нужную сторону. Не забываем нажать ОК, подтверждая операцию.

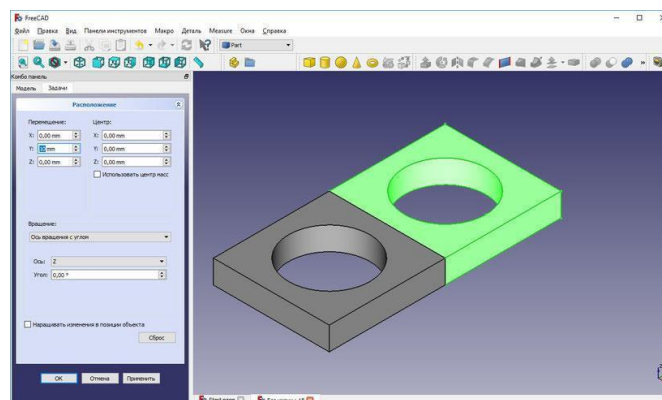


Рисунок 154. Перемещение копии объекта

Повторяя операцию Копировать-Вставить, и меняя положение объектов, создаем нужную конечную фигуру-пазл.

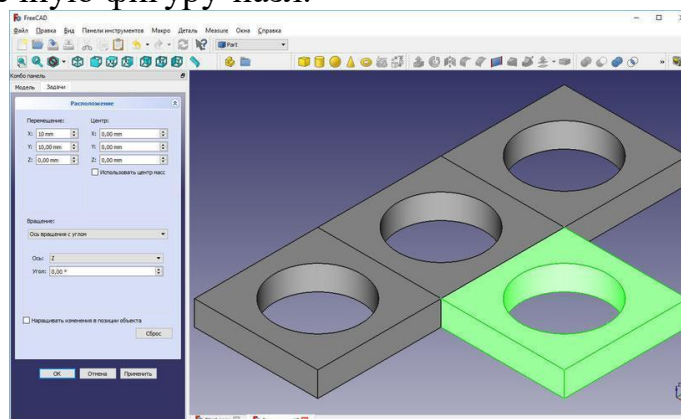


Рисунок 155. Перемещение копий объекта

Для дальнейшей работы удобно объединить полученную фигуру.

Выделяем все объекты в левом окне объектов. Смотрим, подсвечена ли вся фигура.

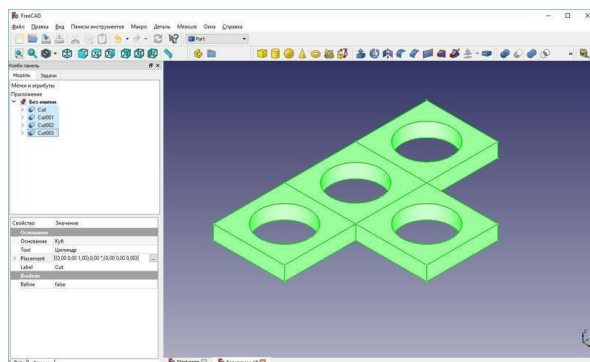


Рисунок 156. Вставка копий

Применяем операцию сложения (объединения нескольких фигур) из панели инструментов Булевы операции.

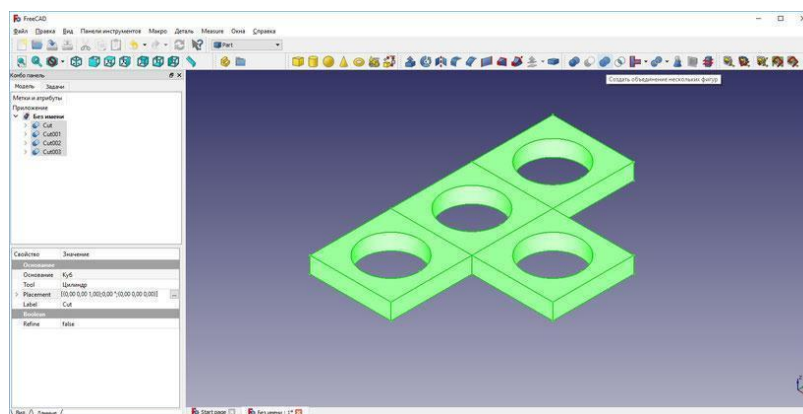


Рисунок 157. Операция сложения

Посмотрите, в результате вместо 4 объектов у нас появился один, результат – Fusion

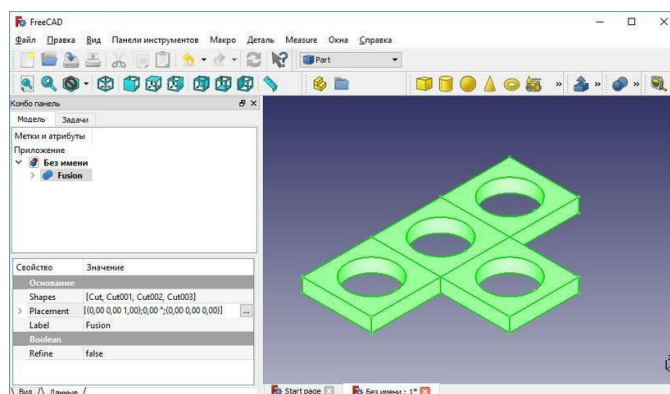


Рисунок 158. Fusion

Теперь файл можно сохранить для последующей редакции (в формате редактора)

Сохраним файл (меню Файл - Сохранить как, формат FCStd) в свою рабочую папку, имя - например Razl 4. Теперь нам нужен файл для последующей печати модели, а это другой файл, в формате STL. Чтобы его получить, выделим весь объект и применим операцию Файл - Экспортировать...

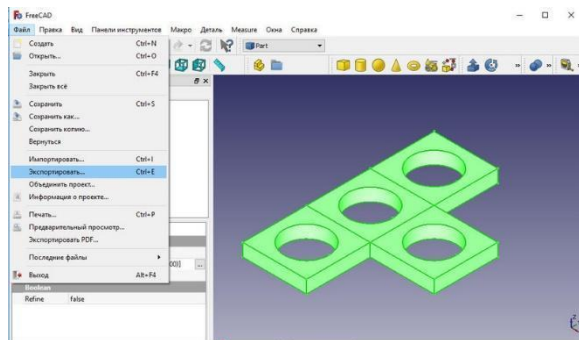


Рисунок 159/ Получение файла в формате STL

В окне экспорта зададим имя объекта, можно также - Razl 4, или tetris, как на рисунке, главное, чтобы вы понимали, что за модель лежит в вашей папке.

И выберем формат экспорта - в STL

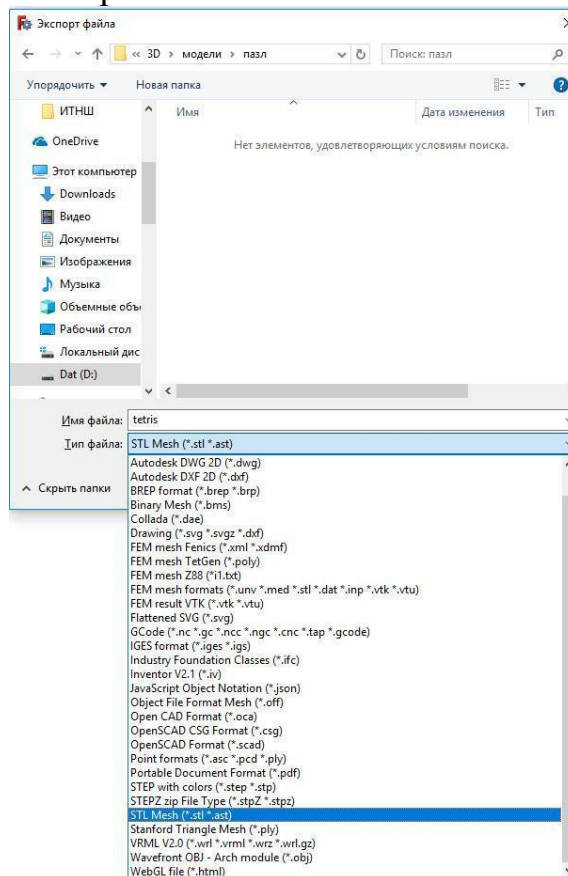


Рисунок. 160. Выбор формата

Теперь в нашей рабочей папке лежит файл в формате STL, а также в формате FCStd на случай, если мы захотим его переделать в редакторе FreeCad.

## Этап 5.

И теперь самое время поговорить о том, как подготовить файл в формате STL для печати на конкретном принтере, вот этом, mz3d-360C, стоящем в классе.

**STL** (от *англ.* stereolithography) — формат файла, широко используемый для хранения трехмерных моделей объектов для использования в аддитивных технологиях [46]. Свое название получил от сокращения термина «Stereolithography», поскольку изначально применялся именно в этой технологии трехмерной печати.

Т.е. файл, описывающий трехмерную модель у нас есть, но принтеры отличаются многими параметрами, отличаются и пластики, которые могут использоваться для печати. А значит, надо переделать имеющийся файл STL в файл, в котором для принтера будет пошагово объяснено, с какой температурой, с какой скоростью и где именно должно двигаться сопло, чтобы нарисовать нужную модель. И так для каждого слоя.

Именно для этого служат программы-слайсеры [47].

**Слайсер** (*англ.* slicer — нож) - это программа для нарезки 3D модели на слои для последующей печати на 3D принтере. Все алгоритмы печати и перемещений задаются в нём. Мы используем свободное ПО, наш основной слайсер - Cura [48] .

*Основы работы со слайсером Cura [49]*

- Запускаем программу Cura (ярлык на рабочем столе, или Программы - ...)
- Заходим в меню Принтер, проверяем, что выбран нужный (у нас - mz3d-360c)
- Команда меню Файл-Загрузить файл модели.

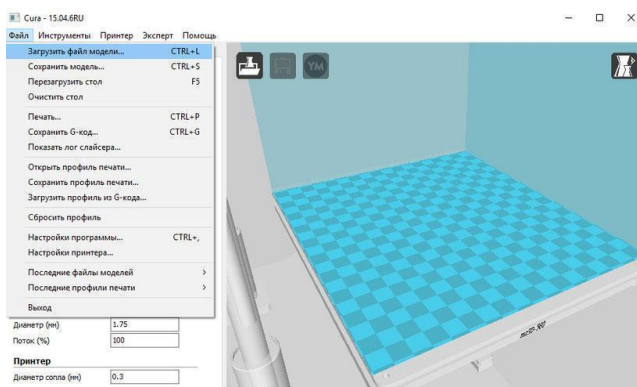


Рисунок 161. Слайсер Cura

Выбираем созданную вами STL-модель в рабочей папке.



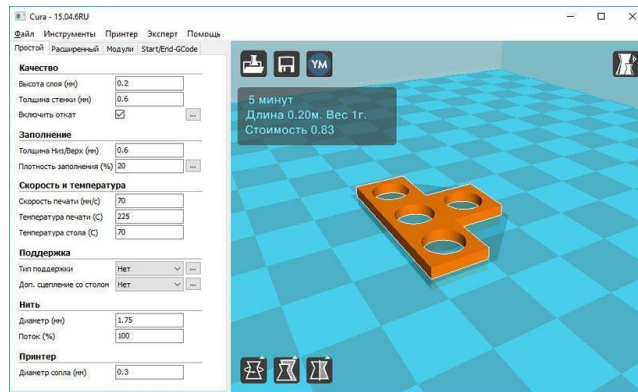


Рисунок 162. Загружаем нашу модель

Проверяем выставленные параметры, в соответствии с рисунком. Наш пластик PETG, для него температура печати 225, стола 70. Остальные параметры не трогаем (но с удовольствием обсуждаем в свободное время).

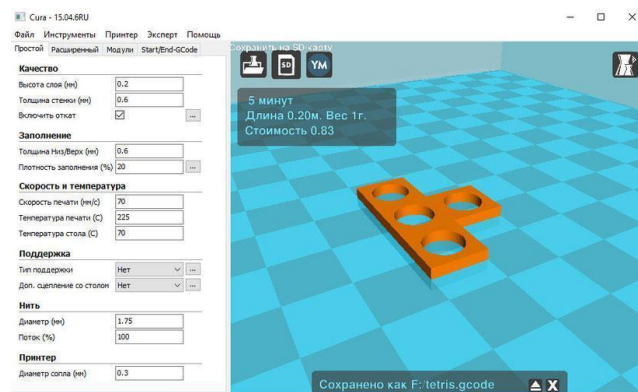


Рисунок 163. Настройка печати

Сохраняем файл на SD карту для принтера. Если карта уже в компьютере, можно просто нажать на кнопку SD, если файл сохраняется для последующей работы, используем меню Файл - Сохранить gcode.

Все, карту памяти с файлом можно вставлять в принтер, выбирать Печать с SD и запускать процесс нагрева и печати.

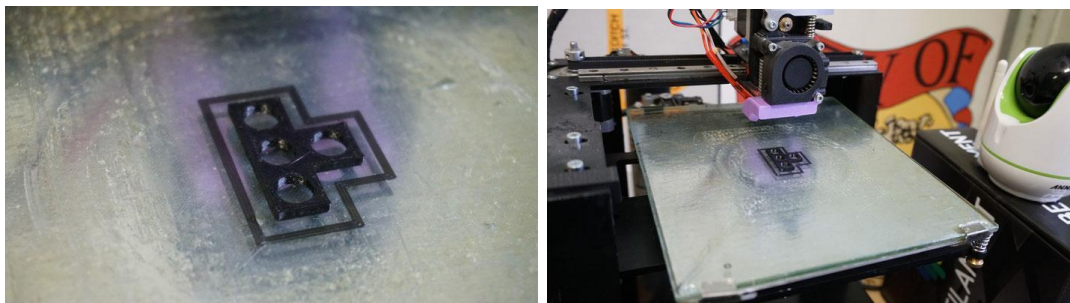


Рисунок 164. Печать изделия

### 3. 3D-печать и работы со станками с ЧПУ

#### 3.1. Repetier Host: использование слайсера для 3D-печати

Repetier Host [50] — это хост для 3D-принтеров и программа для нарезки, которая преобразует файлы STL в G-код с использованием одного из трех известных механизмов слайсинга: Cura, Slic3r или PrusaSlicer. Он также имеет множество функций, направленных на улучшение вашей настройки 3D-печати. В данной главе описаны функции и некоторые важные настройки Repetier Host.



Рисунок 165. Интерфейс Repetier Host

Repetier Host обладает довольно широким набором функций, полезный в процессе 3D-печати.

- **Скоростное рисование (Velocity painting):** эта функция позволяет вам отпечатывать изображения на поверхности, чтобы она выглядела так, как будто был вырезан узор.
- **Ручное управление (Manual control):** Repetier Host изначально создавался как хост-программа для 3D-принтеров, поэтому ручное управление является важной и ведущей функцией данного программного обеспечения.
- **Поддержка нескольких экструдеров:** Repetier Host позволяет настроить до 16 экструдеров для ваших 3D-принтеров, добавляя простые возможности для работы с несколькими материалами.
- **Приложение Repetier Informer:** это приложение работает вместе с Repetier Host, чтобы информировать вас по телефону о ваших текущих 3D-печатах.
- **Пользовательский Repetier:** для продвинутых пользователей, использующих Repetier Host в области 3D-печати и владельцев ферм принтеров [51], Repetier предлагает индивидуальную версию программного обеспечения, оптимизированную под нужды пользователя.



## Начало работы

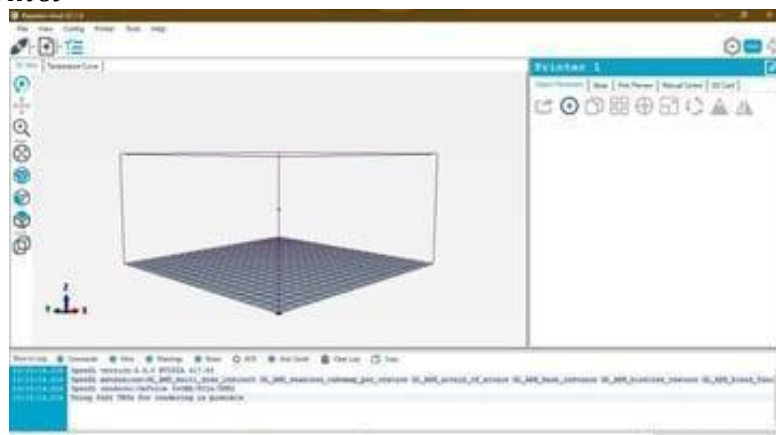


Рисунок 166. Начало работы Repetier Host

Пользовательский интерфейс довольно удобный и простой, и большинство настроек доступны на начальном экране.

Программное обеспечение доступно для Windows, MacOS и Linux, а установочные файлы можно бесплатно загрузить на веб-сайте Repetier - <https://www.repetier.com>.

После установки, пользователя встречает бело-синий пользовательский интерфейс. Рабочее окно разделено на два столбца: один для предварительного просмотра области печати (включая элементы управления слева), а другой — для параметров печати и других конфигураций. В верхней части окна находится строка меню, содержащая большинство настроек слайсера.

Во время нарезки, пользователь может выбрать один из различных механизмов нарезки. Всякий раз, когда вы переключаетесь между слайсерами, вам предоставляется набор настроек, характерных для каждого механизма слайсера. В разделе экспорта есть два варианта извлечения файла: отправить G-код непосредственно на подключенный принтер, либо сохранить его на SD-карту.

### *Основы работы на Slicer*

Ниже приведен ряд функций, считающийся распространенным во всех слайсерах и используемым по умолчанию.

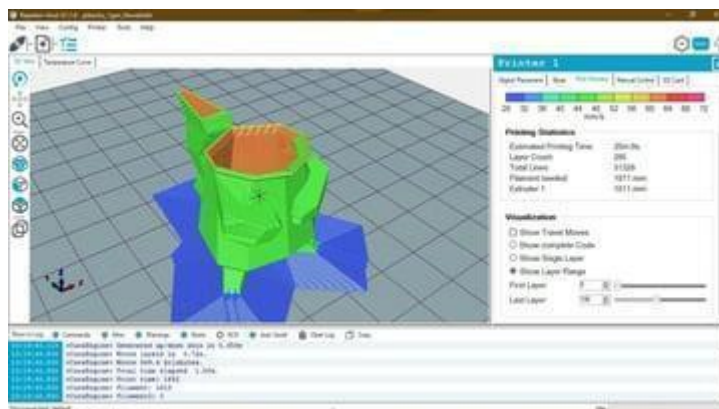


Рисунок 167 .Интерфейс Slicer

*Представление в 3D-виде.* Это - пространство, где вы можете предварительно просмотреть модель на рабочей пластине вашего принтера и внести коррективы в масштаб, размещение и т. д. С правой стороны, на вкладке «Размещение объектов», вы можете выполнить дополнительные действия, чтобы подготовить модель к нарезке.

*Функция самого слайсера.* Из всех доступных слайсеров, именно эта функция Repetier Host выгодно выделяет его среди других альтернатив. Под вкладкой слайсера в правой части окна находится набор элементов управления. Пользователь может выбрать один из трех движков нарезки, каждый из которых обладает своим уникальным набором функций. Вы можете настроить параметры на микроуровне в каждом слайсере и сохранить конфигурации, чтобы была возможность повторно использовать настройку для быстрой печати без необходимости повторного ввода входных настроек.

*Предварительный просмотр печати.* После того, как вы разместили модель и нарезали ее в соответствии с вашими потребностями, можете посмотреть на получившийся результат. В окне предварительного просмотра есть возможность визуализировать, как модель будет напечатана в 3D - регулировать скорость принтера, траекторию движения сопла, обработку выступов (при наличии) и т.д. В случае, если предварительный просмотр обнаружил какие-либо проблемы и ошибки, их можно исправить до печати (Рисунок 168).



Рисунок 168. Предпросмотр

#### *Средство просмотра SD-карт*

3D-печать с SD-карты позволяет пользователям использовать любой компьютер для подготовки 3D-моделей. Данная функция позволяет пользователю хранить G-код на SD-карте, тем самым лишая пользователя необходимости подключать компьютер к 3D-принтеру непосредственно. В средстве просмотра SD-карт Repetier Host есть возможность просматривать большое количество информации, включая визуализацию любых файлов G-кода, хранящихся на карте, в средстве 3D-просмотра, а также редактировать и удалять файлы.

*Ручное управление.* Repetier Host начинался как хост-программа для 3D-печати, разработанная с нуля для ручного управления 3D-печатью. Ручное

управление подключенным 3D-принтером позволяет вам взаимодействовать с принтером в режиме реального времени, без необходимости заходить на принтер и менять настройки. В Repetier Host вы можете отправлять G-код во время печати, а также контролировать и контролировать температуру экструдера (ов) и нагреваемого стола.

### *Особенности Repetier*

Repetier снабжен графиком температуры и регистраторов, позволяющих точно определить любые несоответствия и ошибки, связанные с 3D-принтером (Рисунок 169):



Рисунок 169. Регистраторы Repetier

### *Температурная кривая*

Repetier Host является одним из наиболее удобных программ при работе с печатью моделей.. Он информирует пользователя о различных параметрах 3D-печати и визуализирует процесс в режиме реального времени. Также, пользователь может отслеживать температуру работы 3D-принтера с течением времени на выведенном на экран графике, который позволяет просматривать данные за последний час и следить за их значениями. Если во время печати произошел сбой, эта функция может стать крайне полезным инструментом для устранения неполадок.

### *Регистратор*

Регистратор — еще одна уникальная функция Repetier Host, которая ведет учет всех действий в программном обеспечении. Любое действие, выполняемое пользователем либо слайсером, регистрируется, и может быть просмотрено в дальнейшем. Пользователь может выделить любые ошибки или команды в регистраторе, изменяя либо удаляя их. Как и температурная кривая, эта функция особенно полезна в случае каких-либо непредвиденных ошибок. Поскольку Repetier Host постоянно находится в стадии разработки, регистрация действий и ошибок пользователей предоставляет разработчику ценную информацию для будущей разработки.

## Repetier Informer



Рисунок 170. Интерфейс Repetier Informer

Приложение Repetier Informer представляет собой удобный инструмент для удаленного слежения за процессом 3D-печати. Приложение способно отправлять push-уведомления на ваше мобильное устройство о ходе 3D-печати. Таким образом, пользователь будет немедленно уведомлен о внезапных сбоях при работе, при их наличии.

*3D визуализация.* В меню «Конфигурация» расположен выбор настроек, в которых пользователь может видоизменить пользовательский интерфейс Repetier Host. Опция «3D-визуализация» предоставляет широкий набор вариантов настройки внешнего вида пользовательского интерфейса. Пользователь имеет возможность установить цвета программного интерфейса и изменить внешний стиль предварительного просмотра модели (Рисунок 171). Такие же настройки можно применить к визуализации филамента, а также к освещению модели.



Рисунок 171 . Визуализация модели в Repetier

### *Инструменты для продвинутых пользователей*

Расширенные функции Repetier обладают рядом уникальных возможностей. Некоторые из этих функций требуют довольно глубокого изучения и продвинутого уровня пользования средой Repetier.



*Скоростное рисование (Velocity Painting).* Скоростное рисование — относительно новая функция Repetier Host. Инструмент является доступным в виде отдельного приложения. Инструмент скоростного рисования создает текстурированный 3D-объект на основе изображения.

Иными словами, пользователь может получить отпечаток изображения на своей 3D-модели. Инструмент печатает более темные области изображения со более низкой скоростью, а более светлые области печатаются с несколько скоростью относительно большей. Лучше всего это работает с объектами, напечатанными в режиме вазы [52].

*Группировка объектов.* Группировка объектов — еще одна удобная функция в Repetier Host, полезная для многоцветной печати. При многоцветной печати в памяти устройства обычно создается отдельный файл формата STL для каждого цвета, который может быть сложно разместить как единую модель на печатной платформе. С помощью этой функции вы можете группировать элементы одного цвета и назначать им отдельный экструдер. Подробную информацию о работе этой функции группировки объектов можно найти в документации Repetier Host [53].

*Редактор G-кода.* Редактор G-кода Repetier Host упрощает настройку команд. Начинающим пользователям, возможно, никогда не понадобится использовать эту функцию, однако в случае необходимости вручную отредактировать определенный раздел G-кода, Repetier Host может вам помочь. Редактор G-кода позволяет просматривать весь файл G-кода построчно и редактировать его в самой программе. Пользователь также может сразу визуализировать измененный файл G-кода.

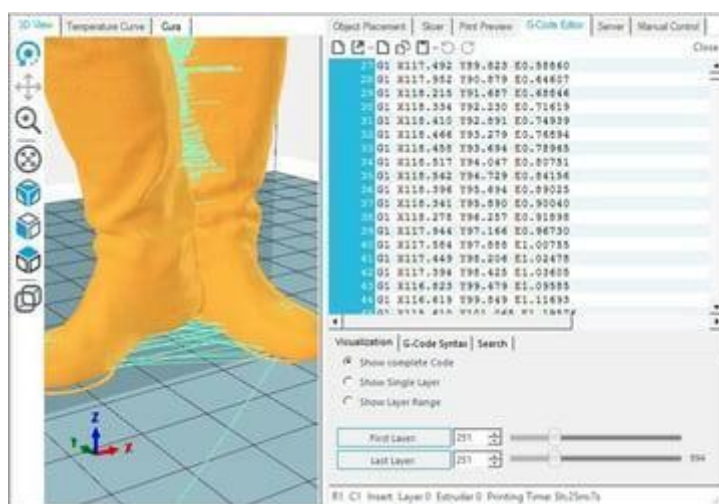


Рисунок 172. Редактор G-кода

В качестве информационной функции, редактор G-кода даже включает пояснения к различным командам. В то же время, на отдельной вкладке вы можете искать и заменять любые строки кода. Несмотря на ориентированность на опытных пользователей, визуализация редактируемого G-кода и объяснение команд делают его полезным для каждого пользователя 3D-слайсера.

*Bed-Height Mapping (Картирование высоты подложки).* Эта функция доступна лишь в случае использования прошивки Repetier с датчиком Z при работе с принтером. Как правило, то, что кажется плоской подложкой для 3D-печати, на самом деле не является идеально плоской рабочей пластиной. На поверхности рабочей пластины есть небольшие неровности или вмятины, которые могут вызвать ошибки при укладке первого слоя 3D-печати.

Инструмент картографирования высоты слоя наносит на карту весь слой с помощью зонда Z, а полученная карта высот показывает неровности поверхности и соответствующим образом калибрует первый слой. Это гарантирует, что первый слой всегда параллелен поверхности печати, что обеспечивает прочную основу для 3D-печати. Это удобная функция, которую можно использовать, если вы сталкиваетесь с постоянными проблемами с первым слоем ваших 3D-отпечатков.

*Полезные ресурсы по использованию Repetier.* На веб-сайте Repetier предоставлена полная документация программного обеспечения. Repetier Host изначально был представлен как бесплатный проект с открытым исходным кодом, однако, в 2014 году, разработчики закрыли исходный код [54]. Однако у программы имеется активное интернет-сообщество, а также обширный набор советов и информации на собственном веб-сайте Repetier.

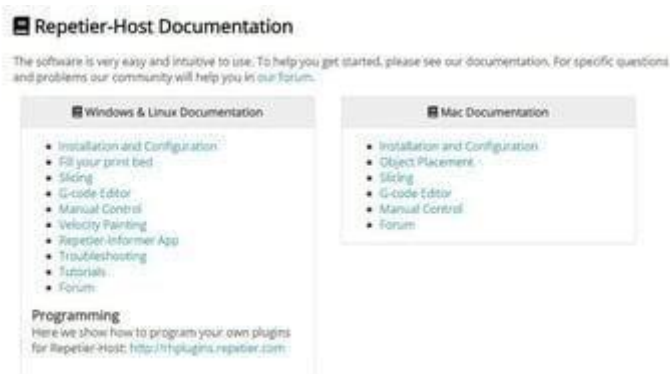


Рисунок 173. Документация программы

Полезные ссылки:

<https://forum.repetier.com/categories/repetier-host> - форум и полезные руководства по использованию программы;

<https://www.felixprinters.com/downloads/3.%20Software/2.%20Repetier-Host/2.%20Manuals/old/Tutorial%20-%20Printing%20with%20Repetier-Host.pdf> - инструкция и руководства по печати в Repetier.

<https://www.youtube.com/watch?v=ZFh70WeDJwI&t=102s> - учебные пособия Repetier lzk самостоятельного изучения 3D-моделирования.

<https://www.repetier.com/documentation/repetier-host/repetier-host-documentation-troubleshooting/> - руководства от разработчиков по устранению неполадок.

### *Иные альтернативные ресурсы для трехмерной печати.*

**Cura.** Cura — это популярное программное обеспечение для 3D-слайсеров, предоставляемое компанией Ultimaker. Repetier использует механизмы нарезки Cura в качестве одной из опций в своем собственном программном обеспечении. Одним из ключевых преимуществ использования Cura по сравнению с Repetier является его открытый исходный код. Cura также предоставляет библиотеку готовых к использованию плагинов, расширяющих функциональность слайсера. В Cura также есть целый набор экспериментальных функций, которые недоступны в программном обеспечении Repetier, а часто и в любом другом слайсере (Рисунок 1).

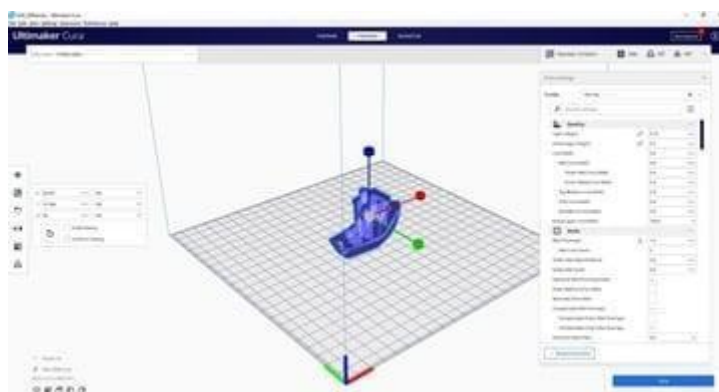


Рисунок 174. Интерфейс слайсера Cura

### **PrusaSlicer**

PrusaSlicer — еще один замечательный 3D-слайсер, разработанный и поддерживаемый компанией Prusa Research. Как и Cura от Ultimaker, этот слайсер имеет открытый исходный код и может использоваться бесплатно. Основное преимущество PrusaSlicer по сравнению с Cura и Repetier Host заключается в том, что он обеспечивает поддержку нарезки для печати смолой. Наряду с предыдущими аналогами, PrusaSlicer также владеет довольно активным сообществом.

В таблице 2 представлено описание ряда программ, предназначенных для 3D-моделирования:

Таблица 2. Популярные программы по 3D-моделированию

№	Наименование	Примечание	Интернет-ссылка
1	Компас 3D LT v.12	бесплатная, но не работает с форматами STL, OBJ, STEP, поэтому рекомендуется более продвинутая версия – 17 и выше	<a href="https://kompas.ru/kompas-3d-lt/about/">https://kompas.ru/kompas-3d-lt/about/</a>
2	Компас 3D v.19	платная, доступна образовательная лицензия или триал	<a href="https://edu.ascon.ru/main/download/freeware/">https://edu.ascon.ru/main/download/freeware/</a>



3	Autodesk Inventor v.20 и выше	бесплатна для образовательных учреждений	<a href="https://edu.ascon.ru/main/download/kit/">https://edu.ascon.ru/main/download/kit/</a>
4	Autodesk Fusion 360	бесплатна для обучающихся и преподавателей	<a href="https://www.autodesk.ru/education/free-software/featured?referrer=%2Feducation%2Ffree-software%2Ffeatured">https://www.autodesk.ru/education/free-software/featured?referrer=%2Feducation%2Ffree-software%2Ffeatured</a>
5	TinkerCad	бесплатно	<a href="https://www.autodesk.ru/products/fusion-360/students-teachers-educators">https://www.autodesk.ru/products/fusion-360/students-teachers-educators</a>
6	Ultimaker Cura	бесплатно	<a href="https://ultimaker.com/software/ultimaker-cura">https://ultimaker.com/software/ultimaker-cura</a>
7	Polygon 2	бесплатна, работает с 3D-принтерами Picaso	<a href="https://picaso-3d.com/ru/products/soft/polygon-2-0/">https://picaso-3d.com/ru/products/soft/polygon-2-0/</a>
8	Polygon X	бесплатна, работает с 3D-принтерами Picaso, нужна регистрация	<a href="https://picaso-3d.com/ru/techsupport/soft/designer-x/">https://picaso-3d.com/ru/techsupport/soft/designer-x/</a>
9	Slic3r	бесплатна	<a href="https://slic3r.org/download/">https://slic3r.org/download/</a>
10	Repetier Host	бесплатна	<a href="https://www.repetier.com/">https://www.repetier.com/</a>

### **3.2. Верстка проектов для работы со станками с ЧПУ**

Верстак Path используется для создания машинных инструкций для станков с ЧПУ из 3D-моделей FreeCAD. Верстка проектов позволяет изготавливать из них изделия на таких станках с ЧПУ, как: фрезерные, токарные станки, лазерные резак и т.д. Инструкции, полученные в результате верстки представляют собой набор G-кодов, управляющих станки с ЧПУ. В источнике [55] представлен общий пример моделирования траектории движения инструмента на токарном станке с ЧПУ.

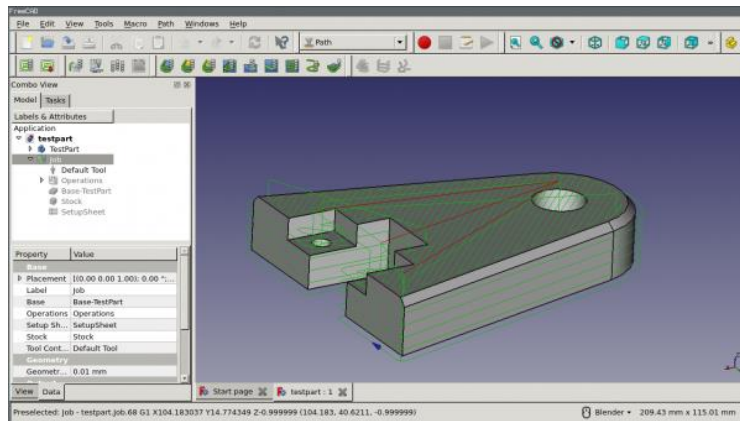


Рисунок 175. Интерфейс верстака FreeCAD Path

Рабочий процесс создания инструкций в верстаке FreeCAD Path выглядит следующим образом:

- 3D-модель - это базовый объект, обычно созданный с использованием одного или нескольких верстаков *Part Design*, *Part* или *Draft*.
- В верстаке *Path* создается *Задание*. Оно содержит всю информацию, необходимую для генерации G-кода для обработки на станке с ЧПУ: здесь определен материал, станок имеет определенный набор инструментов и выполняет команды, контролирующие скорость и перемещения (обычно G-Code).
- *Инструменты* выбираются в соответствии с требованиями *Рабочих Операций*.
- Операции обработки задаются с использованием *Контуров* и *Вырезов*. Эти операции обработки используют внутренний диалект FreeCAD G-Code, независимо от станка с ЧПУ.
- Экспортируйте задание в G-код, соответствующий вашему станку. Этот шаг называется «постобработка», для данного шага доступны разные постпроцессоры.

### ***Основные понятия в верстаке Path***

Верстак Path генерирует G-код, определяющий траектории движения фрезы, необходимую для фрезерования проекта, представленного 3D-моделью на FreeCAD диалекте G-Кода [56], который впоследствии переводится на соответствующий диалект для целевого контроллера ЧПУ, путем выбора соответствующего постпроцессора.

G-код генерируется из директив и операций, содержащихся в *Задании* на обработку. *Job Workflow* перечисляет их в порядке выполнения. Список заполняется путем добавления Path Operations, Path DressUps, Path Partial Commands и Path Modifications из Path меню или кнопок GUI.

Верстак Path включает в себя *диспетчер инструментов* (библиотека и таблица инструментов), *инструменты проверки G-кода* и *симуляции обработки*, который, в свою очередь, содержит *постпроцессор* и позволяет импортировать и экспортировать шаблоны заданий.

### ***Полезные ресурсы по изучению верстака Path;***

1. Path Walkthrough for the Impatient: краткое руководство для ознакомления с верстаком Path [57] .
2. FreeCAD Path: Custom paths with Python: плейлист с серией из пяти видеороликов на английском языке от Sliptonic. В серии обучающих видеороликов подробно описана работа с верстаком Path [58].
3. FreeCAD CAM Path Workbench: плейлист с серией из 7 видеороликов на английском языке от CAD CAM Lessons [59] .
4. FreeCAD CAM CNC — плейлист с серией из 8 видеороликов на английском языке от CAD CAM Lessons [60] .

Название модуля	Тема	Кол-во часов
Интернет вещей (IoT) на основе Arduino	Arduino и шаговые двигатели, сервоприводы.	4
	Сетевой обмен с помощью Arduino	4
	Arduino и карты памяти, светодиодные матрицы, радиочастотная идентификация (RDIF)	4
	Arduino и датчики расстояния, передача данных в инфракрасном диапазоне	4
	Arduino и Bluetooth, радиоуправление, беспроводные радиомодули, работа с USB-устройствами	6
	Умный дом на Arduino	12
		<b>34</b>

## Цели обучения

### Модуль 1. Интернет вещей (IoT) на основе Arduino

1.1. Arduino и шаговые двигатели, сервоприводы - Ознакомить с подключением двигателя постоянного тока и его управлением (построить с ним вентилятор). Ознакомить с подключением сервомотора и его управлением (построить с ним секундомер).

1.2. Сетевой обмен с помощью Arduino - Сформировать навыки форматирования веб-сайтов с помощью HTML. Сформировать навыки отображения веб-сайтов с помощью Arduino и Ethernet Shield. Ознакомить с серверным программированием.

1.3. Arduino и карты памяти, светодиодные матрицы, радиочастотная идентификация (RDIF) - Сформировать навыки поиска и установки дополнительных библиотек. Сформировать навыки работы с картами памяти: создавать, редактировать и удалять файлы на MicroSD карте. Сформировать навыки работы со светодиодными матрицами: подключение модуля NeoPixel на базе WS2812B светодиодов к Arduino. Сформировать навыки работы с модулем радиочастотной идентификации RFID.

1.4. Arduino и датчики расстояния, передача данных в инфракрасном диапазоне - Сформировать навыки работы с датчиками расстояния и передачей данных в инфракрасном диапазоне

1.5. Arduino и Bluetooth, радиуправление, беспроводные радиомодули, работа с USB-устройствами - Сформировать навыки:

подключения к Arduino Bluetooth модуля HC-06 и дистанционного управления и/или передачи данных с его помощью, управления устройствами с телефона, работы с USB-устройствами

1.6. Умный дом на Arduino - Сформировать навыки по созданию систему удаленного управления домом по смс, используя GSM/GPRS Shield.

Образовательная и соревновательная робототехника рассматривается педагогикой как новая, актуальная педагогическая технология, которая находится на стыке перспективных областей знания: механика, электроника, автоматика, конструирование, программирование и технический дизайн. образовательная робототехника отражает все грани научно-технического творчества в настоящее время и является уникальной образовательной технологией, направленной на поиск, подготовку и поддержку нового поколения молодых исследователей с практическим опытом командной работы на стыке перспективных областей знаний.

Робототехника в образовании – это междисциплинарные занятия, интегрирующие в себе науку, технологию, инженерное дело, математику, основанные на активном обучении учащихся.

Робототехника основывается на использовании элементов техники или робототехники, которая развивает универсальные учебные действия. Ученики получают ознакомительные технические знания, которые развивают инженерное и логическое мышление. Это необходимо для знакомства ученика с техникой (т.к. в информационном обществе ребенок с самого раннего возраста знакомится с техникой). Инженерное мышление – это системное творческое техническое мышление, позволяющее видеть проблему целиком с разных сторон, видеть связи между ее частями. Инженерное мышление представляет собой вид познавательной деятельности, которая направлена на исследование прогрессивных технологий и повышения качества техники. Инженерное мышление занимается решением конкретных задач и целей с помощью технических средств. Научно-техническое творчество есть результат в области науки и техники, а также отличается уникальностью и оригинальностью. Поэтому ученик должен получать представление о стартовом моделировании, как о научно-техническом творчестве. В начальном звене робототехника состоит из конструирования, начального технического моделирования и элементарного программирования.

Цели робототехники, как образовательной деятельности в преподавании:

1. Показать возможности робототехники, как одного из главных направлений научно-технического прогресса.
2. Показать роли робототехники в проектировании.
3. Развитие экспериментальных умений и навыков.
4. Углубление и расширение предметного знания.

Мотивация и познавательный интерес детей при изучении робототехники.

«Робототехника» может мотивировать учащихся на занятие инженерными технологиями, позволяет овладеть навыками начального технического конструирования, изучение понятий конструкции и основных свойств, а также приобрести навык взаимодействия в группе.

Изучение основ робототехники будет способствовать развитию у учащихся

конструкторского мышления и умения решать практические задачи, кроме того это очевидно привлекательная образовательная среда, вдохновляющая детей к новаторству через науку, технологию, математику, поощряющая думать творчески, анализировать ситуацию, критически мыслить, применять свои навыки для решения проблем реального мира.

Робототехника имеет ряд преимуществ: большое разнообразие деталей, яркость, безопасность, долговечность, свобода в выборе тематик. Занятия робототехникой помогают детям воплощать в жизнь свои задумки, строить и фантазировать, увлеченно работая и видя конечный результат. В ходе изучения курса робототехники ученики развивают мелкую моторику, логическое и системное и инженерное мышление, практические навыки сборки и построения модели, получают знания о простых механизмах, а также творческие способности. Данные качества очень полезны для ребенка, даже если его будущая карьера не будет связана с техническими науками. Занятия робототехникой соприкасается сразу с несколькими учебными предметами, таких как: физика, информатика, математика. Дети могут, при помощи своих развитых способностей, решать поставленные перед ними нестандартные задачи. Так как робототехника метапредметна, родители в будущем получают развитую и эрудированную личность, которая способна анализировать и логически рассуждать, используя знания из различных областей. Робототехника быстро становится неотъемлемой частью учебного процесса, так как она поощряет детей мыслить творчески, анализировать ситуацию и применять критическое мышление для решения реальных проблем.

На занятиях по робототехнике дети:

- получают математические знания о счете, форме, пропорции, симметрии;
- расширяют свои представления об окружающем мире – об архитектуре, транспорте, ландшафте;

- развивают мелкую моторику, стимулирующую в будущем общее речевое развитие и умственные способности;
- развивают пространственное воображение;
- развивают внимание, память, способность сосредоточиться;
- развивают творческие способности, эстетическое восприятие;
- развивают логическое и аналитическое мышление (умение мысленно разделить предмет на составные части и собрать из частей целое);
- занятия по робототехнике учат детей работать в коллективе и находить совместное решение задач.

Проектно-ориентированное обучение – это систематический учебный метод, вовлекающий учащихся в процесс приобретения знаний и умений с помощью широкой исследовательской деятельности, базирующейся на комплексных, реальных вопросах и тщательно проработанных заданиях.

Методы, которые могут усовершенствовать занятие - это соединение исследовательского метода обучения, игрового метода и обучения в сотрудничестве.

В процессе игровой деятельности формируются не только коммуникативные, но и личностные и познавательные универсальные учебные действия (учебно-познавательный интерес к новым знаниям и способам решения поставленной задачи, умение адекватно воспринимать оценку педагога). Дети адекватно реагируют на ошибки, которые они допускают, и тут же решают эти проблемы.

Задачи робототехники:

- развитие общеучебных, коммуникативных навыков;
- формирование элементов информационной культуры через навыки информационного видения явлений и процессов окружающего мира при создании моделей (текст, диаграмма, рисунок, модель, конструктора);
- формирование навыков программирования; развитие образного и логического мышления, мелкой моторики руки и творческих способностей;
- развитие технического мышления, формирование умения самостоятельно решать поставленную задачу через реализацию метапредметных связей;
- развитие коммуникативных умений и способность строить комфортные коммуникативные отношения в малой группе и коллективе. создание условий для творческого развития личности ребенка;
- развитие мотивации личности к познанию и творчеству;
- обеспечение эмоционального благополучия ребенка;
- приобщения учащихся к общечеловеческим ценностям; профилактика асоциального поведения.



## **Игровые технологии**

Занятия по основам робототехники для детей младшего и среднего школьного возраста могут иметь игровой характер. Игра — это ведущий вид деятельности младшего школьника. Именно в игре происходит тренировка многих важных жизненных навыков, формируются черты характера. В начальной школе игра является главным способом получения ребенком социального опыта. Ведущая роль в этом принадлежит сюжетно-ролевым и деловым играм. Их форма проведения предполагает импровизированное разыгрывание определенной ситуации. Причем, в большинстве случаев, ребенок в ходе данных игр выполняет роли взрослого человека. Зачем нужно проводить ролевые и деловые игры для детей младшего школьного возраста? Какие ролевые игры можно проводить? Включаясь в такую деятельность, школьникам приходится:

- выражать разные гражданские позиции;
- организовывать или влиять на поступки других участников;
- устанавливать коммуникативные связи, налаживать контакты;
- решать выполнять различные социальные роли;
- возникшие по ходу игры конфликтные ситуации.

В процессе игры устанавливается непринужденная обстановка. Ее органичный ход включает естественное командование и подчинение, которое дети не воспринимают агрессивно. Общение в игре спонтанное, школьникам приходится вырабатывать такую модель поведения, которая предполагает формирование следующих коммуникативных умений:

- готовность к сотрудничеству;
- толерантность;
- терпимость к чужому мнению;
- умение вести диалог;
- умение находить компромиссное решение.

В процессе игры между детьми устанавливаются такие отношения, которые они потом воспроизводят в реальной жизни. Ребята обучаются таким приемам и правилам общения, которые в дальнейшем смогут использовать в сходных по форме жизненных ситуациях. Проигрывание ситуации несколько раз дает возможность школьникам поменяться ролями, испробовать другие варианты поведения, провести рефлексию деятельности.

## **Работа по технологическим картам**

Работу по технологическим картам можно отнести к демонстрационным методам, которые реализуют принцип наглядности обучения (условно-графическая наглядность). Демонстрационные методы активизируют у учащихся сенсорные и мыслительные процессы, облегчая усвоение учебного материала. Сочетание демонстраций со словом педагога делает обучение более доходчивым. Рассматривая технологическую карту, учащиеся сами

устанавливают, с чего начинать, из каких элементов выполнена модель, какие приготовить детали и какими приемами осуществить технологические операции. На технологической карте размещают схему конструкции, сведения о необходимых деталях, словесные инструкции в краткой форме. Таким образом, в технологической карте имеются все необходимые данные для самостоятельной работы учащихся: процесса конструирования и самоконтроля. Для полноценного и эффективного использования технологических карт необходимо знать ряд принципов и положений, которые помогут работать с ней.

Технологические карты классифицируются на группы:

- печатные пособия (карточки);
- проекционный материал (слайды).

### **Основные этапы разработки робототехнического проекта**

- Обозначение темы проекта
- Цель и задачи представляемого проекта
- Разработка механизма на основе конструктора Лего модели EV3
- Составление программы для работы механизма в среде Lego Mindstorms.
- Тестирование модели, устранение дефектов и неисправностей.

### **Примерные требования к содержанию презентации проекта**

- Название проекта
- Цель проекта: создание модели робота (процесса) ...

Задачи:

- разработать проект модели
- собрать проект (модель)
- составить программу
- проверить работу проекта (модели)
- настроить технические параметры

Технический состав проекта: основные используемые элементы (блоки микрокомпьютера, сервомоторы, датчики, оригинальные механизмы и узлы)

- Фото- и видео-материалы (включая скриншот программы)
- Анализ проделанной работы, выводы
- Авторы проекта и наставники

### **Подготовка рабочего пространства/подготовка к уроку**

Несмотря на то, что спрос на кружки робототехники все-таки вырос, современные дети проявляют недостаточный интерес к робототехническим секциям дополнительного образования. Отчасти отсутствие интереса вызвано устаревшей материальной базой учреждений. Сейчас необходимо создавать

иные условия развития образования, которые позволят внедрять новые образовательные технологии.

### **Образовательная робототехника**

Преподавание робототехники в школе будет проводиться в рамках уроков STEM. Однако 45 минутного урока может оказаться недостаточно для того, чтобы полностью раскрыть тему, увлечь детей и предоставить им возможность для собственного творчества и экспериментов. Поэтому робототехника может выступать и в качестве дополнительного образования детей после уроков.

Дети не проявляют достаточного интереса к изучению инженерных навыков отчасти из-за старого оборудования, отсутствия квалифицированного преподавателя или системной учебной методики. Поэтому оснащение кабинета современными системами — одна из важнейших задач, которая стоит при организации уроков.

Оборудование должно отвечать сразу нескольким требованиям:

- 1) Оборудование для робототехники должно быть высокого качества и современным;
- 2) Подходить для обучения детей разного возраста и уровня владениями навыками;
- 3) Отвечать всем требованиям безопасности;
- 4) Позволять применять полученные знания на практике;
- 5) Учитывать современные тенденции развития робототехники.

Важно учесть, что современная робототехника продвинулась далеко за рамки привычного конструктора. В образовательном контексте эта область охватывает сразу несколько дисциплин, таких как информатика, физика, химия, естественные науки, а также дополнительные навыки: программирование, пайка, конструирование, черчение и так далее. Все это органично совмещается через проектную деятельность. При подготовке проекта дети учатся ставить цели и решать задачи по их достижения, работают в команде и учатся взаимодействию. Поэтому для преподавания робототехники нужно серьезно подойти к вопросу подготовки материальной базы для выполнения заданий. Дети могут заниматься:

- 1) 3D моделированием и прототипированием с помощью 3D принтера.
- 2) Сборкой механизмов и пайкой.
- 3) Освоением базовых языков программирования и закреплением навыков на практике и тд.

К слову, несмотря на то, что обычно организация уроков технологии происходит в разделенных по половому признаку классах, занятия

робототехникой могут проходить совместно. В кружках робототехники дети обычно занимаются в малых группах — до 14 человек, поэтому целесообразно при возможности разделить детей на подгруппы.

#### Включение педагогов в процесс

Тем не менее, просто поставить технику на стол будет недостаточно. Какой бы современной она не была, без участия педагога, дети рано или поздно потеряют интерес к происходящему. Важно, чтобы учитель не просто был рядом, а сам действительно интересовался процессом. Увлеченный учитель не менее важен для занятий робототехникой с детьми.

Самое главное — не нужно учить детей тому, что им никогда не пригодится. Собирать примитивный конструктор, написать базовую программу и наблюдать за тем, как конструкция двигается — уже в некотором смысле анахронизм. Работа с электроникой, изучение механизмов движения, пайка, машинное обучение — примеры навыков, которые пригодятся детям, которые планируют связать свою жизнь с инженерными специальностями.

Образование детей сегодня должно заключаться во всестороннем развитии. По итогу они должны уметь ставить цели, комплексно подходить к решению самых разнообразных задач, обладать полезными прикладными навыками и уметь адаптироваться в переменчивых условиях современных реалий.

Для успешной реализации программы необходимы:

- Компьютерный класс – 8-12 мест для учащихся – на момент программирования робототехнических средств, программирования контроллеров конструкторов, настройки самих конструкторов, отладки программ, проверка совместной работоспособности программного продукта и модулей конструкторов;
- от 5 базовых наборов конструкторов;
  - зарядное устройство для конструктора – 3 шт.;
  - программный продукт – по количеству компьютеров;
  - поля для проведения соревнования роботов – 4 шт.;
  - стол для проведения соревнований;
  - мультимедиа (проектор или интерактивная доска).



Рис. Кабинет робототехники

### Кадровое обеспечение

Дополнительную общеразвивающую программу реализуют педагоги дополнительного образования, с образованием в области информатики и программирования либо прошедшие курс переподготовки

## Информация про спортивную робототехнику

Олимпиады, конференции и соревнования — это способы проверить свои способности. Как правило, задания на таких мероприятиях выше уровня школьной программы и требуют дополнительно творческого подхода. Особенно, если это касается такой достаточно нестандартной области, как робототехника.

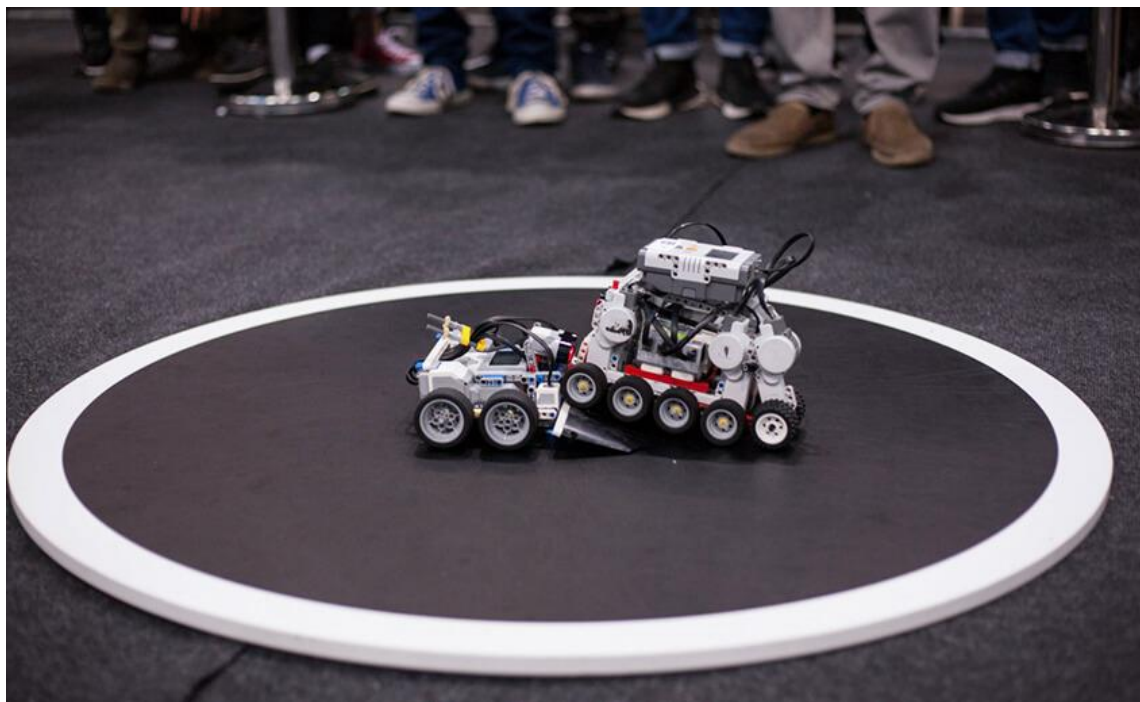


Рисунок 6. Робо-сумо

Соревновательная робототехника — это прикладное направление, в основе которого практическая работа конкурсантов. С целью тестирования возможностей роботов и их разработчиков придумывают множество разных заданий. Например, на протяжении многих лет не угасает интерес к боям роботов. И на то есть ряд причин. В первую очередь, это открытость мероприятия — принять участие в нем может любой, кто способен создать своего бойца.

Кроме этого, на таких мероприятиях можно встретить достаточно много единомышленников и при необходимости найти наставника-практика. Нельзя рассматривать бои роботов исключительно в развлекательном контексте. За зрелищностью всегда стоит длительная исследовательская и кропотливая практическая работа.

Для многих операторов мобильный робот, собранный из базовых элементов, становится первым серьезным устройством, а тестирование в формате боя помогает выявить серьезные недостатки и работать над усовершенствованием функций.

В регламенте четко указано, что роботы не могут повреждать конструкцию и системы робота-оппонента, то есть нельзя его сжигать, обливать чем-нибудь, расстреливать разнообразными веществами и так далее.

Пример регламента по робо-сумо:

- Сделать робота низким и достаточно тяжелым. Высокие и легкие конструкции легко лишит равновесия и опрокинуть.
- Убедиться в том, что колеса достаточно хорошо прорезинены. Чем лучше сцепление с поверхностью, тем больше шансов на успех.
- Обязательно выбрать правильное соотношение скорости к крутящему моменту и рассчитать энергоснабжение, чтобы робот спокойно выдержал все раунды боев.

Ниже список разных видов соревнований по робототехнике: Кегельринг — за определенное время робот должен вытолкнуть кегли за пределы ринга, но при этом сам не должен нарушать очерченные границы.

Лабиринт — суть ясна из названия: автономный робот должен проехать от точки «А» в точку «Б» по лабиринту, минуя разнообразные препятствия.

Лестница — соревнование подразумевает, что робот может преодолевать разные подъемы.

Траектория — робот должен добраться до конечной точки назначения, соблюдая указанную на пути траекторию.

Футбол — зрелищный вид соревнований, когда команды роботов или самостоятельные единицы, играют в футбол, то есть забивают противнику как можно больше голов.

Прикладная робототехника — серьезное мероприятие, когда робототехники в команде или по одному продумывают систему для решения реально существующих производственных задач.

Зачастую соревнования проводят в формате фестивалей, сочетая несколько заданий.

Как правило, принять участие в соревнованиях может любой желающий, проходящий по возрасту. Напомним, что соревнования по робототехнике проводятся для школьников и студентов, есть ряд мероприятий для самых маленьких. В конкурсах в зависимости от условий могут принимать участие как команды от 2-х человек, так и одиночки. Каждому соревнованию предшествует этап подготовки и, как правило, участники готовятся к заранее определенному турниру: некоторые подразумевают, что конкурсанты принесут готовые модели на конкурс, другие — сборку модели на месте.

### **Список крупнейших соревнований по робототехнике в Казахстане:**

1. Roboland
2. КазРоботикс
3. Мобильная робототехника WorldSkills
4. World Robot Olympiad
5. First Lego League Challenge
6. First Tech Challenge
7. First Global Challenge



8. VEX Robotics Competition
9. Robotek Grand Tournament
10. Almaty Tech Cup

### **Знакомство с спортивной робототехникой**

Спортивная робототехника - это где команды знакомятся с основами инженерного дела, исследуют проблемы касающегося регламента турнира на примере реального мира, учатся проектировать и программировать, а также создают уникальные решения с помощью деталей LEGO и робототехнических наборов.

Спортивная робототехника вдохновляет ребят на эксперименты, развивая в них уверенность в себе, критическое мышление и навыки проектирования средствами практико-ориентированного обучения.

Тренерская работа при подготовке к соревнованиям должна включать следующие виды деятельности:

1. Формирование состава команды;
2. Специальная подготовка к конкретным соревнованиям;
3. Морально-психологический настрой команды на предстоящие соревнования.

### **Формирование состава команды по методике VEX Robotics**

Алгоритм формирования состава команды:

1. Составление расписания
2. Регулярные собрания команды
3. Распределение ролей в команде
4. Разработка уникальной идентификации команды
5. Приобретение, хранение и использование оборудования
6. Коллективный поиск решений

### **Составление расписания**

Чтобы обеспечить эффективность работы команды, необходимо разработать соответствующее расписание и придерживаться его. Рационально составленный график способствует эффективности учебного процесса, снижению и устранению перегрузки учащихся, повышению работоспособности учащихся и педагогов, оптимальному использованию учебных помещений учебного заведения.

Вот несколько вопросов, на которые ваша команда должна найти ответы на одном из первых собраний.

- Насколько регулярно и где будет собираться команда?
- Сколько времени остается до первых соревнований?

Важно отметить то, что частое собрание может способствовать к снижению эффективности работы команды. Это из самых первых аспектов, которые должны решаться на первом собрании. Некоторые команды собираются один или два раза в неделю на несколько часов, тогда как другие команды собираются более регулярно, и каждая встреча длится довольно долго.

### **Собрания команды**

Важно, чтобы команды правильно строили свои собрания и концентрировали внимание на разработке плана реализации проекта. В ходе первого или второго собрания исключительно рекомендуется разработать перечень задач и сроков выполнения. Запишите задачи и сроки, а также назначьте члена команды (капитан), который будет контролировать рабочий процесс команды.

### **Распределение ролей в команде**

Чтобы обеспечить продуктивность команды, очень важно распределить роли между всеми ее членами. Выбор участниками команды определенных ролей помогает команде функционировать более эффективно и обеспечивает вовлеченность в процесс всех участников команды.

Во время занятий некоторые роли могут исполняться несколькими детьми одновременно. Например, если упражнение подразумевает работу в парах, роли конструктора и программиста могут играть сразу по двое участников команды. Вот несколько ролей, которые ребята могут распределить между собой:

**Фото/видео репортер команды.** Запечатлевает путь команды в течение сезона, делая фотографии или снимая видео. Собранные материалы могут быть использованы при создании командного плаката (если плакат входит в обязательные требования).

**Специалист по поиску деталей Lego.** Находит детали Lego, необходимые для каждого этапа сборки.

**Тренер.** Исходя из целевых результатов занятий, помогает участникам команды выстраивать процесс занятий и освоения новых знаний.

**Специалист по материалам.** Собирает материалы, необходимые для занятия, и убирает все на место в конце занятия.

**Конструктор.** Собирает модели из деталей LEGO, следуя инструкциям по сборке.

**Капитан команды.** Информировывает тренера о прогрессе команды. Обеспечивает выполнение заданий в рамках каждого занятия.

**Программист.** Работает с устройством (ноутбуком, планшетом и т.п.) и создает программы в приложении.

## 1. Основы запуска IT стартапа

### 1.1. Цифровые ресурсы для развития стартапа

Главными задачами стартапа в современном мире являются решение важных проблем клиентов, построение эффективного продукта, налаживание маркетинга, продаж и быстрое масштабирование. Однако, начиная с самых ранних этапов, у подавляющего большинства стартапов возникает необходимость в поиске инвесторов [1].

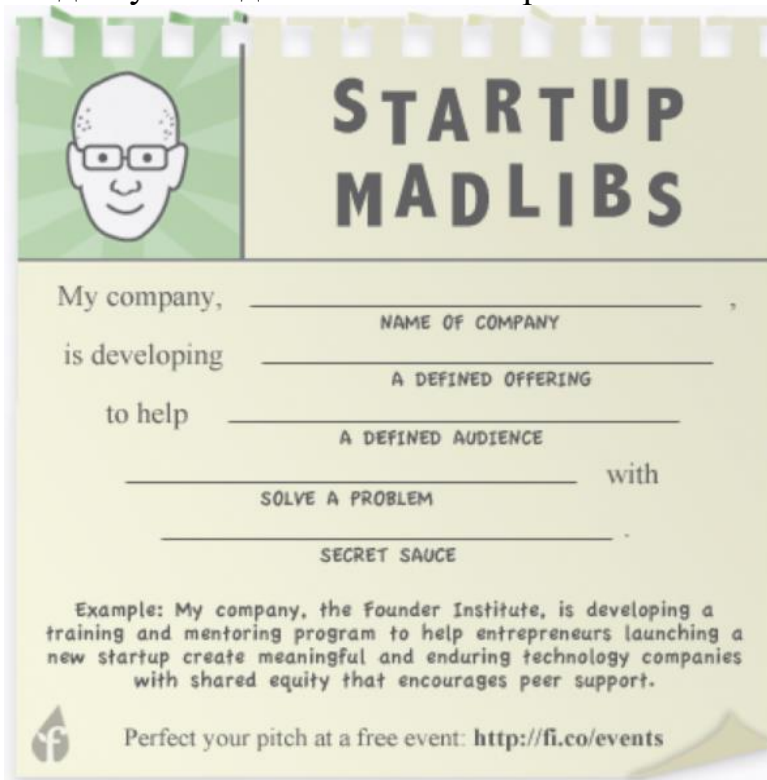
Для достижения успеха в поиске инвестора, стартапу необходим бизнес-план, нужно знать, какой объем средств понадобится для развития компании и на что они будут потрачены после получения. Именно поэтому, данная глава будет посвящена поиску инвесторов и привлечению капитала для развития бизнеса.

Читательский портал *AppCraft* предлагает своим пользователям несколько практических советов по привлечению инвестиций в свой стартап [2]:

Прежде чем начать задумываться о привлечении инвестора в стартап, начинающему предпринимателю нужно проделать огромный объем работы. Особенно, если цель – привлечь инвестиции из источников за пределами семьи, друзей и коллег.

Для начала, необходимо разработать понятную и цепляющую презентацию своей идеи в одном предложении. Это – суть всей истории, всего объема информации о проекте, которую вы будете использовать, чтобы попытаться продать свою компанию людям.

Для ёмкого и краткого описания своего проекта, можете использовать шаблон от Адео Ресси (Рисунок 1). Он поможет основателям стартапов перевести идею в пять или шесть доступных для понимания строк:



The image shows a template card for a startup pitch, titled "STARTUP MADLIBS". The card has a green and yellow color scheme. At the top left, there is a simple line drawing of a man's face with glasses. The title "STARTUP MADLIBS" is written in large, bold, black letters. Below the title, there are several lines of text with blank spaces for user input, each followed by a small label in all caps. The text reads: "My company, \_\_\_\_\_, NAME OF COMPANY", "is developing \_\_\_\_\_, A DEFINED OFFERING", "to help \_\_\_\_\_, A DEFINED AUDIENCE", "\_\_\_\_\_ with", "SOLVE A PROBLEM", and "\_\_\_\_\_ SECRET SAUCE". At the bottom, there is an example sentence: "Example: My company, the Founder Institute, is developing a training and mentoring program to help entrepreneurs launching a new startup create meaningful and enduring technology companies with shared equity that encourages peer support." Below the example, there is a small logo and the text "Perfect your pitch at a free event: <http://fi.co/events>".

Рисунок 1. Шаблон от Адео Ресси

### Использование современных технологий для поиска инвесторов

Создайте профиль на **AngelList**. Опишите вашу компанию, членов команды и продукт, чтобы инвесторы и другие стартапы могли найти и узнать о вас.

Стартап должен понимать свой рынок, сегмент и чётко описывать преимущества своего продукта, бизнес-модели и подходов перед конкурентами].

Существует несколько основных этапов, которые проходит стартап [3]:

- *Problem/Solution Fit* — подтверждение проблемы и того, что предлагаемый продукт решит;
- *Minimum Viable Product (MVP)* — разработка рабочего прототипа с минимальным функционалом для решения проблемы; выпуск альфа-версии и тестирование ее недочетов;
- *Product/Market Fit* — подтверждение, что у продукта есть рынок сбыта (закрытая бета-версия продукта);
- *Scale* — масштабирование бизнес-модели (открытая бета-версия и выпуск потребительской версии продукта);
- *Maturity* — переход от стартапа к бизнесу.

Если всё это есть, то один из главных порталов в «реальность инвесторов» легко доступен в онлайн и называется [AngelList](#). Там стартап ждут тысячи инвесторов в поисках выгодных сделок, а ещё примеры успешных подходов к построению стартапа, финансированию — и полезное предпринимательское сообщество.

**AngelList** - это онлайн-платформа и база данных для инвесторов и предпринимателей. AngelList активно используют инвесторы из разных уголков мира, вкладывающиеся в различные сегменты, этапы развития и с разным размером чеков. Здесь крупнейшие венчурные фирмы и ангелы ищут и подтверждают интересные сделки.

Начинающие стартапы используют AngelList для поиска сотрудников, обмена опытом и привлечения капитала. Сумма инвестиции через AngelList составляет в среднем \$200 тыс. и доходит до \$1 млн.

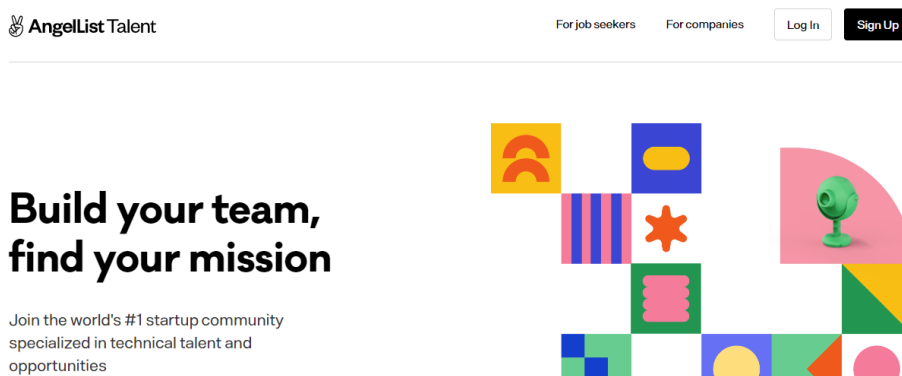


Рисунок 2. Сайт AngelList

Профиль стартапа в AngelList — это место, где можно делиться своими текущими потребностями в средствах, прошлыми успехами

в предпринимательстве и прошлыми инвесторами (по крайней мере теми, кто зарегистрирован на сайте).

#### *Как инвестируют на AngelList*

На раннем этапе AngelList был похож на Kickstarter: компании продвигали себя или свой продукт на сайте, а инвесторы присоединялись. AngelList зарабатывал деньги, взимая процент от капитала компании в обмен на листинг сделки. Но сайт быстро развивался и создал несколько других моделей инвестирования.

Среди них есть, например, интересная модель синдиката — это группа людей, которые вкладывают деньги в компанию. У каждого синдиката AngelList есть ведущий инвестор, который вкладывает определённую сумму денег, а затем обращается к другим инвесторам с просьбой о соинвестировании. Этот ведущий инвестор, как правило, пользуется уважением в отрасли, и другие выбирают соинвестирование и поддержку синдиката, основываясь на репутации ведущего инвестора.

Синдикаты обычно инвестируют от \$100 тыс. до \$1 млн при средней сумме инвестиций в \$200 тыс. Лучший способ подойти к синдикату — думать о лидерах синдиката как об ангельском инвесторе. Будет ли этот главный участник синдиката ценным инвестором и поможет ли он компании? Хороший лидер синдиката, как и хороший ангел, может принести большую ценность в компанию.

Если вы решили привлечь инвестиции от синдиката, внимательно изучите, кто в него входит, чем они могут быть полезны и какие у них уже есть инвестиции. В синдикате бывает до 70 участников, и все эти люди будут поддерживать ваш проект и помогать вам в случае инвестиции.

#### *Как построить присутствие на AngelList*

Зарегистрируйтесь с минимальными данными и изучите сайт. Обратите внимание на то, каким образом представляют себя популярные стартапы: описание продуктов и бизнеса, общение с посетителями. Это даст вам понимание культуры общения на AngelList. Особенно показательны профили с большим signal — это те, которые сайт выдает в вершине поиска.

Чаще всего, популярные профили стартапов на AngelList — продвинутые компании, с большим количеством сотрудников, но есть и множество исключений. Молодые стартапы могут попасть в топ с высоким «сигналом» благодаря детальным описаниям, большому количеству активных подписчиков, интересным дискуссиям в их профиле и интересу от инвесторов, которые пишут им на сайте.

Лучшие профили максимально информативны, но кратки: 2—3 предложения о продукте и бизнесе, короткое видео о продукте, доступ к презентации, профили основателей, правильные теги, запоминающиеся ответы на вопросы из анкеты AngelList (например, про культуру в компании) и чёткие цифры, характеризующие бизнес.

Совет: не бойтесь выдать стратегическую информацию. Пока ваш стартап не достиг хотя бы многомиллионных продаж, эти цифры не особо интересны даже вашим конкурентам.

### *Стратегии поведения на AngelList*

Одним из полезных опытов для начинающих может служить история компании Outbox. Свои шаги компания описывает следующим образом:

1. Пришли на площадку с уже собранной половиной от всего необходимого капитала. Это показало сообществу на платформе, что компания настроена серьёзно и уже имеет некоторый успех в привлечении средств, — значит, дело стоящее.

2. Команда подключила к своему профилю всех влиятельных людей на платформе. С кем были знакомы, кто был советником компании. Благодаря этому информацию их профиля увидело больше пользователей.

По итогу, компании Outbox удалось привлечь на AngelList \$8 млн.

Ваш потенциальный инвестор должен стать предметом исследования. Узнайте, кто этот человек, чем он занимается, во что инвестирует. Найдите с ним какие-то общие интересы в бизнесе. Например, если вы занимаетесь разработкой игр, на AngelList вы сможете найти огромное количество людей, которые занимаются играми.

AngelList – далеко не единственная площадка для привлечения потенциальных инвесторов. Аналогичным ресурсом, который пользуется большой популярностью, является платформа *Crunchbase* (Рисунок 3). Этот сайт позволяет найти конкретную компанию, человека или инвестора и посмотреть детально, кто во что и сколько средств вложил. Таким образом, вы составите представление о компаниях и людях, с которыми должны будете попытаться наладить контакт. Сюда входят венчурные капиталисты, бизнес-ангелы и все, что между ними.

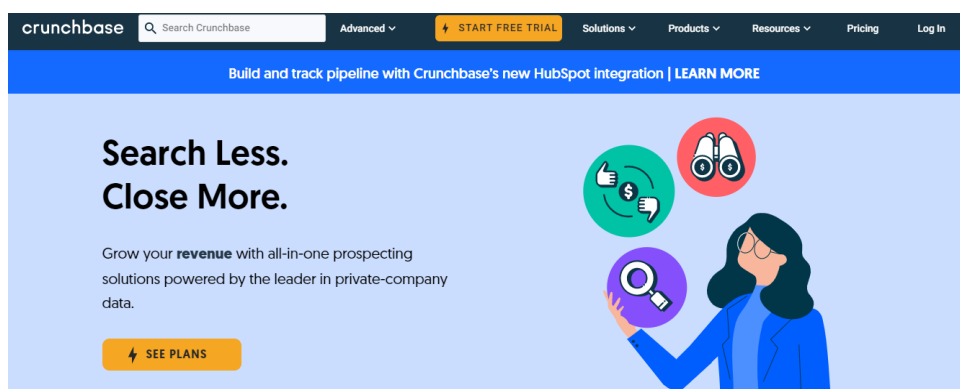


Рисунок 3. Официальный сайт Crunchbase

На русскоязычном рынке аналогом вышеперечисленных инвестиционных площадок себя позиционировал *RusBase*. Но по содержанию этот ресурс все же отличается от Crunchbase. RusBase предоставляет пользователям возможность вести поиск инвестиций, искать поставщиков бизнес-услуг, и иметь доступ в базу данных о всех публичных венчурных сделках индустрии [4]

Не менее полезными могут быть и следующие ресурсы:

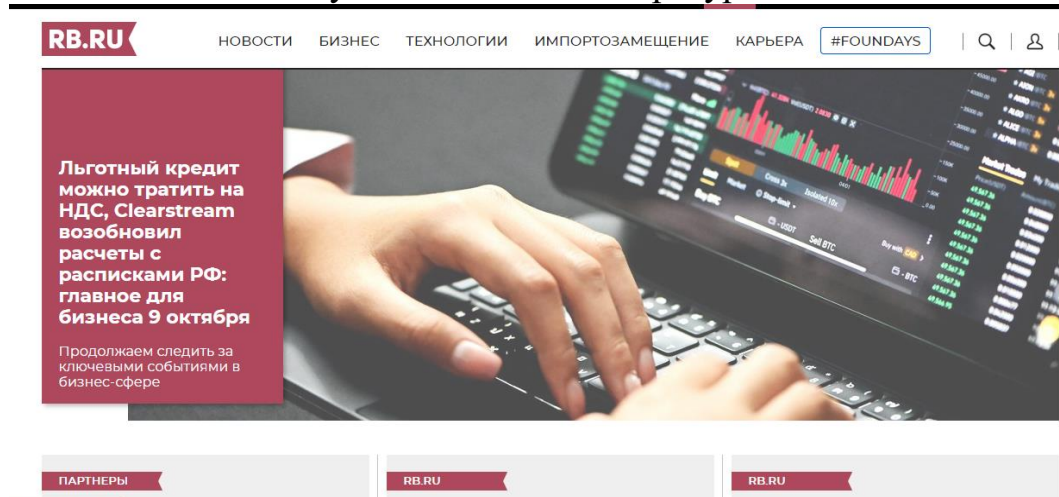


Рисунок 4. Официальный сайт *RusBase*

*Product Hunt* — американский веб-сайт, на котором можно делиться и открывать для себя новые продукты. Он был основан Райаном Гувером в ноябре 2013 года.

Пользователи отправляют продукты, которые перечислены в линейном формате по дням. Сайт включает в себя систему комментариев и систему голосования, аналогичную Hacker News или Reddit. Продукты, набравшие наибольшее количество голосов, поднимаются на вершину списка каждый день.

Продукты разделены на четыре категории; технологические продукты (веб-приложения, мобильные приложения, аппаратные продукты и т. д.), игры (ПК, веб-приложения, мобильные приложения), книги и подкасты. Для представления просто требуется название продукта, URL-адрес и слоган. По данным Гувера, по состоянию на 2016 год на веб-сайте было обнаружено более 100 миллионов продуктов от 50 000 компаний.

Сайт также включает в себя ежедневный список адресов электронной почты, который рассылает вчерашние лучшие технические «охоты» (продукты), а также избранную коллекцию. Также доступна версия дайджеста для игр и книг.

Product Hunt также доступен в виде приложения для iOS, приложения для macOS, приложения для Android и расширения для Google Chrome. Штаб-квартира компании находится в Сан-Франциско.

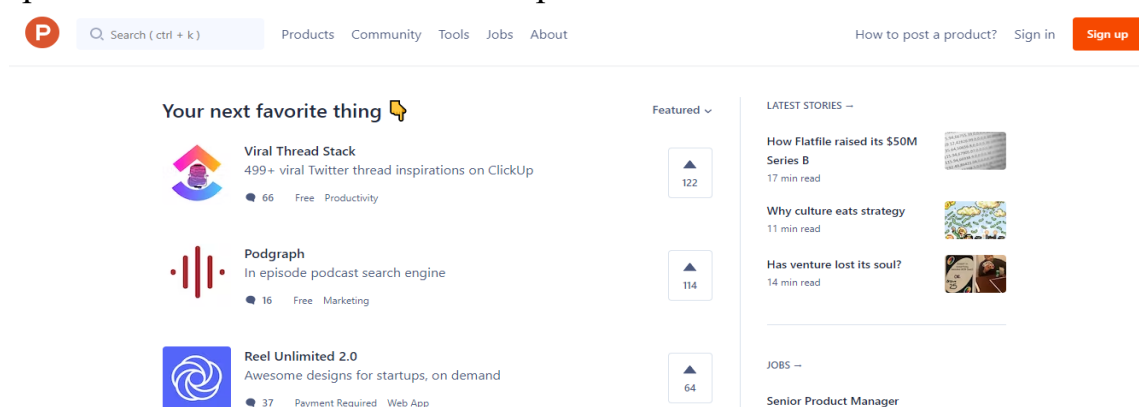


Рисунок 5. Product Hunt



Сайт получил финансирование от *Y-Combinator*. В ноябре 2016 года *AngelList* приобрела *Product Hunt* за 20 миллионов долларов.

*PIPELINE* - сервис, разработанный при поддержке РВК в рамках программы *Talk2VC* [5], помогает предпринимателям сделать первые шаги в поисках инвестора и вести статистику о качественных и количественных изменениях проекта. Также, участники сервиса могут в приоритетном порядке подавать заявки на участие в инвестиционных сессиях "Speed Dating" и в Личном кабинете записываться на встречи с инвесторами в других форматах.

*BetaList* — это сообщество создателей стартапов, которые демонстрируют свои проекты и устанавливают обратную связь с их первыми пользователями [6].

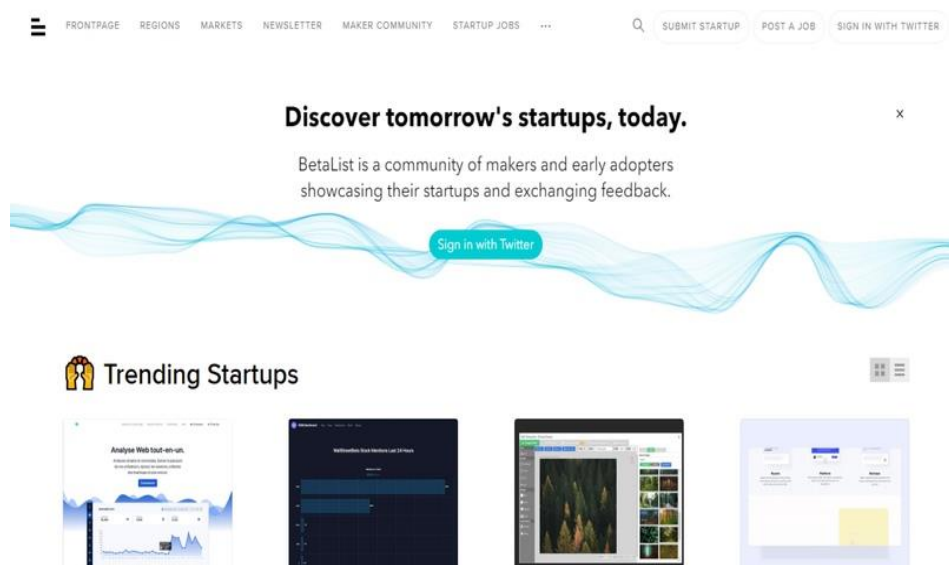


Рисунок 6. Сайт *BetaList*

Данный сервис позволяет пользователю публиковать информацию о своем стартапе, предварительно ознакомив с критериями отбора. Следует отметить, что *Betalist* не занимается продвижением блогов, информационных бюллетеней, книг, онлайн-курсов и интернет-магазинов. Если стартап соответствует требованиям сервиса, владелец получает электронное письмо, подтверждающее участие проекта-претендента. В течение 7-8 недель сервис публикует проект своем сайте, и попадаете в список опубликованных проектов на сайте. Этот процесс можно ускорить за дополнительную плату.

Веб-сайт получает более 250 000 просмотров страниц в месяц. Количество переходов по ссылкам в первую очередь зависит от вашего продукта и его популярности, но в среднем может наблюдаться около 500 кликов в неделю.

*UsabilityHub* — это платформа для онлайн-исследований, которая позволяет проверить различные бизнес-гипотезы на реальных пользователях.

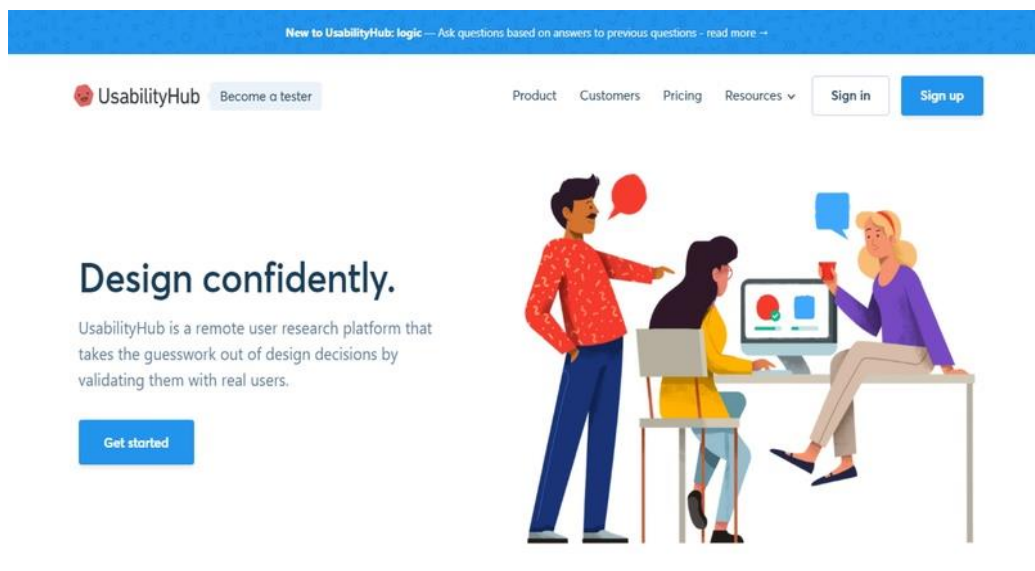


Рисунок 7. Сайт UsabilityHub

Если автор проекта уже выяснил, что его продукт нужен рынку и успел запустить сайт, появляется необходимость измерения количественных показателей его использования. Данный сервис предлагает для этого несколько видов тестов. Например, при проведении «пятисекундного тестирования» можно измерить, насколько хорошо дизайн сайта передает основной смысл вашего продукта потенциальному потребителю. Участникам демонстрируется изображение в течение пяти секунд, после чего они отвечают на вопросы, основываясь на своей памяти и впечатлении от дизайна. Также есть тесты предпочтений, которые могут помочь вам выбрать между вариантами дизайна. Проводится опрос среди пользователей, и они отвечают, какой из вариантов им больше нравится. Тестирование по первому щелчку — это метод, который позволяет убедиться, что первый щелчок, который пользователь делает на сайте для выполнения заданной задачи, понятен и прост.

*Proved.co* - был создан экспертами по исследованию рынка с миссией помочь предприятиям любого размера без проблем, быстро и за небольшую плату проверять новые идеи для стартапов. Можно протестировать все, что клиенты могут купить или использовать.



Рисунок 8. Сайт Proved.co

Чтобы пользоваться сервисом, необходимо заполнить анкету, описать проблему, над которой вы работаете, разрабатывая собственный проект и само решение проблемы. Далее пользователи отвечают на пять ключевых вопросов: насколько остро стоит проблема, подходит ли ваше решение, уникально ли оно и стоит ли того, чтобы рассказать о нём друзьям или заплатить за него. Также, здесь можно настроить нацеленную рекламу, выбрать группу пользователей, от которых вы хотите получить отзыв, установить необходимый возраст, пол, род занятий и страну, в которой находится респондент. По словам разработчиков, первый отзыв, касательно идеи вашего продукта, приходит уже через несколько минут от реальных людей.

Сервис позволяет пользователям отслеживать ответы и получать информацию в режиме реального времени. Отзывы всех тестировщиков (в зависимости от тарифного плана) поступают в течение суток. На панели инструментов можно посмотреть, в каких критериях идея получила высокую оценку, и в каких – низкую. Скачать и проанализировать данные с обратной связью пользователей можно самостоятельно.

*QuickMVP* - бесплатный и удобный сервис для тестирования ниши.

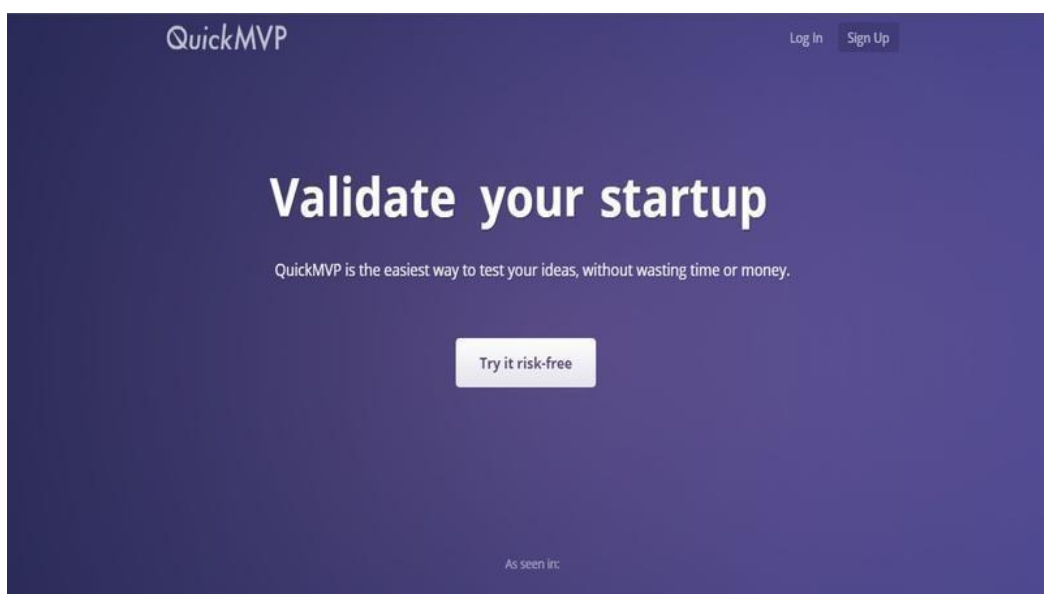


Рисунок 9. Сайт QuickMVP

Инструменты, которые предлагает платформа:

- Опросы и интервью с клиентами. В сервисе существует возможность количественно и качественно оценивать потребности потенциальных потребителей. Платформа поможет определить основные проблемы клиентов, а следовательно и бизнес-возможности, на которых нужно будет сделать акцент.

- Лендинг. После того, как проблема подтверждена, можно протестировать несколько своих решений с помощью целевых страниц, создать веб-сайт для своего продукта по шаблону, без навыков в программировании и дизайне. Также, можно сделать свою страницу доступной для большего числа потенциальных клиентов с помощью встроенного средства создания рекламы - Google Ad.

- **Экономические показатели.** QuickMVP рассчитывает бизнес-прогнозы для каждой идеи на основе результатов экспериментов, представленных выше. Прежде чем приступить к реализации идеи, вы можете получить данные о стоимости привлечения клиентов, марже, размере рынка и потенциальной прибыли для своих идей.

*Gust* — это глобальная платформа SaaS для создания, эксплуатации и инвестирования в масштабируемые быстрорастущие компании [7]. Онлайн-инструменты *Gust* помогают предпринимателям создавать, управлять и собирать средства для компаний, а также обеспечивают управление потоком сделок и взаимоотношениями с инвесторами.

Являясь крупнейшим в мире сообществом предпринимателей и начинающих инвесторов из 192 стран, *Gust* стал пионером в области сотрудничества в области акционерного финансирования и является официальной платформой ведущих мировых федераций бизнес-ангелов и венчурных акселераторов.

*Gust* поддерживает *Gust Launch*, первую платформу «Компания как услуга» (SaaS) для основателей стартапов, которая облегчает все юридические, финансовые и бэк-офисные операции, необходимые для регистрации в качестве Delaware C Corp и управления новым быстрорастущим предприятием.

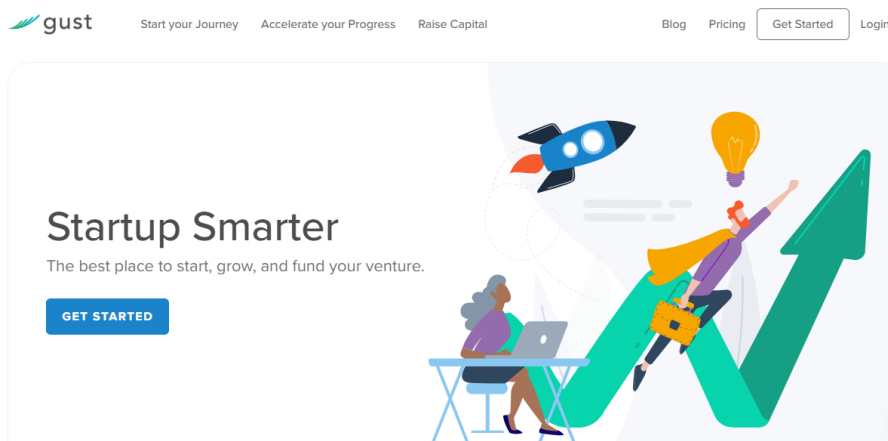


Рисунок 10. Сайт Gust

Платформа *Fbs* (“for success”) - один из крупнейших в мире порталов для стартапов, на которой представлены более тысячи акселерационных программ и фондов со всего мира. Согласно открытым данным, на этой площадке 11 428 стартапов привлекли более 3,3 миллиардов долларов.

После регистрации на портале можно получить доступ к акселераторам из любой точки мира, включая корпоративные акселераторы и фонды; узнать о мероприятиях, сроках сборов заявок; посмотреть на стандартные анкеты крупнейших акселераторов (YC, TechStars) и даже узнать о вакансиях.

Платформа объединяет основателей стартапов (личные профили), сами стартапы (профиль стартапа), менторов, инвесторов, акселераторы, организаторов мероприятий, представителей другой инфраструктуры для стартапов.

## Welcome Home

F6S is where Founders grow together

**Connect** with Accelerators, Grants and Contracts

**Get help** with free stuff, talent search & exposure

**Grow** your Startup in the #1 global community

Join 4,611,356 Founders & Startups  
Always free - connect now

**in** CONTINUE WITH LINKEDIN

**f** CONTINUE WITH FACEBOOK

or use e-mail

By signing in or joining F6S, you agree to the [Terms of Service](#) and [Privacy Policy](#). We use cookies on our site to enable us to provide the best user experience for you and to protect against bots, fraud and

Рисунок 11. Сайт F6s

*StartupBlink* — это самая обширная в мире *карта экосистем стартапов* и исследовательский центр, предоставляющий разработчикам экосистем государственного сектора, инновационным организациям и исследователям знания и инструменты для картирования, брендинга и развития их экосистем стартапов.

Глобальная карта стартап-экосистем насчитывает десятки тысяч зарегистрированных стартапов, коворкингов и акселераторов. Годовые отчеты и рейтинговые таблицы *StartupBlink* используются сотнями тысяч читателей для принятия решений о переезде, инвестициях и исследованиях.

Раз в год *StartupBlink* публикует отчет *Global Startup Ecosystem Index*, в котором оцениваются стартап-экосистемы 1000 городов и 100 стран.

*StartupBlink* также проводит ежеквартальный саммит *Startup Ecosystem Summit*, собрание ведущих мировых экспертов и лидеров в области развития экосистем, предлагая форум для обучения и обмена знаниями о том, что нужно для создания среды, комфортной для развития предпринимательства [8].

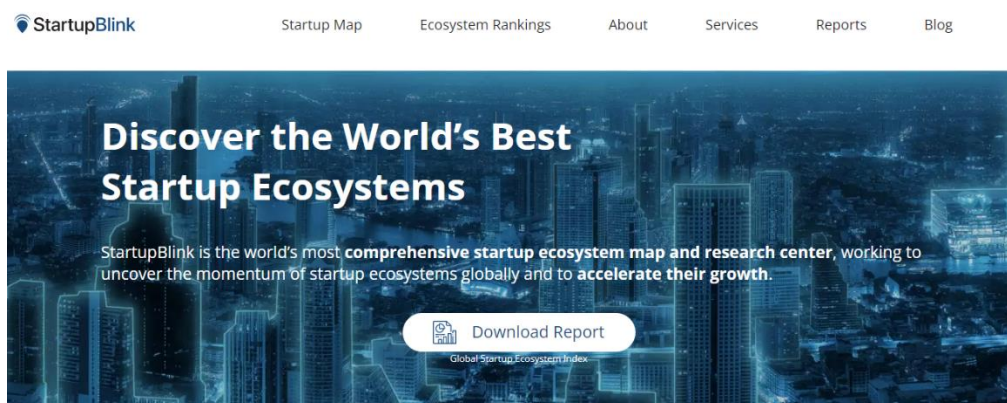


Рисунок 12. Сайт StartupBlink

Сервис *XMind* предоставляет пользователям возможность структурировать информацию в виде диаграмм и таблиц. Для работы необходимо установить софт на ПК. Есть шаблоны карточек, маркеры, ярлыки, фигуры. Доступна опция вставки ссылок, файлов, таблиц, подпунктов. Реализована интеграция с Evernote.





Рисунок 13. Логотип Xmind

С помощью предоставленных инструментов, стартапер сможет планировать, управлять задачами и структурировать большой поток данных.

Есть специальные площадки, где собраны предложения для инвесторов. Каждый сможет разместить свой проект и получить отклик. Объявления оформляются в виде мини-сайта с возможностью установить конкретную цель, например, поиск инвестиций или получение кредита. Например, [business-platform.ru](http://business-platform.ru), [startup.network](http://startup.network), [beboss.ru](http://beboss.ru), [innmind.com](http://innmind.com)

Предложение напрямую

Многие крупные компании готовы выступить инвестором в отраслевых стартапах. Это отличная возможность найти финансы для развития проекта, но в таком случае все идеи тщательно отбираются и проверяются. К тому же, можно найти список инвесторов в вашей области и попробовать связаться с ним напрямую. Учтите, что в таком случае у вас должна быть готова презентация и команда. Платформы для предложения напрямую: I2BF GLOBAL VENTURES, Flint Capital, AddVenture, АдмиТад Инвест.

Краудфандинговые сайты

На таких сайтах вы можете получить финансы от любого заинтересованного человека, если он поверит в ваш проект. Если человек решается вложить в ваш проект, ему необходимо будет предложить подарок или призы, небольшой процент или долю будущего предприятия. С помощью таких платформ можно не только получить финансирование, а и просто привлечь внимание общественности, используя его как маркетинговый канал.

Примерами могут выступить такие платформы, как Flint Capital, [kickstarter.com](http://kickstarter.com), [indiegogo.com](http://indiegogo.com), [rockethub.com](http://rockethub.com).

Flint Capital — международный венчурный фонд, специализирующийся на финансировании высокотехнологичных проектов в области интернет-технологий, медиа, а также финансовых сервисов, проектов в сфере виртуальной и дополненной реальности. Основан 13 мая 2013 года партнёрами управляющей компании Flint Management: Дмитрием Смирновым и Олегом Сейдаком, в прошлом CEO и Managing Director Finam Global соответственно. В настоящий момент управляющим партнером фонда является Дмитрий Смирнов.

Представители фонда характеризуют свой основной фокус на перспективные, масштабируемые проекты, ориентированные на глобальные развитые и быстрорастущие рынки.

*Runa Capital* — международный венчурный фонд с головным офисом в Пало-Алто, Калифорния, инвестирующий в высокотехнологичные компании на ранних стадиях развития. Фонд поддерживает стартапы в сферах искусственного интеллекта, машинного обучения, инфраструктурного ПО и открытого ПО и компании, разрабатывающие корпоративное ПО, технологии для финансовой сферы, образования и здравоохранения.

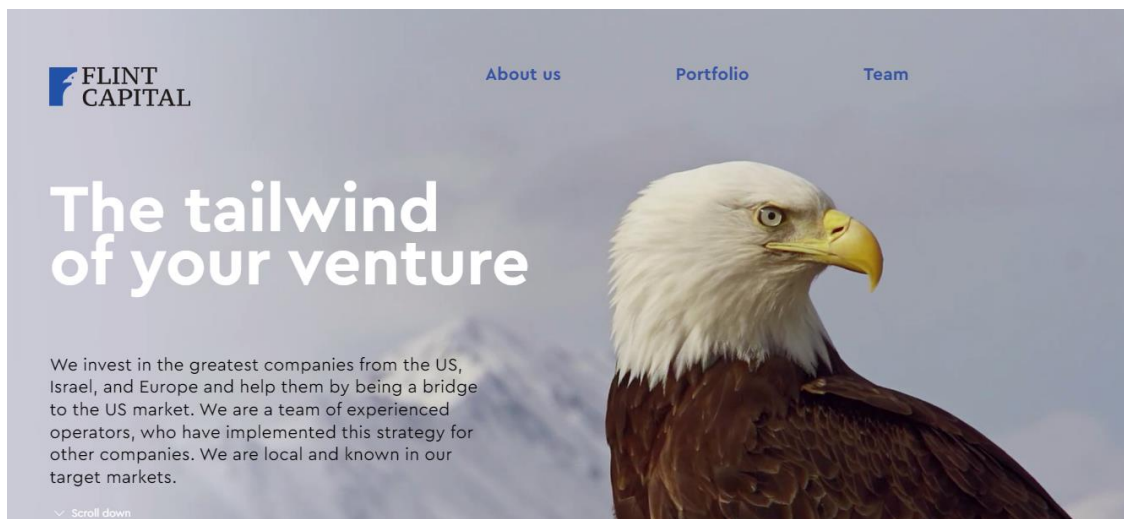


Рисунок 14. Сайт Flint Capital

В 2010—2019 годах Runa Capital привлекла средства в 3 фонда и проинвестировала более 60 компаний из Европы и Северной Америки, в том числе Nginx, MariaDB, Brainly, drchrono, Smava and Mambu.

Indiegogo — сайт финансирования творческих проектов по схеме общественного финансирования, основанный в 2008 году []. Штаб-квартира находится в Сан-Франциско, Калифорния. Девять миллионов людей со всего мира посещают сайт ежемесячно. Наряду с Kickstarter, Indiegogo является одним из самых популярных веб-сайтов общественного финансирования в Америке.

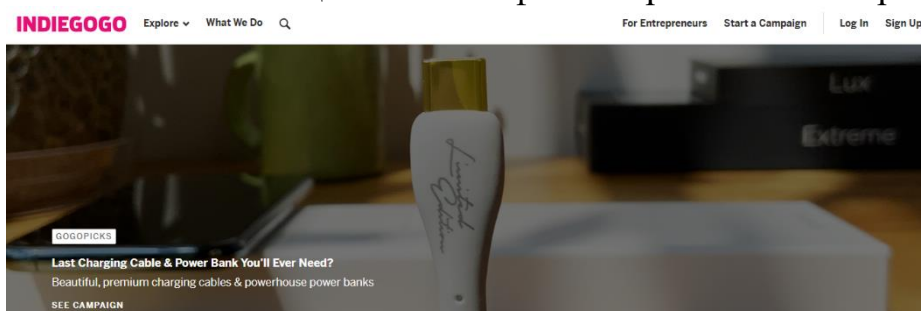


Рисунок 15. Сайт Indiegogo



## 2. IT -продукт и его Продвижение

### 2.1. Сервисы для коммуникации с сотрудниками

#### Trello

Trello — это канбан-доска для организации дел компании и хранения документов. Карточки можно перемещать по мере выполнения задания, менять статусы, прикреплять комментарии, выставлять сроки и назначать ответственных лиц.

Вы можете использовать доску бесплатно. Если нужны дополнительные возможности: создание графиков и таблиц, добавление изображений, календаря и карт, — оформите подписку за 10 долларов в месяц. При этом премиум-аккаунт для ведения бизнеса обойдется в 3,75 долларов в месяц, если купите годовой абонемент. Он открывает дополнительные интеграции, позволяет загружать файлы до 250 мегабайтов и настраивать доску под любые задачи.

Особенности:

- Сервис для учета задач.
- У программы есть как десктопная, так и мобильная версия.

Преимущества:

- Удобный интерфейс.
- Интеграция с сервисами — можно загружать таблицы с аналитическими данными и исследованиями, подключать напоминания о новых email, обрабатывать заявки и многое другое. Выбор приложений позволяет настроить доску под ваш проект.

- Есть возможность подключить к доске команду.
- Можно использовать горячие клавиши и быстро переключаться между задачами и досками.

Минусы:

- Для полноценной работы нужно приобрести лицензию.

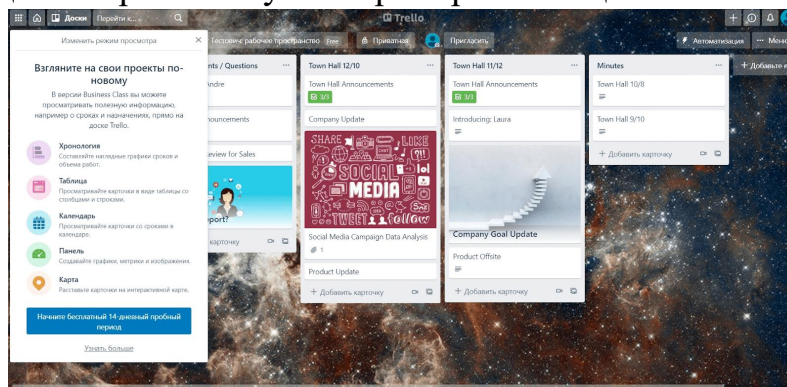


Рисунок 16. Интерфейс Trello

**Slack.** Позиционирует себя как «убийца Skype и внутрикорпоративной почты». Здесь можно не только создавать проекты, но и строить коммуникацию

между участниками. Возможна интеграция с различными сервисами, например с Google Drive, Dropbox, Asana, Twitter и пр. Стоимость: от 6,25 евро в месяц. Есть бесплатная версия с доступом к 10 000 последних сообщений, возможностью интеграции с 10 приложениями и создания неограниченного количества публичных и закрытых каналов.

**Особенности:**

- Мессенджер для общения внутри компании и обсуждения задач.

**Плюсы:**

- Легкий поиск сообщений.
- Интеграция с приложениями — Google Диск, Zoom, Zendesk, Asana, Box File Picker и так далее.
- Можно «призвать» пользователя, как в обычном приложении — и сотрудник получит уведомление о новом сообщении.
- Есть возможность отправки email выбранным коллегам.

**Минусы:**

- Нужно платное подключение для полноценной командной работы.

*Telegram*

Кроме обычного общения, отправки файлов и документов любых форматов, синхронизированного доступа с любых устройств, сервис гарантирует защиту от хакерских атак. Удобное приложение, которое обеспечивает быструю доставку сообщений и создание чатов до 5 000 человек.

**Особенности:**

- Приложение можно установить на компьютер, смартфон или включить в браузере.

**Плюсы:**

- Групповые чаты и звонки.
- Создание чатов, закрытых каналов и пабликов.
- Можно создавать свои стикеры, большие статьи, писать посты, обмениваться документами, изображениями и музыкой.

Есть редактор картинок.

**Минусы:**

- Нужны дополнительные инструменты для полноценной работы — таблицы, документы и тому подобное.

*Microsoft To Do*

Планировщик дел на Android и iOS. Вы сами загружаете туда задачи и ставите им сроки выполнения, а сервис присылает напоминания о том, что нужно что-то сделать.

Приложение не только уведомляет о задачах: оно само добавляет нужные дела в список на день. А еще в нем можно создавать повторяющиеся задачи, классифицировать их по цветам, добавлять заметки и файлы до 25 Мб. Задачами можно делиться с кем-то — например, с сотрудником, с которым вы должны поехать на встречу, или с семьей.

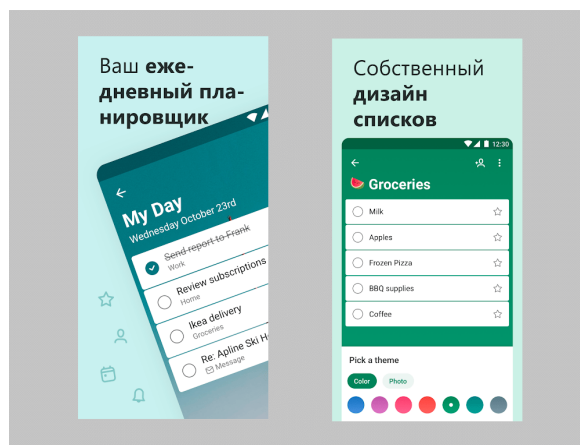
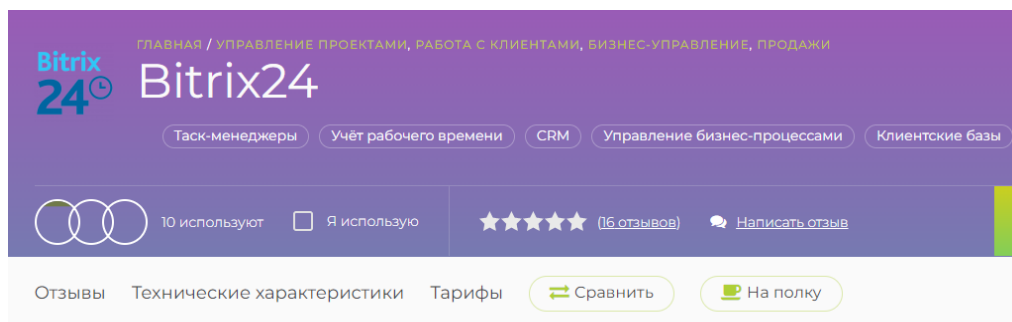


Рисунок 17. Microsoft To Do

### Bitrix24

Платформа предназначена для автоматизации бизнес-процессов. В Личном кабинете можно вести сделки от начала и до конца, следить за всеми этапами работы. К каждой задаче можно подключать коллег, выставлять дедлайны, смотреть динамику изменений. Все почтовые ящики подключаются к личному домену.



### Обзор Bitrix24

Рисунок 18. Страница Bitrix24

У сервиса есть встроенный антиспам и антивирус, который проверяет почтовые ящики. Общаться с партнерами можно через чат или с помощью видео. Реализована виртуальная АТС, где все звонки распределяются автоматически.

### Jira

Jira — коммерческая система отслеживания ошибок, предназначена для организации взаимодействия с пользователями, хотя в некоторых случаях используется и для управления проектами. Разработана компанией Atlassian, является одним из двух её основных продуктов (наряду с вики-системой Confluence). Имеет веб-интерфейс.

Название системы получено путём усечения слова «Gozilla» — японского имени монстра Годзилла, что, в свою очередь, является отсылкой к названию конкурирующего продукта — Bugzilla; создавалась в качестве замены Bugzilla и во многом повторяет её архитектуру. Система позволяет работать с несколькими проектами. Для каждого из проектов создаёт и ведёт схемы безопасности и схемы оповещения.

Первый выпуск состоялся в 2002 году. Изначально применялась в процессах разработки программного обеспечения, впоследствии нашла применение в качестве инструмента управления проблемами, задачами, проектами в различных отраслях.

Позволяет контролировать выполнение текущих задач, следить за дедлайнами и работать в команде. Все изменения сохраняются в истории проектов.

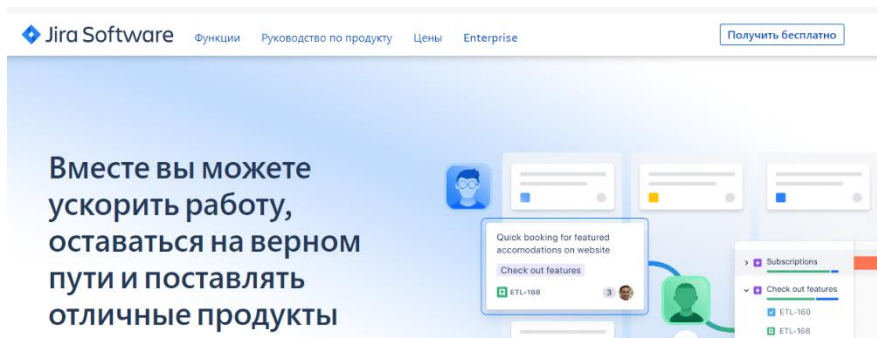


Рисунок 19. Сайт Jira

## 2.2. Сервисы для коммуникации с клиентами

### Google Ads

Сервис для создания контекстной рекламы в поисковике Google или на сайтах партнерах, позволяющий привлекать заинтересованных пользователей. Оплата происходит за клики, но есть возможность указать стоимость целевого действия. Один из наиболее конверсионных источников трафика.

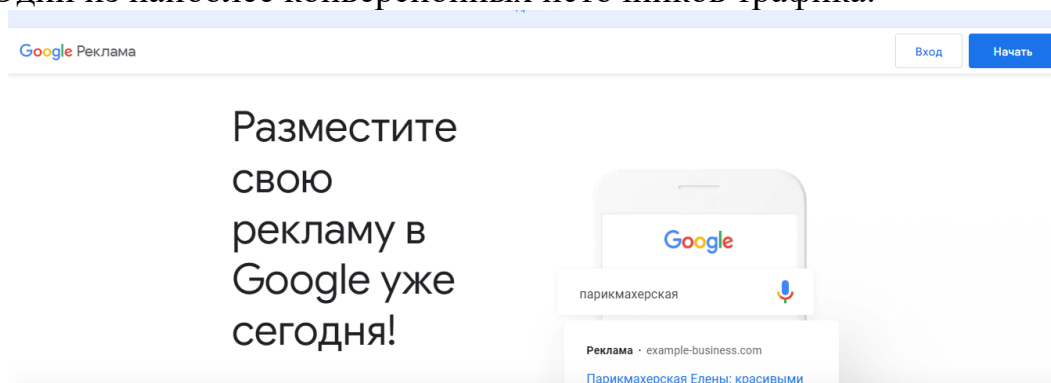


Рисунок 20. Страница Google Ads

Ads (ранее известный как AdWords) — сервис контекстной рекламы от компании Google, предоставляющий удобный интерфейс и множество инструментов для создания эффективных рекламных сообщений. Ads — флагманский рекламный проект Google и основной источник доходов компании. 24 июля 2018 года бренд AdWords изменил название на Google Ads, а также получил новый логотип. В России название сервиса AdWords изменилось на «Google Реклама». Новый бренд символизирует весь доступный спектр рекламных кампаний, включая поисковые, медийные и видеокампании.

Создание стандартного рекламного объявления в Google Ads занимает, как правило, менее 30 минут. Процесс размещения рекламы происходит в 8 шагов.

1. В первую очередь рекламодателю необходимо создать аккаунт в Google Ads, для чего достаточно указать адрес электронной почты и пароль, используемые для доступа к другим сервисам Google. При отсутствии таковых можно зарегистрировать аккаунт Ads на любой другой адрес электронной почты.

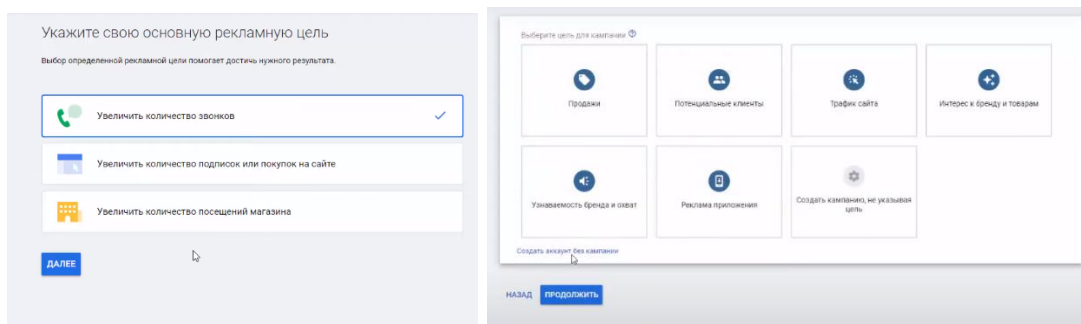


Рисунок 21. Регистрация на Google Ads

2. Затем рекламодателю предлагается выбрать целевую аудиторию (т.е. произвести таргетинг) путём выбора местоположения (страна, регион, город), языка потенциальных клиентов, а также демографических показателей (возраст и пол).

3. На третьем этапе необходимо указать бюджет рекламной кампании.

4. На четвёртом устанавливается цена за клик по рекламному сообщению. При выборе рекомендованного варианта «Автоматическое назначение ставок» указывается верхний предел цены за клик, к примеру, 10 руб., а система автоматически подбирает минимально необходимую цену за клик для показа рекламного объявления на лучшем в каждом конкретном случае позиции. При этом фактическая цена за переходы в большинстве случаев меньше максимальной цены и никогда её не превышает из группы, выявляя при этом наиболее эффективные.

5. Пятый этап посвящён созданию непосредственно рекламного объявления, для чего необходимо ввести 2 заголовка (по 30 символов каждый), кратко описать рекламируемый продукт в строке описания (90 символов) и указать адрес веб-страницы, на которую перейдет пользователь, нажавший на контекстное объявление. Внести изменения в текст объявления возможно в любой момент.

6. На шестом этапе рекламодателю предлагается выбрать ключевые слова, при использовании которых в поисковом запросе пользователя будет показано размещаемое объявление. Предлагаемый на этом этапе инструмент подсказки ключевых слов позволяет подобрать ключевые слова, проверить их популярность (количество запросов пользователей по ним) и узнать уровень конкуренции по данным словам среди рекламодателей.

7. Затем можно создать дополнительные варианты объявлений, из которых впоследствии Ads эмпирическим путём отберёт наиболее эффективные.

8. На заключительном этапе рекламодателю необходимо выбрать способ оплаты и ввести платежные данные.

*Momentive* Inc. (ранее SurveyMonkey Inc. ) — компания по управлению опытом , которая предлагает облачное программное обеспечение для анализа брендов, анализа рынка, взаимодействия с продуктом, взаимодействия с сотрудниками, взаимодействия с клиентами, разработки онлайн-опросов, а также набор платных серверных программ [10].

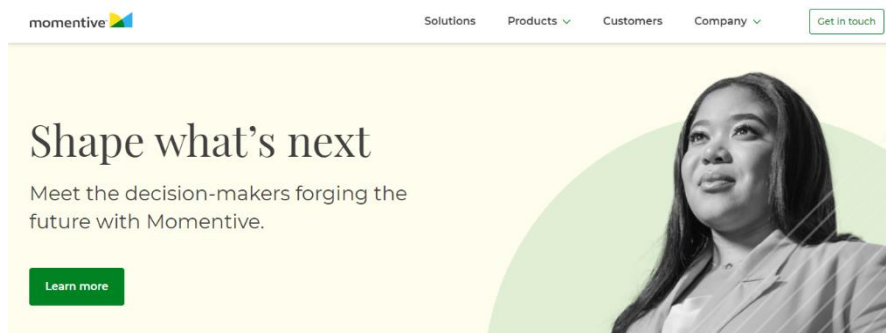


Рисунок 22. Сайт Momentive

На сегодняшний день, популярными аналогами Momentive являются такие сервисы, как Ezoic, RichAds, Media.net, InfoLinks, Amazon Affiliates.

#### **4. Методические рекомендации по возможностям использования ИКТ ресурсов при изучении курса "Робототехника"**

##### **Модуль 1. Конструирование и программирование робота**

Изучение робототехники позволяет применить на практике теоретические знания, полученные школьниками при изучении математики, химии, физики, информатики, и способствует более глубокому пониманию основ научного знания. Закрепление на практике полученных в школьном курсе знаний и навыков обуславливает повышение качества образовательных результатов, формируемых в системе общего образования.

При внешней привлекательности, управление роботом складывается из решения целого комплекса довольно сложных инженерных задач. При проектировании и конструировании роботов ребятам приходится решать задачи, выходящие за содержание школьного курса и относящиеся к инженерным дисциплинам: механика, теория автоматического управления, сопротивление материалов и др.

Программирование как один из основных разделов робототехники является не только универсальным инструментом компьютерного моделирования объектов и процессов, но и мощным инструментом в развитии мышления школьников. Необходимость описать средствами языка программирования и стандартных алгоритмических конструкций нестандартные процессы, протекающие в реальном мире, формирует умение определять цели своей деятельности, самостоятельно выстраивать учебную задачу, определять пути и методы ее решения и таким образом достигать поставленной цели. Конструирование и проектирование робота позволяет обучающимся самостоятельно проверить уровень владения теоретическим знанием и учебными навыками.

Робототехника способствует развитию пространственного воображения, мелкой моторики, внимательности, аккуратности, а также навыков проектирования и эффективного использования кибернетических систем.

Кроме того, занятие робототехникой позволяет школьникам принимать участие в различных соревнованиях, групповых играх, конкурсах и состязаниях роботов, что позволяет закрепить изученный материал, формирует адекватную самооценку своих возможностей и становится новым мотивационным стимулом к обучению (Рисунок 1.1).

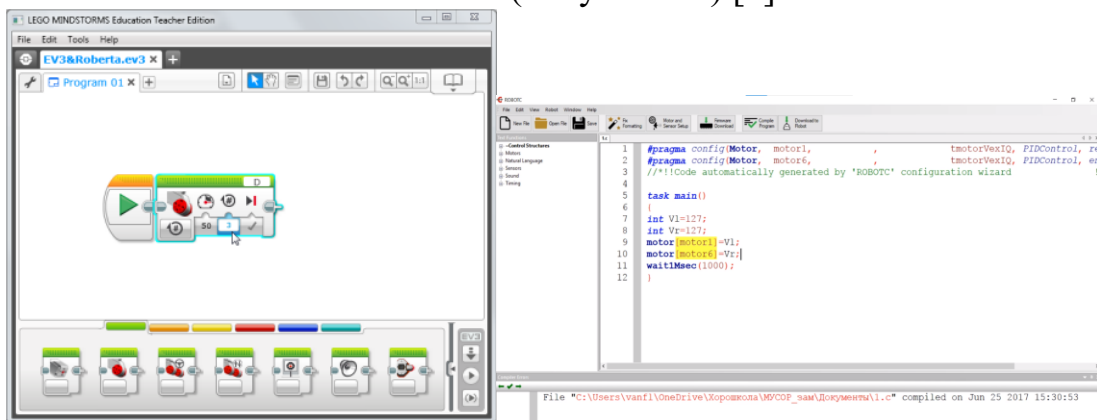




Рисунок 1.1. Международный этап соревнования WRO (World Robot Olympiad)

### Отличительные особенности

При изучении курса робототехники для организации активной деятельности обучающихся необходимо создание интерактивной дидактической среды на основе современных робототехнических конструкторов и площадок, полей, троп для тестирования роботов. На сегодняшний день наиболее часто используемыми средами программирования и конструирования роботов являются среды Robolab, LEGO MINDSTORMS EV3 и RobotC (Рисунок 1.2) [1].



а)

б)

Рисунок 1.2. Интерфейс среды программирования Lego Mindstorms EV3 (а) и RobotC (б)

При выборе робототехнического конструктора стоит обратить внимание на комплект навесного оборудования, а также на время, затрачиваемое на сборку и приведение в исходное состояние конструктора. Чем шире спектр навесного оборудования робота, различных датчиков, механических устройств, электрических двигателей, сервоприводов и иных электронных компонентов, тем разнообразнее спектр предлагаемых для учащихся задач, при решении которых учащиеся смогут изучить принципы конструирования и программирования роботов,

формировать метапредметные результаты и межпредметные связи. Также, разнообразие электронных компонентов вызывает у учащихся интерес к инженерному творению, что позволит реализовать серьезные и интересные в исследовательском плане научно-практические работы и проекты.

Основными задачами в обучении являются:

- знакомство обучающихся с комплексом базовых технологий, применяемых при создании роботов;
- раскрытие межпредметных связей с такими предметами как физика, информатика, математика;
- решение учащимися базовых кибернетических задач, результатом каждой из которых является работающий механизм или робот с автономным управлением;
- повышение мотивации учащихся к изобретательству и созданию собственных роботизированных систем;
- формирование у учащихся стремления к достижению цели и созданию собственного качественного учебного продукта;
- формирование навыков проектного мышления и коммуникативных навыков работы в команде, коллективной работы.

Наиболее распространенной моделью обучения робототехнике в рамках школы является *концентрическая модель обучения*. Стандартный цикл обучения включает следующие разделы: конструирование, программирование, проектирование.

Рекомендуется выстроить график реализации программ по робототехнике с учетом массовых мероприятий в научно-технической сфере для учащихся (турниры, соревнования, конференции). Это позволит, с одной стороны, эффективно реализовывать программу обучения, а с другой, принимать участие в соревнованиях, не превышая нормы времени, затрачиваемые на обучение.

В целях создания условий для индивидуального развития личности одаренных учащихся необходимо предусмотреть и разработать образовательные траектории для реализации индивидуальных образовательных маршрутов, как при обучении, так и в особенности при подготовке к олимпиадам, конкурсам и соревнованиям; систему индивидуальных занятий и консультаций.

Основное содержание предмета включает в себя следующие темы:

1. *Основы конструирования*: виды и особенности механических передач, использование механических передач, оптимизация конструкции согласно поставленной задаче;
2. *Основы программирования*: использование циклов, логических и математических операторов, блоков действий;

3. *Соревновательная (спортивная) робототехника*: использование полученных знаний для решения прикладных задач, творческая сборка собственных роботизированных систем.

**Программирование в среде Robolab.** Программное обеспечение ROBO LAB позволяет программировать NXT и RCX роботов. Недостатком данного языка программирования является отсутствие поддержки стандартного датчика цвета и соединения по Bluetooth. Несмотря на это, среда программирования ROBO LAB является довольно распространенной по всему миру и ее разрешается использовать на Всемирной робототехнической олимпиаде WRO.

При запуске Robolab предлагает три уровня работы: «Администратор», «Программист» и «Исследователь» (Рисунок 1.3):

- Режим администратора позволяет настраивать контроллер на работу со средой.
- Режим программиста позволяет создавать программы и загружать их в микроконтроллер.
- Режим исследователя позволяет осуществлять запись данных, поступающих с датчиков микроконтроллера, с их последующим анализом.

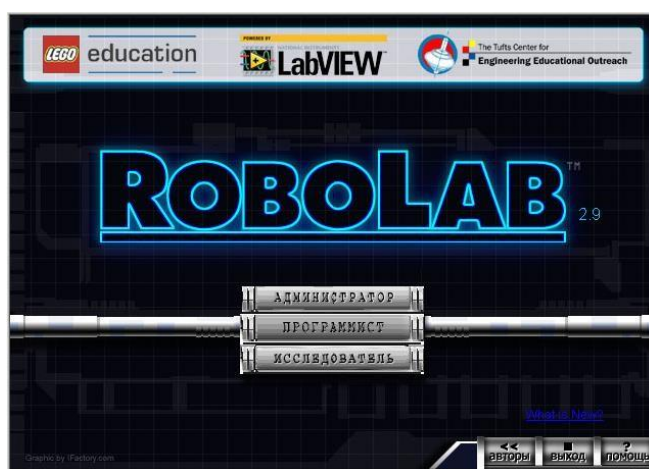


Рисунок 1.3. Окно запуска Robolab 2.9

Среда программирования представляет собой два окна, относящихся к одному проекту: Front Panel и Block Diagram. Первое (передняя панель) используется в режиме исследователя. Второе, в котором уже расположены две пиктограммы светофоров (рабочее поле программы), предназначено для составления программы (Рисунок 1.4).

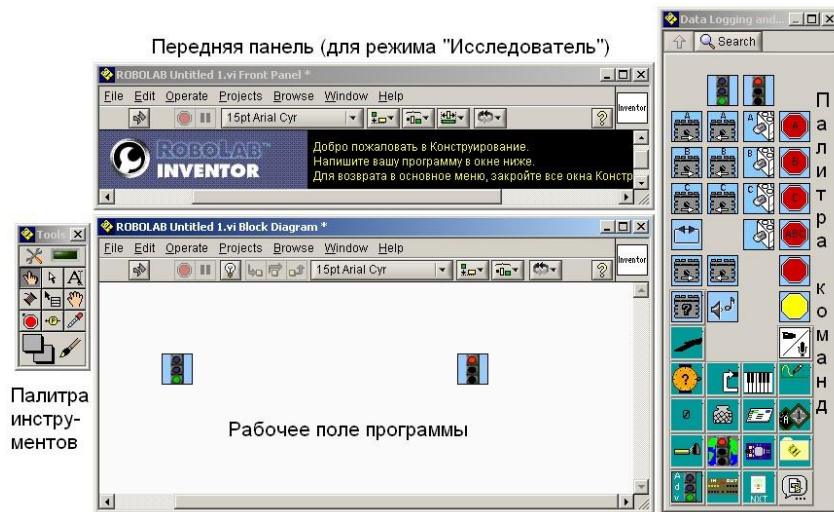


Рисунок 1.4. Окна среды программирования Robolab.

Два вспомогательных окна — Tools Palette и Functions Palette — содержат все необходимое для составления программы. В случае закрытия их снова можно вывести на экран через пункт Windows верхнего меню.

Одним из новшеств Robolab 2.9 является палитра примеров Behaviors (Рисунок 1.5). Это блоки, содержащие в себе готовые части программ под определенные задачи. Например, блок Go Straight (двигаться прямо) запускает моторы А и С вперед и через 1 секунду выключает их (Рисунок 1.6). Однако для запуска этого фрагмента программы его надо обязательно поместить между светофорами, соединив их розовыми проводами.

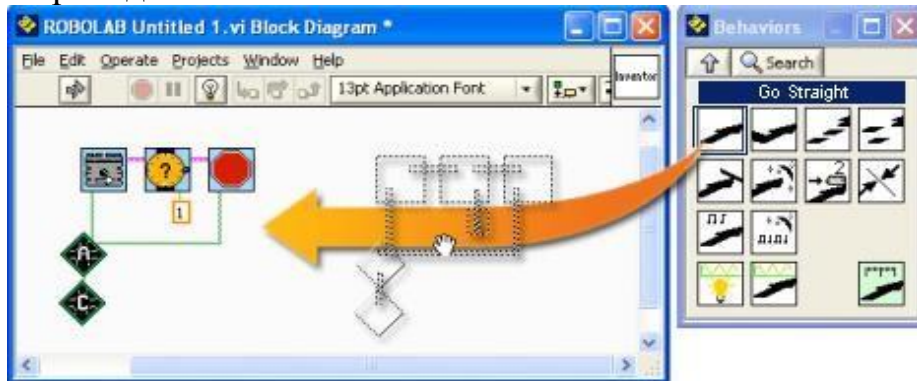


Рисунок 1.5. Внедрение примера в поле программы.

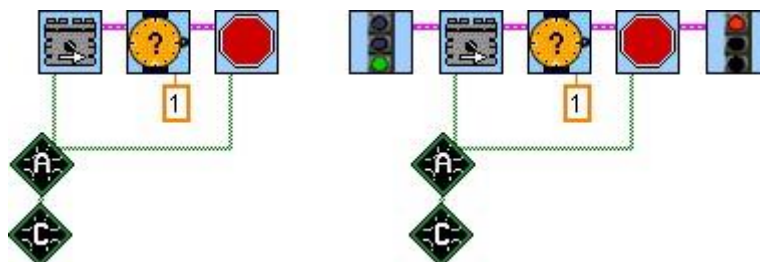


Рисунок 1.6. Содержание примера и размещение его между светофорами.

**Программирование в среде RobotC.** RobotC - кросс-робототехнический язык программирования, созданный на основе языка C (Рисунок 1.7). В одном пакете приложения существуют три варианта программирования в RobotC: графический естественный язык, текстовый естественный язык и текстовый. Версии языка предназначены для облегчения естественного перехода от графического (Scratch-подобного, упрощенного) к естественному языку и к полному тестовому RobotC.

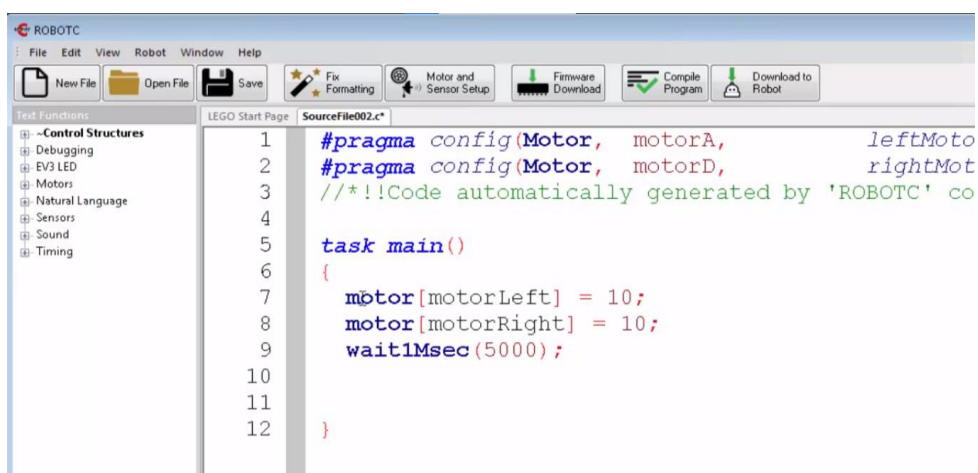


Рисунок 1.7. Программирование на языке RobotC

Рассмотрим пример использования языка программирования RobotC для решения прикладных задач на уроке робототехники [2].

*Напишите программу поворота робота вокруг своей оси на 90 градусов. Не используйте гироскоп и эмпирический подбор данных. Результаты вычислений выведите на экран.*

Для решения задачи учащемуся необходимо применить свои математические познания из области геометрии.

Одно из решений может быть сведено к следующему алгоритму:

1. Определение пути колеса робота за один оборот. Для вычисления пути необходимо значение диаметра колеса умножить на число Пи.

2. Определение пути, который необходимо проделать колесу для поворота на 90 градусов. Робот будет делать разворот вокруг своего центра. Это может быть центр расположения своих колес, вращающихся в разные стороны. В этом случае необходимо измерить расстояние между колесами (данные могут различаться в зависимости от конструкции). Также возможно левый и правый моторы находятся в центре конструкции, посредством зубчатой передачи они управляют каждый своей парой колес, расположенных по бокам. В этом случае нужно определить диаметр окружности, по которой будет проходить вращение робота. Возможно измерение расстояния между первым левым и вторым правым колесом. Поэтому данный путь строго определяется в зависимости от конструкции. Предположим, он равен 23 см. Умножив данное число на Пи, мы узнаем



путь для полного оборота, на 360 градусов. Поделив на четыре, мы вычислим какой путь необходимо проехать роботу, для того чтобы развернуться на 90 градусов.

3. Разделив путь, который необходимо проехать, на путь колеса, вычислим число оборотов колеса. Умножив данное число на 360, получаем данные для разворота в градусах.

Решение на RobotC:

```
float cir = Pi * 6.37; float turnover = PI * 23;  
float t90 = turnover/4; float target = (t90/cir)*360;  
resetMotorEncoder(leftMotor); resetMotorEncoder(rightMotor);  
displayTextLine(1, «%d», cir); displayTextLine(2, «%d», turnover);  
displayTextLine(3, «%d», t90); displayTextLine(4, «%d», target);  
setMotorTarget(leftMotor, target, 50); setMotorTarget(rightMotor, -  
1*target, 50);  
waitUntilMotorStop(leftMotor); waitUntilMotorStop(rightMotor);
```

### Робототехника в среде LEGO Mindstorms EV3: использование виртуальной симуляции.

Использование виртуальной симуляции позволяет учащимся получать образование инженерно-технического профиля в условиях отсутствия учебных наборов, реализации дистанционной образовательной деятельности или для теоретического моделирования определенных процессов и наборов условий, в которых будет работать робот.

На сегодняшний день одной из наиболее распространенных сред виртуальной симуляции в образовательной робототехнике является среда Virtual Robotics Toolkit [3].

Симулятор Virtual Robotics Toolkit поддерживает следующие функции:

- Импорт моделей из LEGO Digital Designer;
- Программирование EV3;
- Базовые поля симуляции (полигоны);
- Создание собственных полигонов (Рисунок 1.8);
- Встроенные в приложение обучающие видео.
- 

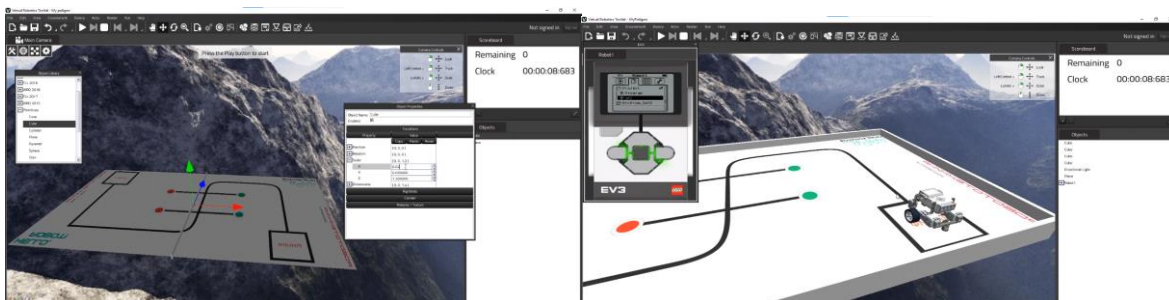


Рисунок 1.8. Создание полигона в среде виртуальной симуляции Virtual Robotics Toolkit

## Робототехника в среде LEGO Mindstorms EV3: конструирование и программирование роботов

Отличной книгой для получения теоретической базы по основам программирования и конструирования в среде LEGO MINDSTORMS EV3 является «Большая книга Lego Mindstorms EV3» Лоренса Валка [4] (Рисунок 1.9). Книга представляет собой полное руководство по изучению среды LEGO MINDSTORMS EV3 и включает в себя ряд интересных практических задач с примерами решений. В книге подробно описаны особенности конструирования роботов с использованием наборов Lego Mindstorms EV3, а также некоторые инструкции по их сборке.

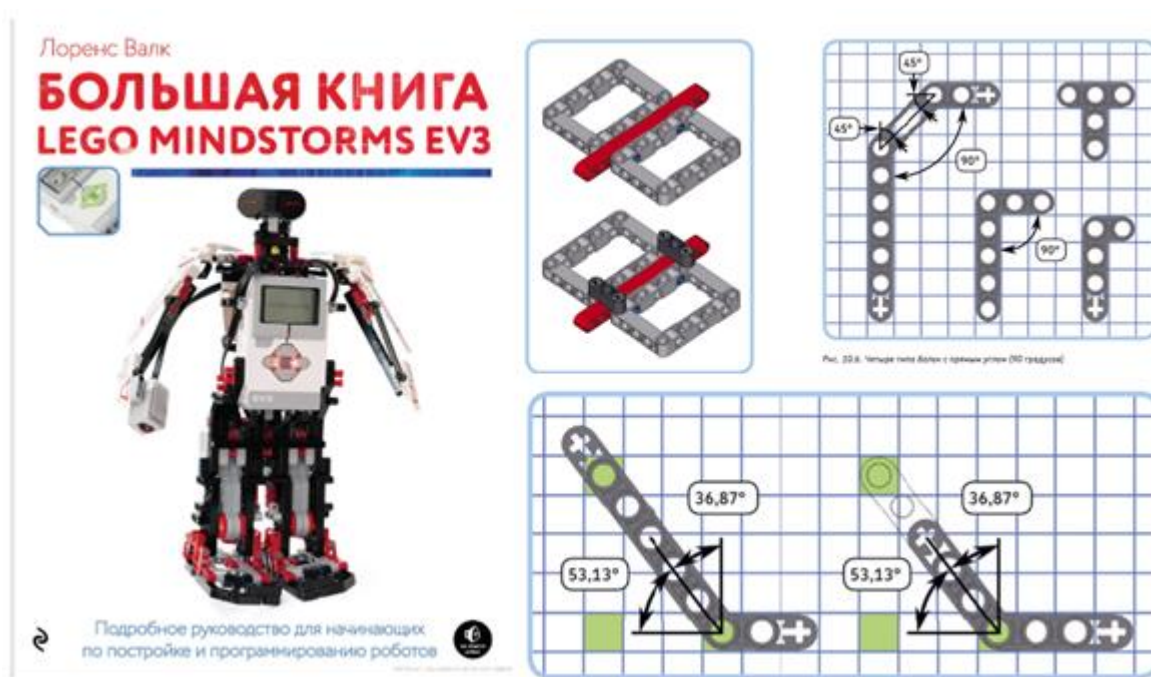


Рисунок 1.9. Иллюстрации из книги «Большая книга Lego Mindstorms EV3»

В источнике [5] приведены примеры задач, активизирующих изучение датчиков и операторов ветвления.

Сайт <https://robot-help.ru> также является примечательным ресурсом для формирования теоретической основы обучения робототехнике. Курс рассчитан на делающих первые шаги в мир робототехники с помощью конструктора Lego Mindstorms. В программе курса довольно подробно описана среда программирования Lego Mindstorms EV3, представлены примеры задач, которые можно использовать на занятиях.

Школа робототехники “Роботек” предлагает следующие системы практико-ориентированных задач для программирования робота:

1. Найди ошибку в программе.
2. Объясни программу.
3. Напиши программу согласно задаче.



Приведем примеры задач согласно вышеуказанной систематизации:

**Пример задачи типа “Найди ошибку в программе”:**

Дано условие задачи: робот должен двигаться вперед, если зажат хотя бы один из двух датчиков касания. Предлагаемая программа имеет следующий вид (рисунок 1.10):

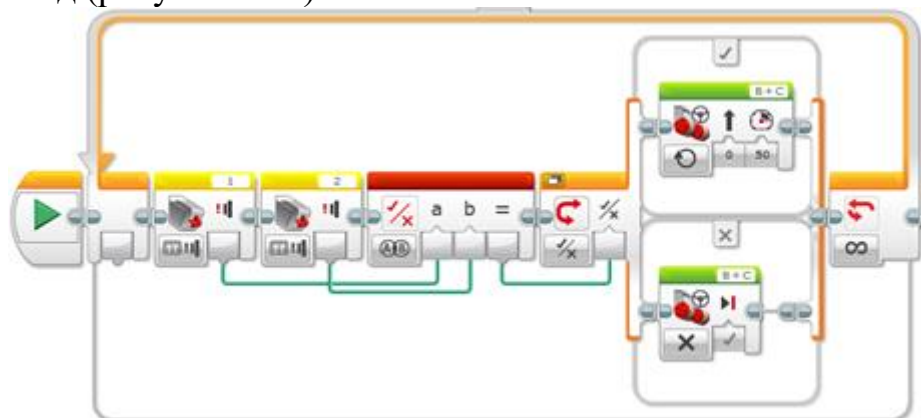


Рисунок 1.10. “Найди ошибку в программе”

Есть ли ошибка в программе? Если да, то какая?

**Пример задачи типа “Объясни программу”:**

Объясните программу (каждую секунду на экране появляется случайное число от 1 до 6.) (Рисунок 1.11). Дополните программу так, чтобы каждую секунду на экране отображалась случайная картинка.

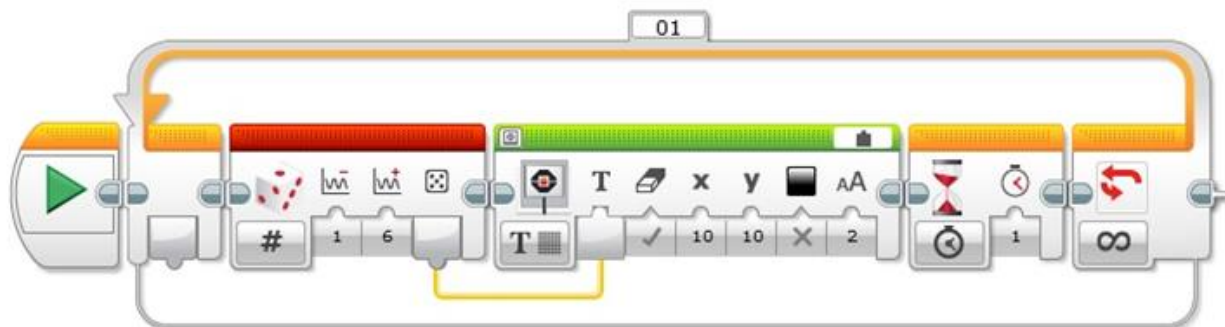


Рисунок 1.11. “Объясни программу”

**Пример задачи типа “напиши программу”:**

Робот спит. Как только датчик замечает яркий свет, робот просыпается, и, если он видит перед собой какой-то объект, начинает вращаться в случайном направлении случайное количество секунд, после чего снова засыпает.

Поощряются попытки учеников решать задачи творчески, при этом необходимо понимание у учащихся важности краткости и простоты написанной программы, согласно принципам чистоты кода. Рациональное

использование логических и математических операторов, циклов, значений датчиков позволяет пользователю решать задачи наиболее оптимальным образом.

Интернет-ресурсы поддержки учителей и педагогов, занимающихся свободной робототехникой:

- <https://education.lego.com/v3/assets/blt293eea581807678a/blt782f3a404152d30c/5f8803a2b703d12407e48b7d/ev3-programming-lesson-plan-rus.pdf> - учебный курс предоставляется в качестве средства обучения с пошаговой инструкцией, содержащей практические советы по использованию мобильного приложения для программирования EV3 в условиях класса;
- [http://kolpduc.tom.ru/files/Robot/UMK/Lego\\_Mindstorms\\_EV3.pdf](http://kolpduc.tom.ru/files/Robot/UMK/Lego_Mindstorms_EV3.pdf) - учебное пособие по программированию в среде Lego Mindstorms EV3
- <https://robot-help.ru/lessons.html> - задача данного курса - познакомить учащихся с конструктором Lego Mindstorms, научить собирать базовые конструкции роботов, программировать их под определенные задачи, разобрать базовые решения наиболее распространенных задач-соревнований. Курс рассчитан на делающих первые шаги в мир робототехники с помощью конструктора Lego Mindstorms. Хотя все примеры роботов в этом курсе сделаны с помощью конструктора Lego Mindstorms EV3, программирование роботов объясняется на примере среды разработки Lego Mindstorms EV3, тем не менее, владельцы Lego Mindstorms NXT тоже могут присоединиться к изучению данного курса.

## **Модуль 2. Спортивная робототехника**

### **Знакомство с соревнованиями**

Инновации в образовании в первую очередь должны быть направлены на создание личности, настроенной на успех в любой области приложения своих возможностей. Олимпиада по робототехнике – это как раз та сцена, где все участники могут продемонстрировать свои навыки и умения, где можно представить себя и поделиться опытом. Возможность участия в таких мероприятиях способствует формированию поколения свободных, образованных, творчески мыслящих людей.

Соревнования по робототехнике - важный составляющий элемент в вопросе повышения мотивации учащихся к обучению, их успеваемости, а также общей и функциональной грамотности.

Существует ряд очевидных преимуществ, которые дает решение олимпиадных задач как форма обучения робототехнике. В качестве основных можно выделить следующие:

- возможность получить новые знания по физике, информатике, электронике, механике и т.д. (особенностью олимпиадных задач является то, что для их решения нужны комплексные знания из области физики, механики, информатики, математики и т.д., поэтому на подготовку к олимпиадам требуется много времени);
- умение решать олимпиадные задачи позволяет легче осваивать школьную программу;
- возможность получить опыт участия в олимпиадах;
- победители и призеры олимпиад I, II, III уровня могут иметь льготы при поступлении в вузы;
- возможность выявить одаренных детей для организации адресной работы с ними;
- большинство серьезных соревнований по робототехнике предполагают умение решать задачи [6].

### **Виды соревнований по робототехнике**

На сегодняшний день в области образовательной робототехники существует огромное разнообразие олимпиад: только в Казахстане можно принять участие в World Robot Olympiad, KazRobotics, Roboland, Vex, First и т.д [7].

Больше информации можно найти на официальных сайтах олимпиад, где подробно описана каждая категория, возрастные группы, регламент, а также порядок проведения самих соревнований:

WRO - <https://wro-association.org/home>;

KazRobotics - <http://www.kazrobotics.org/>;

RoboLand - <http://roboland.kz/>

На сегодняшний день олимпиады по робототехнике находятся на стадии бурного развития и не так популярны, как предметные, и, как следствие, соревновательных этапов в них намного меньше: школьный, областной и республиканский этапы, возможные учебно-тренировочные сборы (УТС), а затем последний, международный этап. Как правило, перерыв между каждым этапом составляет обычно 2-3 месяца, что предоставляет участникам достаточно времени на хорошую подготовку и возможность участвовать в других внешкольных мероприятиях, поддерживать хорошую успеваемость, а также заниматься другими хобби.

Средний возраст учащихся 10 - 18 лет. Всего на сегодняшний момент, с учетом всех возрастных категорий, существует 13 номинаций соревнования по робототехнике. Остановимся на наиболее популярных направлениях:

- основная
- творческая
- футбол
- спортивная категория (лабиринт, кегельринг и др.)

Международная олимпиада по робототехнике WRO в 2021 году сменила направления и названия своих соревнований, добавив новую категорию «Future Engineers».

#### *RoboMission (Основная категория)*

Командам дается определенное задание, которое должно быть выполнено как можно лучше. Задания меняются каждый год в зависимости от темы соревнований. Каждая команда, согласно критериям, должна сконструировать робота, основанного на платформе LEGO, так, чтобы робот смог выполнить поставленную задачу с минимальным числом ошибок.

#### *Future Innovators (Творческая категория)*

С 2021 года на официальном сайте WRO появилась новая категория под названием «WRO Future Engineers» (на русском «Будущие инженеры WRO»). Ее направление – беспилотные автомобили.

В категории «Будущие инженеры WRO» участники будут сосредоточены на развитии навыков в следующих областях:

- Использование компьютерного зрения и сочетания датчиков для оценки состояния парковки и самого транспортного средства;
- Разработка работающего автомобиля с аппаратным обеспечением, таким как электромеханические компоненты и контроллеры, и с открытым исходным кодом;
- Планирование действий и управление роботами с движущимися частями и кинематикой, отличной от дифференциального привода (например, рулевого управления);

- Разработка оптимальных стратегий для выполнения задачи;
- Работа в команде, общение, решение проблем, управление проектами, творчество.

*Спортивная категория (движение вдоль линии, кегельринг, робосумо, лабиринт и т.д.)*

#### *Движение вдоль линии*

Алгоритм движения робота вдоль линии является базовым алгоритмом при изучении робототехники, потому что отражает суть программирования роботов: робот выполняет основную задачу (движение), меняя свое поведение в зависимости от изменений окружающих условий (черные и белые области).

На этапе изучения движения вдоль линии с учащимися 5-класса стоит обратить особое внимание изучению алгоритма движения вдоль линии зигзагом с использованием: 1) одного; 2) двух и более датчиков (Рисунок 2.1). Принципиальным отличием двух методов является избирательность робота с одним датчиком цвета к местоположению относительно линии (справа или слева), в то время как робот с двумя датчиками способен передвигаться вдоль самой линии (середины) [8].

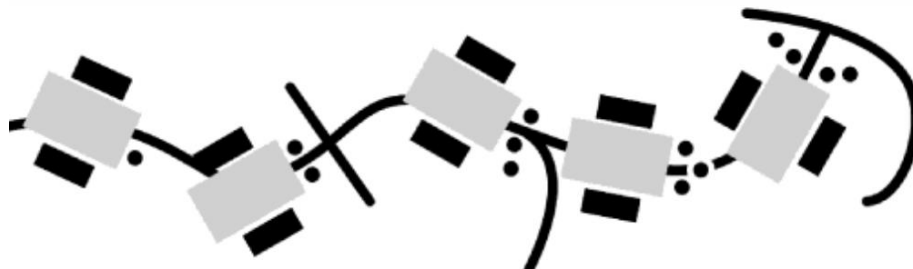


Рисунок 2.1. Варианты расположения робота относительно линии

Плавное движение робота осуществляется путем применения алгоритмов пропорционально-интегрально-дифференциального (ПИД-) регулирования движения. ПИД-регулятор представляет собой управляющий сигнал, являющийся суммой трёх слагаемых, первое из которых пропорционально разности входного сигнала и сигнала обратной связи (сигнал рассогласования), второе — интегралу сигнала рассогласования, третье — производной сигнала рассогласования. Комбинируя составные части, можно добиться разного поведения регулятора. Полученные регуляторы принято называть по имеющимся в них частям: ПИД-регулятор имеет все три компоненты, П-регулятор содержит только пропорциональную, ПД-регулятор содержит пропорциональную и дифференциальную части и т.д. [9]

Важнейшим компонентом ПИД-регулятора является пропорциональная часть, а работа И- и Д- компонентов направлена на дополнение и исправление ее работы. Работа П-регулятора направлена на

стремление робота к желаемому значению определенного датчика, И-регулятор отвечает за стремление робота к сохранению “правильного” состояния тем больше, чем дольше робот пребывает в этом состоянии, работа Д-регулятора направлена на дифференциацию ошибок (чем больше ошибка, тем быстрее робот стремится к исходному правильному положению). Поэтому, наиболее распространенными видами регуляторов являются П-, ПИ-, ПД-регуляторы, а регуляторы типа И- и ИД- на практике практически не используются.

В качестве примера рассмотрим создание программы П-регулятора для движения по черной линии (Рисунок 2.2 - 2.6). В бесконечном цикле считываются показания левого и правого датчика цвета и рассчитывается разность между показаниями правого и левого датчиков. Разность показаний датчиков цвета умножается на коэффициент усиления. Чем больше эта разность, тем больше робот отклонился левым датчиком на черную линию, соответственно, нужно сильнее повернуть направо, чтобы съехать с черной линии. И наоборот, если отклонение отрицательное, то робот въехал правым датчиком на черную линию, а значит, нужно сильнее повернуть налево, чтобы съехать с черной линии.

Формулы для определения мощности моторов EV3 пропорционального регулятора с двумя датчиками:

*Мощность правого мотора = Мощность вперед – разность показаний правого и левого датчиков\*коэффициент усиления*

*Мощность левого мотора = Мощность вперед + разность показаний правого и левого датчиков\*коэффициент усиления*

Коэффициент усиления пропорционального регулятора при движении по линии и скорость движения в направлении вперед задаются в программе исходя из крутизны траектории. Чем круче повороты на траектории, тем больше значение коэффициента усиления пропорционального регулятора.

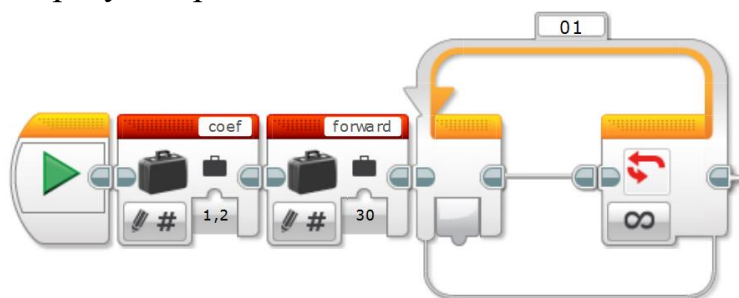


Рисунок 2.2. Программа П-регулятора: инициация переменных и цикл

Сначала вычисляется разность между показаниями датчиков, разность умножается на коэффициент усиления, данное выражение записывается в переменную Error. Математические вычисления осуществляются с помощью блока Математика в расширенном режиме ADV.



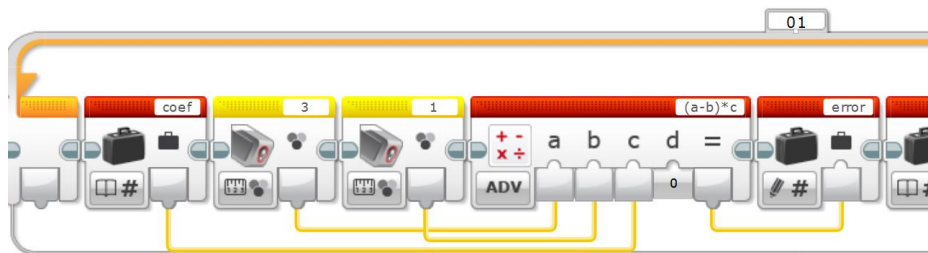


Рисунок 2.3. Программа П-регулятора: вычисление отклонения

Скорость левого мотора рассчитывается по формуле “к скорости движения в направлении вперед прибавляется значение отклонения”:

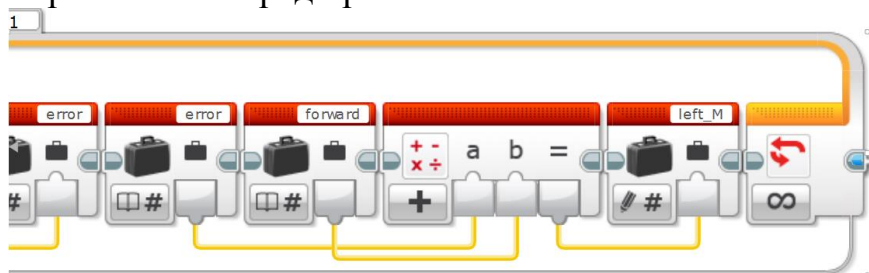


Рисунок 2.4. Программа П-регулятора: мощность левого мотора

Скорость левого мотора рассчитывается по формуле “со скорости движения в направлении вперед отнимается значение отклонения”:

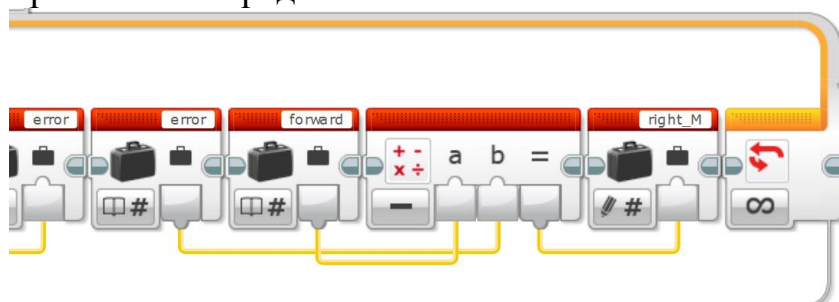


Рисунок 2.5. Программа П-регулятора: мощность правого мотора

Включаем левый и правый моторы с вычисленными значениями мощности:

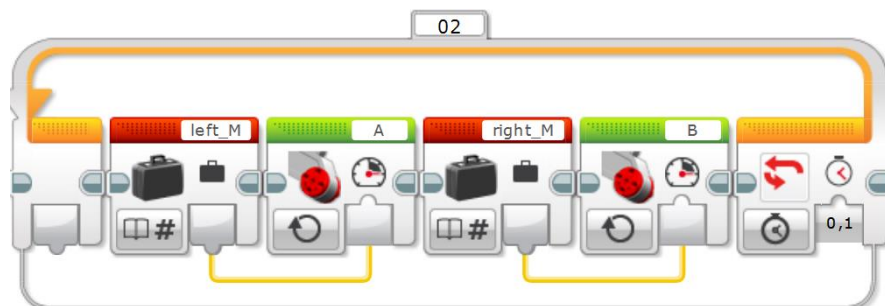


Рисунок 2.6. Результирующее движение по П-регулятору

Примечательной является книга “Алгоритмы и программы движения по линии робота Lego Mindstorms EV3”. Работа подробно описывает тонкости работы с датчиками при движении вдоль линии, представлены варианты прохождения инверсии, перекрестков, штрих-кодов, прерывистой линии и объезда препятствий.

#### *Кегельринг*

Одна из самых простых дисциплин в спортивной робототехнике. Робот стартует в центре круглого белого поля, ограниченного черной линией. По краю поля установлены несколько кеглей. Кеглями, как правило, выступают жестяные банки емкостью 0,33 л, а роль ринга выполняет белое поле, ограниченное черной окружностью с шириной линии 50 мм (Рисунок 2.7). За минимальное время робот должен переместить все кегли за пределы белого поля. Робот не должен выезжать за пределы черной линии ни одной своей частью. Ограничения по размеру робота: робот должен помещаться на лист А4 [10].

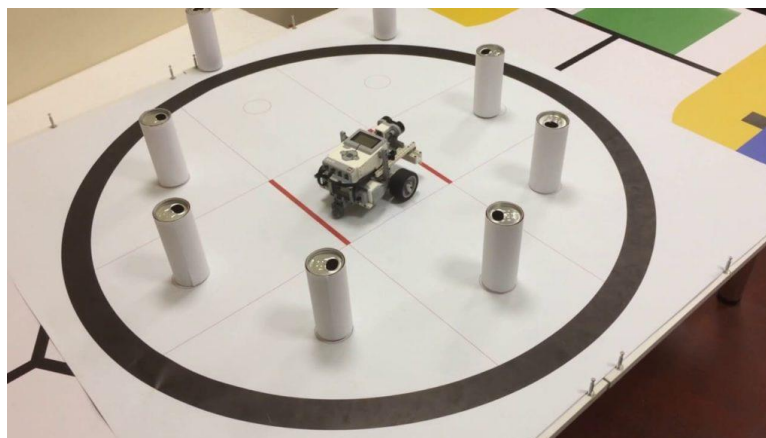


Рисунок 2.7. Поле кегельринг-квадро.

<https://robotportal.ru/zanimatel'naya-robototehnika/lego-ev3-kegelring-kvadro>

Разберем пример разработки ключевого алгоритма поведения робота во время выполнения задания. Для этого рассмотрим следующую поведенческую модель. Робот, находясь в центре поля, начинает вращаться по часовой стрелке, пока не заметит кеглю (Рисунок 2.8, а). Двигаясь в направлении кегли, робот выталкивает её за пределы окружности (Рисунок 2.8, б). Заметив черную границу поля, робот возвращается назад, в место старта (Рисунок 2.8, в).

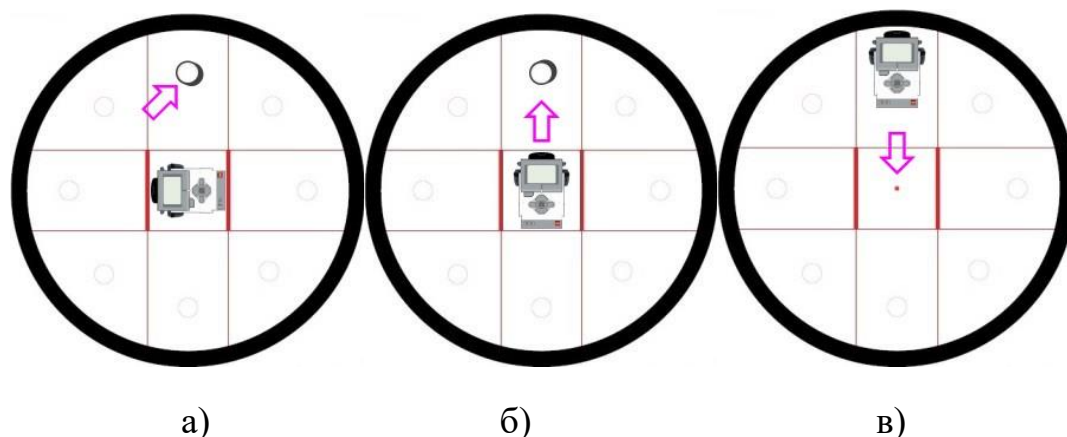


Рисунок 2.8. Положения робота на поле

Отметим, что при решении практических задач по робототехнике успех возможен только при многократном количестве тестирований робота. В процессе тестирования возможно возникновение ситуации, когда вам потребуется вносить изменения не только в программу, но и в конструкцию робота. Важно настраивать учащихся не бояться экспериментировать, не бояться ошибаться. Вдумчивый анализ ошибок обязательно принесет вам и вашей команде большой опыт и положительный результат.

#### *Робосумо*

Наиболее популярным и в то же время довольно простым направлением в соревновании роботов является робосумо. Главная задача робота-сумоиста состоит в том, чтобы вытолкнуть за пределы ринга своего соперника (Рисунок 2.9). Существует множество вариантов правил проведения этого состязания, накладывающих различные ограничения, в первую очередь, на размеры и конструкцию роботов, тем не менее, базовые алгоритмы поведения робота при этом очень похожи.

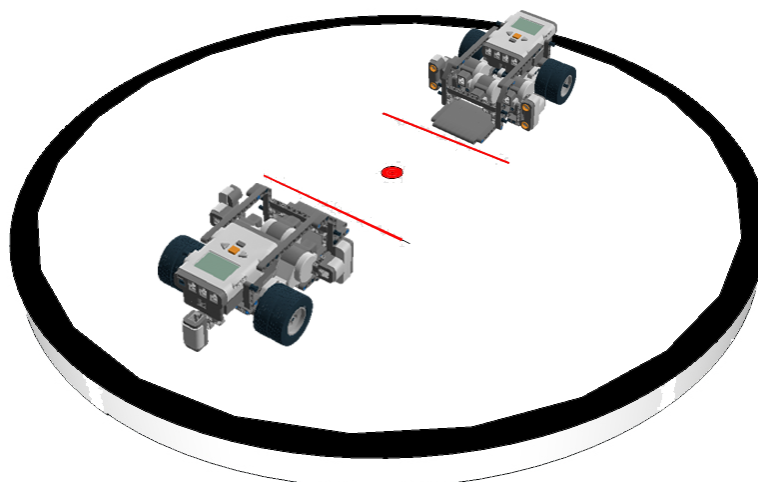


Рисунок 2.9. Робосумо

### *Рекомендации по подготовке к соревнованию робосумо:*

- Изучите и соблюдайте правила соревнования. Помните, что за несоблюдение правил судья имеет право дисквалифицировать вашу команду. При подготовке обращайте внимание на ограничения по размеру и весу робота, на разрешенные виды используемых материалов и технологий. Обратите внимание учащихся на указанные в условиях соревнования особенности поведения робота при запуске (при наличии). К примеру, положение соревнования может установить, что робот не должен двигаться в течение 5 секунд после того, как вы его активировали.
- Детально планируйте алгоритм ведения борьбы на поле. Для этого вам скорее всего понадобится как минимум один датчик цвета (освещенности) для определения белого края поля. Рекомендуется установить два датчика цвета в передних углах вашего робота. Это повысит ваши шансы вовремя обнаружить край в том случае, если робот подошел к нему под углом. Также стоит отдельным образом настроить свое поведение при отступлении, чтобы робот с меньшей вероятностью мог выпасть с поля. Поведение отступления обычно состоит из отступления и поворота назад к центру поля или поиска вашего противника. Также, довольно эффективным может оказаться движение назад относительно датчика, который первым обнаружил край. В редких случаях допускается использование третьего датчика цвета, обнаруживающего край, расположенный позади робота, который, вопреки ожиданиям, часто бывает малоэффективен. Такое расположение датчика может быть полезным в том случае, если в программе вашего робота задан алгоритм движения, который позволяет роботу маневрировать задним ходом, когда задняя часть вашего робота близка к краю. Несмотря на кажущуюся перспективность, данный маневр на практике довольно часто оказывается нереализуемым.
- Максимизируйте вес и плотность робота. Тяжелый вес может обеспечить вам преимущество в тяге и инерции. Довольно часто различные команды используют дополнительные маломощные двигатели для увеличения веса робота. Однако, помните о регламентах соревнования: используйте весы, чтобы убедиться, что вы не превышаете разрешенный вес.
- Стройте компактно. Ваш робот должен быть как можно меньше и плотнее. Воздушные зазоры внутри вашего робота и снаружи должны быть сведены к минимуму. Чем больше ваш робот, тем больше вероятность того, что ваш противник коснется части вашего

робота, находящейся далеко от его центра масс, в результате чего он может легко развернуть вашего робота в другом направлении или вовсе перевернуть его.

- Постарайтесь снизить высоту конструкции робота. Чем ниже ваш центр тяжести, тем меньше вероятность того, что соперник сможет опрокинуть вашего робота или заставить его колеса потерять сцепление с дорогой.
- Постройте «юбку» и защитный каркас — плавный клин, спускающийся от передней части вашего робота к поверхности ринга. Цель состоит в том, чтобы создать некий пандус, который сможет пролезть под вашим противником в тот момент, когда вы вступите в контакт. Клин поднимет вашего противника, передав его вес вашему роботу. В результате ваши колеса могут увеличить сцепление с дорогой, а их — уменьшить. Юбка — это щит, который окружает всего вашего робота, делая его похожим на конус или пирамиду, поэтому он работает везде, где находится точка контакта. Спроектировать юбку зачастую бывает крайне непросто, поэтому будьте готовы оказать помощь ученикам, испытывающим сложности в конструировании. Юбка должна быть прочной, не мешать вашим движениям и работе ваших датчиков. Юбка и защитный каркас также увеличивают размер вашего робота, поэтому в случае, если поле боя сумо расположено на плоскости, риски касания опасного края могут увеличиться. В том случае, если поле находится на возвышенности над плоскостью за рингом, этими опасениями можно пренебречь. Юбка также играет роль стены, построенной вокруг вашего робота, и может быть как слабо закрепленной, так и вовсе не закрепленной с вашим роботом. Наличие юбки может свести к минимуму вероятность наличия бреши, за которую может зацепиться корпус робота соперника. Также ее наличие может спасти робота от атакующего клина соперника, не давая ему возможность поднять корпус вашего робота. Также, вы можете воспользоваться датчиком, который будет определять состояние юбки, и отступить, когда она поднимается.

### *Лабиринт*

В этом состязании участникам необходимо подготовить автономного мобильного робота, способного наиболее быстро добираться из одного конца лабиринта в другой, составленного из типовых элементов, и вернуться обратно (Рисунок 2.10).

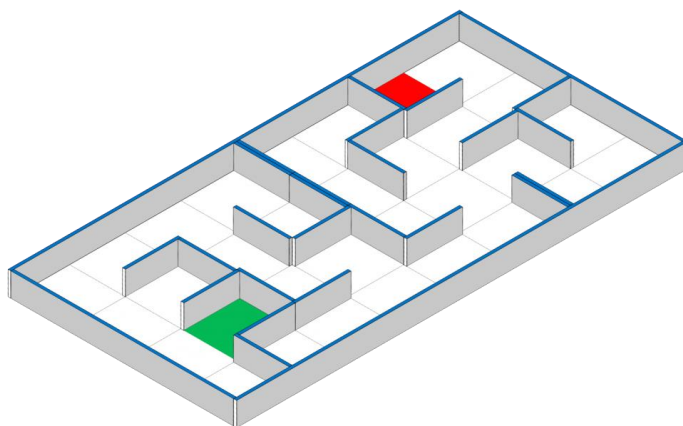


Рисунок 2.10. Лабиринт

При прохождении лабиринта основополагающей концепцией программирования алгоритма движения является так называемые “правила одной руки”: для выхода из лабиринта необходимо, двигаясь по лабиринту, все время касаться правой или левой рукой его стены. Блок-схема движения робота представлена на рисунке 2.11 (Приложение 1) [11]

**Подготовка к соревнованиям. Сборка и программирование роботов по правилам соревнований.**

При подготовке к олимпиаде следует уделять большое внимание и поощрять самостоятельную работу ученика. Самостоятельный творческий поиск является самой эффективной формой подготовки к олимпиаде. Очень важную роль при подготовке школьников к олимпиадам различного уровня играет психологический тренинг. Участник олимпиады испытывает стресс, причем, чем выше уровень олимпиады, тем стресс больше. От правильного настроения зависит не только результат участника, но и его психологическое здоровье. Особое внимание необходимо уделить умениям концентрироваться на главном, не заикливаться на неудачах.

Многими учеными отмечается, что для успешной подготовки школьников к олимпиадам необходимо, во-первых, желание учителя этим заниматься. Вторым условием является наличие увлеченных школьников. Данные условия необходимо дополнить еще одним критерием – целенаправленной подготовкой учащихся к олимпиадам по конкретному предмету [12].

Для подготовки учащихся к соревнованиям предлагаются следующие рекомендации учителям:

1. Уделять большое количество времени устным логическим рассуждениям при решении задач.
2. Изучать с учащимися материал, который не входит в стандартную образовательную программу курса.
3. Необходимо учить учащихся очень внимательно знакомиться с условиями задания.

Важно помнить, что успех связан не только со способностями, но и со знанием классических олимпиадных заданий. Подготовка к олимпиаде по робототехнике подразумевает внеклассную, внеурочную форму обучения. Чтобы подготовить учащихся к участию в соревнованиях и проводить олимпиады, учителю необходимо вести кружки, факультативы, проводить большую подготовительную работу, подбирать и выполнять различные задачи и задания спортивно-соревновательного типа, детально знакомиться с различными вопросами, с новинками литературы. Для подготовки школьников к соревнованиям следует иметь индивидуальный подход к каждому ученику и основной упор делать на самостоятельную работу обучающегося.

Роль преподавателя в подготовке детей к соревнованиям огромна. В первую очередь, преподаватель создает благоприятные условия для того, чтобы ученик мог постигать новое, и с легкостью переживал стрессовые ситуации. С помощью знаний учителя, умения методически правильно поставить перед учеником посильную задачу, и после её решения вызвавшую чувство победы, ученик с большим азартом будет заниматься подготовкой к соревнованиям. Интерес ученика к получению знаний в той или иной области позволяет развить у него нестандартность мышления, что является очень актуальным на данном уровне развития общества. Умение мыслить логически и нестандартно поможет подрастающему члену общества занять достойное место в этом обществе.

Некоторые рекомендации при подготовке учащихся к соревнованиям по робототехнике:

1. Ознакомьтесь и освоите набор/инструменты для робототехники, которые вы будете использовать.

Крайне важно установить возможные проблемы и трудности, которые могут возникнуть в ходе подготовки и самого соревнования. Знание инструмента, предмета соревнования придаст уверенности как вам, так и вашим ученикам. Некоторые соревнования могут требовать приобретения дополнительных деталей или наборов для соревнований, которые также могут требовать дополнительного изучения и подготовки.

2. Уделяйте время сплочению коллектива готовящихся к соревнованиям.

Убедитесь в том, что члены вашей команды хорошо знают своих товарищей по команде. Кроме всего прочего, умение работать в команде — крайне ценный навык, который необходимо развивать всем тем, кто нацелен на STEAM-карьеру в будущем.

3. Используйте возможности ИКТ для подготовки.

Стандартной практикой многих опытных команд является упражнение в виртуальных средах программирования и сборки роботов. Современные реалии развития ИКТ позволяют учащимся и учителям работать и в условиях большого перерыва от занятий. Например, вы



можете создать бесплатную учетную запись на Thaka, чтобы попрактиковаться в программировании роботов Lego Mindstorms EV3/SPIKE Prime.

4. Часто тренируйтесь очно.

Очная реальная практика — лучший способ учебы и тренировок, поскольку учащиеся могут опробовать свои идеи в реальности. Потратьте столько времени, сколько считаете нужным, для того, чтобы ваша команда смогла найти лучшие решения для соревнования. Удобное расположение места подготовки и рациональное планирование времени - залог продуктивной подготовки. Это также дарит отличную возможность участникам команд сблизиться друг с другом и оттачивать свои социальные навыки.

5. Устраивайте дружеские внутренние соревнования для взаимной оценки слабых сторон стратегий каждой из команд/участников.

Если в вашей школе имеется несколько команд, готовящихся к соревнованиям, будет крайне полезным провести дружеское соревнование среди этих команд.

6. Проверьте на практике как можно больше теорий и различных стратегий.

При наличии свободного времени пытайтесь реализовать свои самые яркие и смелые идеи. Как показывает практика, даже самые неоднозначные идеи могут оказаться эффективным способом решения предстоящих задач.

7. Старайтесь быть готовым и подготовить учащихся ко всевозможным внезапным неприятным событиям.

Во время соревнования нередко случаи различных технических неполадок и сбоев. Таковыми, например, могут быть:

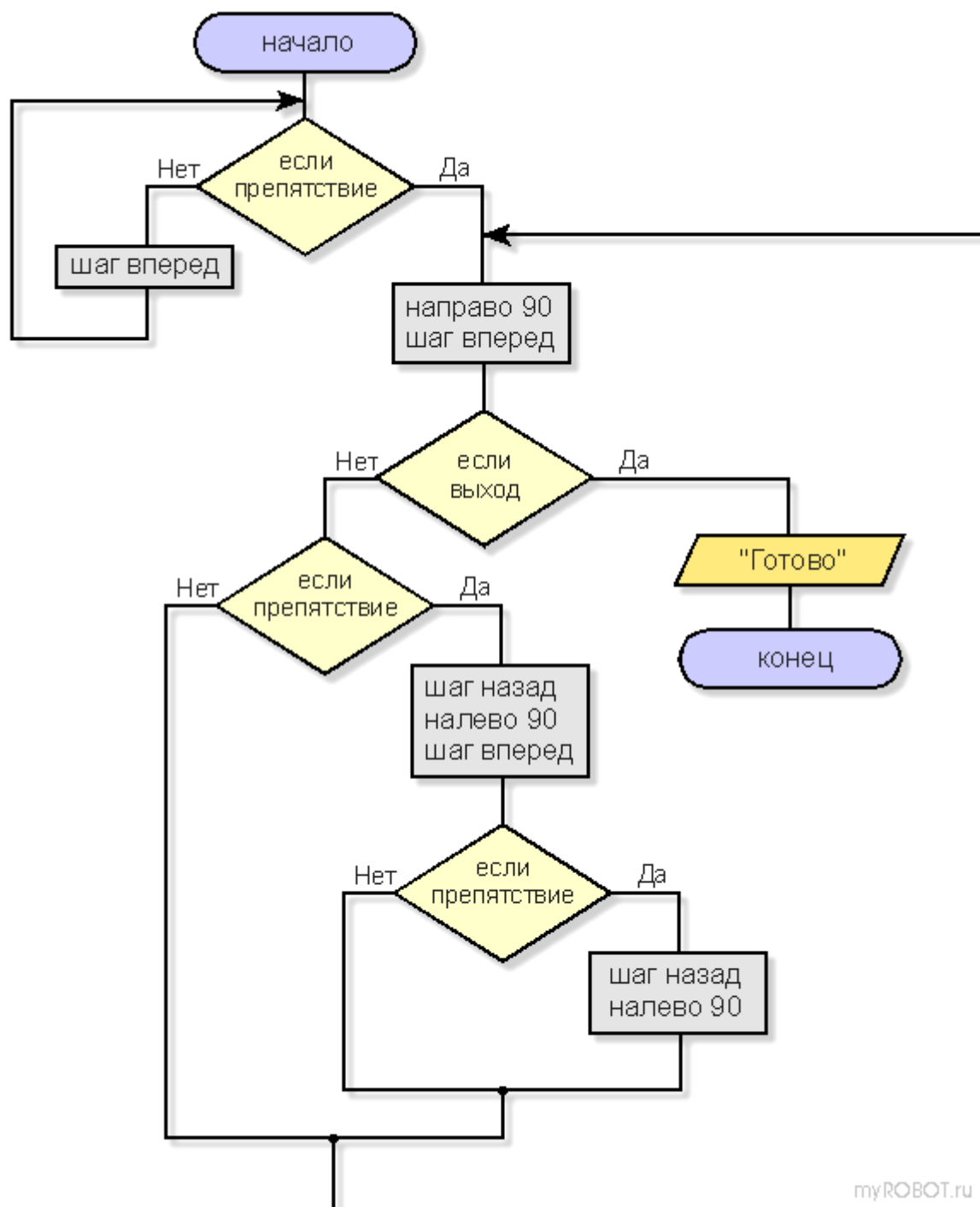
- Поврежденные файлы программ;
- Отсутствие части оборудования;
- Болезнь члена команды в день соревнований;
- Низкий уровень заряда контроллера.

8. При возникновении спорных ситуаций либо при применении новых инновационных приемов решения задач, заранее уточняйте любую неясность у организаторов соревнований/главного судьи.

Опыт победителей олимпиад по робототехнике говорит о том, что готовиться к ним нужно по особой программе, выделять дополнительные часы и разрабатывать индивидуальные траектории для учащихся. Дисциплинированность, желание познавать и творить - вот необходимые качества не только участников, но и их учителей.

## Приложение

Рисунок 2.11. Блок-схема алгоритма для робота, работающего по правилу "правой руки"



myROBOT.ru

### **Модуль 3. 3D моделирование на Sketchup**

К целям и задачам современного технологического образования относятся, в частности, обеспечение понимания обучающимися сущности современных технологий и перспектив их развития, а также формирование технологической культуры и проектно-технологического мышления у обучающихся. Использование трехмерного моделирования на уроках технологии успешно справляется с этой задачей [13].

Трехмерное моделирование – это раздел компьютерной графики, посвященный методам создания объемных объектов в трехмерном пространстве. Это процесс создания трехмерной модели объекта для визуализации и исследования данного объекта.

3D-моделирование прочно вошло в современный мир, область его применения очень широка: это не только реклама, компьютерные игры, фильмы со спецэффектами, но и разработка архитектурных сооружений, техники, транспорта, оборудования в самых различных отраслях. Все это обуславливает актуальность изучения трехмерного моделирования в школе.

Применение 3D технологий на уроках способствует развитию творческих способностей и познавательного интереса учащихся, концентрации их внимания на учебном материале, а некоторых даже ориентирует на выбор технических и инженерных профессий. С помощью трехмерной графики можно разработать визуальный объемный образ желаемого объекта: создать как точную копию конкретного предмета, так и разработать новый, ещё не существующий объект (прототип). 3D-моделирование применяется как в технической среде для создания промышленных объектов, так и для создания эстетических и художественно-графических образов и объектов.

Использование сред трехмерного проектирования на уроках и во внеурочной деятельности способствует развитию интеллектуальных умений и творческих способностей учащихся, а также их практической подготовке к овладению техническими специальностями, помогает ориентироваться в мире высоких технологий.

Учащиеся формируют в себе метапредметные компетенции, образовывая межпредметные связи с математикой, информатикой, физикой, черчением и другими науками. Овладение основами трехмерного моделирования помогает ученикам получить научное мировоззрение, раскрыть свои творческие способности, развить логику и пространственное мышление, научиться применять полученные навыки в своей практической деятельности. Трехмерное моделирование тренирует память и внимание, стимулирует воображение и логическое мышление, воспитывает сосредоточенность и целеустремленность.

Моделируемые объекты выстраиваются на основе чертежей, рисунков, подробных описаний и другой информации.

Актуальность 3D-моделирования объясняется тем, что для создания традиционных (твердотельных) моделей требуется гораздо больше времени, чем для создания цифровых моделей. Таким образом создание цифровых моделей позволяет значительно ускорить процесс проектирования и существенно снизить затраты на создание конечного продукта (объекта), независимо от того, создаем ли мы мультфильм, арт-объект или строим промышленное здание (Рисунок 3.1). Таким образом, в дополнительном образовании выделяется два направления: инженерное 3D-моделирование и художественно-графическое 3D-моделирование.



Рисунок 3.1. Проектирование дома на среде SketchUp

SketchUp (Google SketchUp) - программа для быстрого создания и редактирования трехмерной графики, разработанная компанией @Last Software с 1999 г., поглощенной в 2006 г. Основной идеей SketchUp является простота интерфейса, что позволяет освоить работу с программой даже непрофессиональному пользователю.

**Используемые методы, приемы, технологии. Рекомендации преподавателям.**

Во время обучения моделированию учащиеся сначала выполняют задания учителя, опираясь на его помощь, осваивая процесс создания трехмерных моделей последовательно, по шагам, знакомясь с основными инструментами и технологическими операциями. Постепенно, по мере освоения компьютерного моделирования, учащиеся переходят к более сложным и интересным заданиям, справляясь с ними все в большей степени самостоятельно. Самостоятельное моделирование сложных трехмерных объектов создает ситуацию успеха для учеников, положительно влияет на их самооценку и стимулирует интерес к обучению в дальнейшем.

В течение учебного года у каждого учащегося может образоваться своя уникальная коллекция собственноручно смоделированных объектов, от простых к более сложным. Это наглядно показывает ученикам их прогресс в освоении компьютерного трехмерного моделирования и стимулирует их дальнейший рост в данном направлении.

В SketchUp также существует возможность экспортировать файл с объектами в программу Lumion (Рисунок 3.2), которая позволяет создать видео-презентацию объекта среды, интерьера здания и др. объектов строительства для более детального показа [14].



Рисунок 3.2. Экспорт файлов с использованием программы Lumion



Кроме того, 3D-моделирование открывает широкие возможности для командной работы, ведь проекты, выполненные отдельными учащимися, можно объединять в целостные композиции. К примеру, по итогу окончания учебного года, в качестве самостоятельной работы можно предложить учащимся полностью обустроить комнату либо целый дом. Это, в свою очередь, способствует укреплению командного духа, формированию навыков сотрудничества и коллективного взаимодействия, улучшает психологическую атмосферу в классе.

#### **Рекомендации при работе со SketchUp.**

Для удобства во время работы рекомендуется включить английский язык ввода (и не переключать его). Также, отдельное внимание стоит уделить изучению горячих клавиш для подбора инструментов работы. К примеру, некоторые важные комбинации горячих клавиш инструментов навигации приведены в таблице 3.1.

Таблица 3.1. Горячие клавиши инструментов навигации - Орбита, Панорама, Масштаб, В размер окна.

Команда	Иконка	Клавиатура	Мышь
Орбита		О + движение мышью с зажатой ЛКМ	Зажать колесо прокрутки
Панорама		Н + движение мышью с зажатой ЛКМ	Shift + зажать колесо прокрутки

Масштабирование		Z + движение мышью с зажатой ЛКМ	Вращать колесо прокрутки
Отмена последнего действия		Ctrl+Z	-

При моделировании по образцу или требованиям, важно объяснить, что рисование «на глаз» редко может соответствовать требованиям задачи, и от учащихся требуется соблюдение точных пропорций и размеров.

Следует указать адрес (ссылку в сети либо памяти ПК) электронных пособий, способ их открытия и способ переключения между электронными пособиями и окном SketchUp. [15]

SketchUp предоставляет пользователям ряд возможностей для интересной демонстрации проекта – его презентации.

1. Статичный экспорт каждой страницы как отдельной картинки (File>Export>2D Graphic) с предварительно назначаемыми в окне опций (Options) параметрами. Одно замечание – поскольку фактически экспортируется «снимок с экрана» со стандартным «мониторным» разрешением 72 dpi (а экранное сглаживание линий (Antialiasing), в программе практически отсутствует), назначаем размер картинки по меньшей мере раза в два больше, чем в итоге необходимо. Затем открываем ее в графическом редакторе и уменьшаем до нужного размера (с пропорциональным увеличением разрешения) – за счет эффекта сглаживания получаем намного более качественный результат.

2. Анимированный показ страниц в виде слайдшоу, т.е. последовательной смены кадров с визуальным эффектом плавного перехода между ними. В этом и состоит принципиальное отличие анимации в SketchUp от «классической», где анимируются еще и сами объекты. Параметры показа слайд-шоу настраиваем через Окно>Инфо по модели>Тур (Window>Model Info>Tourguide) двумя значениями в секундах: Временем перехода страниц (Page Transitions) и Временем демонстрации страницы (Page Delay).

Примечание: если модель достаточно сложная, скорее всего переходы между страницами будут чрезвычайно резкими, что совсем не украшает презентацию (особенно при демонстрации ее «заказчику»). Выход из этой ситуации довольно простой – экспорт слайд-шоу в формате видео-файла (File>Export>Animation) (рисунок 3.3).

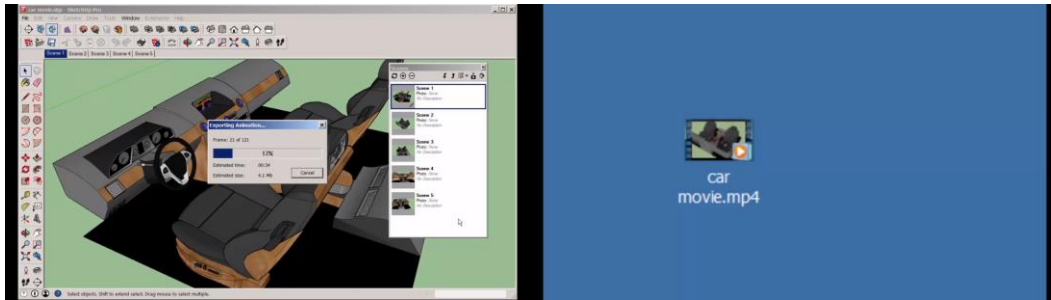


Рисунок 3.3. Экспорт анимации в формате видеопрезентации.

### **Опыты внедрения SketchUp в образовательный процесс учащихся средней школы**

Интересным является опыт Тамбовского государственного университета им. Г.Р. Державина (Российская Федерация). Университетом было разработано учебно-методическое пособие "3D моделирование средствами Google Sketchup в прикладных и гуманитарных исследованиях". В пособии авторы обобщают опыт работы со студентами в среде трехмерного моделирования SketchUp компаний Google и Trimble. Трехмерное моделирование изучается в рамках курсов "Информатика" и "Новые информационные технологии" [16].

В работе рассматриваются инструменты и методы работы в среде трехмерного проектирования SketchUp.

Кажущаяся простота SketchUp позволяет быстро начать самостоятельную работу в нем, однако осознанные результаты могут достигаться при рассмотрении различных типовых конструкций и преобразований. Результата можно достичь разными способами. Рассматривая эти способы, нужно указывать преимущество каждого и направлять студентов к более удобным (по затрате времени) и оптимальным по размеру итогового проекта.

Пособие содержит систематизированное изложение основных понятий и построений. Описаны инструменты для построения плоского эскиза (инструменты "линия", "окружность", "прямоугольник") и работа с размерами. Показан переход в пространство и работа с объемными преобразованиями: выдавливание (инструмент "тяги-толкай"), тела вращения (инструмент "ведение"), работа с текстурированием. Рассмотрены методы прямого построения, построения модели по размерам, копирования и сборки, текстурной обводки. Главы, посвященные основным методам построения в SketchUp, начинаются с описания группы инструментов, участвующих в построении. Показаны пиктограммы инструментов, "горячие клавиши", положение инструментов на рабочей панели и в меню (на русском и английском языках), пошаговая настройка, примеры использования при построении архитектурных элементов и предметов интерьера.



Авторы приводят примеры построений основных трехмерных конструкций с помощью описанных инструментов. В конце каждой из глав приведены лабораторные работы, которые можно использовать для аудиторной и самостоятельной работы студентов.

В качестве показательного образца в работе приведен пример построения памятника архитектуры XX в. с использованием изученных в пособии инструментов. Построение здания начинается с анализа и сбора всех входных данных: фотографий, расчета размеров, подробно показано построение элементов фасада и итоговая сборка здания.

В настоящее время создана электронная версия данного учебного пособия. Отличием от печатного издания являются цветные изображения, а также наличие интерактивных переходов по гиперссылкам. Кроме того, пошаговые инструкции дополнены видеоматериалами, не только дублирующими иллюстративную и текстовую информацию, но и выгодно расширяющими ее. В электронную версию включены в качестве примеров проекты, созданные студентами в SketchUp за последние три года.

Применение системы SketchUp позволяет по-новому строить методику изучения геометрии в начальных классах средней школы, повышая наглядность, позволит более рационально использовать время на уроке, применять дифференцированные подходы в обучении, вносить в урок элементы игры, расширять эрудированность учащихся [17].

Прогнозируемые эффекты от применения данной технологии:

- повышение интереса к изучаемому предмету у слабоуспевающих учащихся;
- повышение уровня самооценки;
- развитие навыка самоконтроля;
- побуждение к открытию и изучению нового в сфере информационных технологий, желанию поделиться с товарищами своими знаниями.

Примеры работ учащихся показаны на рисунке 3.4.

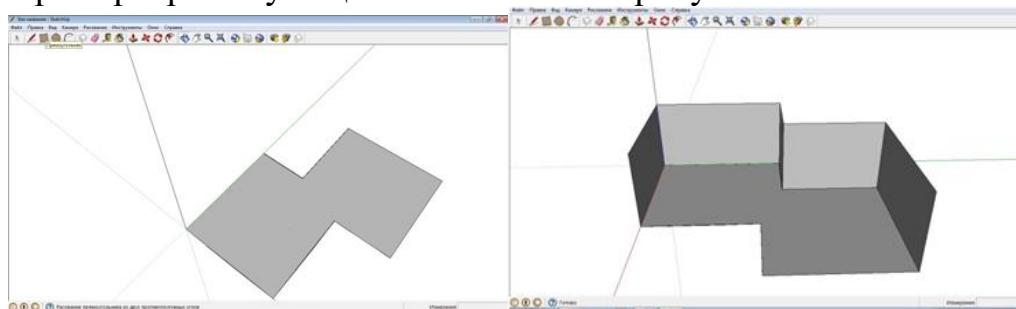


Рисунок 3.4. 3D-модели геометрических фигур, нарисованных учащимися

### **Полезные возможности SketchUp**

SketchUp предоставляет своим пользователям возможность удаленного взаимодействия с учащимися. Поскольку результатом

обучения по всем содержательным темам образовательной программы является электронная трехмерная модель, такие работы могут выполняться дистанционно при условии организации занятий и консультирования обучающихся средствами интерактивной видеосвязи. В случае отсутствия у учащихся возможности установки приложений, необходимых для выполнения проекта, есть возможность использовать веб-приложение Sketchup: <https://app.sketchup.com/app?hl=ru>.

Отправка выполняемых заданий преподавателю в этом случае может быть осуществлена по электронной почте.

Часть занятий, если позволяет уровень технической грамотности обучающихся, может быть организована в режиме отложенного времени, когда формулировки заданий рассылаются по электронной почте, обучающиеся их выполняют и присылают файлы для проверки учителю. Вопросы в этом случае также могут задаваться по электронной почте.

Учебно-методическое пособие “Методические рекомендации использования программы SketchUp на уроках геометрии” содержит рекомендации по применению на уроках геометрии программы «SketchUp». Согласно авторам, использование 3D-моделирования в качестве инструмента решения геометрических задач способствует формированию межпредметных связей у учащихся, а также повышает их мотивацию к изучению предмета [18].

#### **Полезные ресурсы для преподавателей**

- 3D-Skills: компьютерное моделирование и конструирование (<https://stepik.org/course/107176/promo?search=1182245048>).

Курс содержит в себе три модуля, один из которых посвящен теоретическому материалу. Два других модуля посвящены практическим вопросам моделирования, в которых на примере конкретных сервисов будут рассмотрены основные инструменты и операции для создания моделей (SketchUp, Blender). Курс может быть встроен в программу внеурочной деятельности, посвященной компьютерному моделированию.

- Мир 3D: SketchUp - сайт для начинающих свое изучение среды SketchUp с подробным описанием инструментов среды и дополнительными заданиями по проектированию. Курс включает в себя ряд обучающих видео и практических материалов в текстовом формате и включает в себя следующие модули (Рисунок 3.5):
  - SketchUp для школ: введение;
  - 3D-моделирование для класса геометрии;
  - Моделирование учебного класса;
  - Дизайн детской площадки;
  - 3D-моделирование и математика;
  - 3D-карты на уроке географии;
  - Дизайн сцены в театре;
  - Моделирование клетки живых организмов;
  - Эскиз декорации для представления;
  - Моделирование декоративного скворечника;
  - Руководство по 3D-печати SketchUp для школ;
  - Логотип SketchUp.

Ссылка на сайт:

(<https://sites.google.com/site/mir3dsketchup/zanatie-16-17-razrabotka-proekta>)

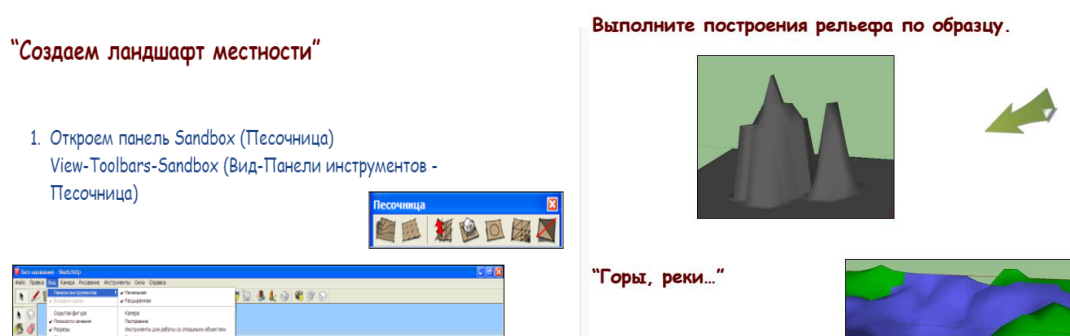


Рисунок 3.5. Содержимое курса “Мир 3D: SketchUp”

- Шаблоны занятий на среде SketchUp для школ. Данный ресурс представляет собой перевод методической разработки Темпла Грандина, автора известных курсов и статей по 3D-моделированию. Ссылка на занятия:

[https://docs.google.com/presentation/d/19OqC7vHfzYvr07FRc6lB\\_Lx3WOF\\_b4Um2RwBhS5bPek/htmlpresent](https://docs.google.com/presentation/d/19OqC7vHfzYvr07FRc6lB_Lx3WOF_b4Um2RwBhS5bPek/htmlpresent).

**Ссылки на официальные ресурсы от разработчиков:**

- Официальный сайт онлайн-среды 3D-проектирования SketchUp - <http://sketchup.google.com/>
- Google 3D-модели - <http://sketchup.google.com/3dwarehouse/>
- Бесплатный плагин “Driving Dimensions” для параметризации в SketchUp - [http://plmpedia.ru/wiki/Driving\\_Dimensions](http://plmpedia.ru/wiki/Driving_Dimensions)

- Руководство по работе со средой разработки SketchUp “SketchUcation” - <http://www.sketchucation.com/>
- <https://www.sketchup.com/education/sketchup-for-schools/curriculum/app> - пошаговое изучение среды SketchUp на английском языке. Инструкции представлены в виде презентаций Google Docs (Рисунок 3.6).

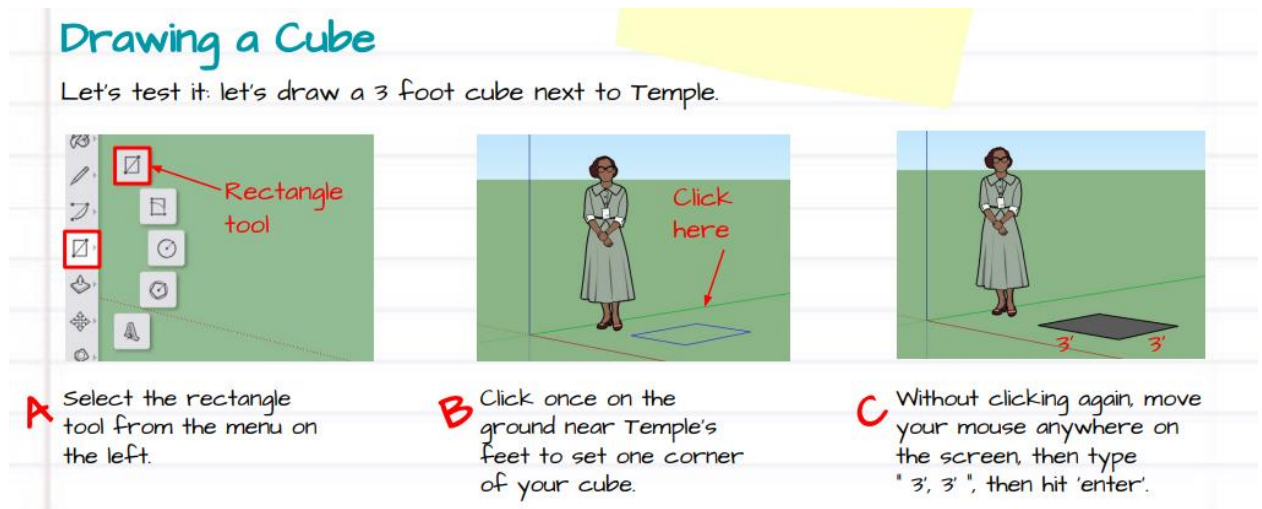


Рисунок 3.6. Отрывок из занятий курсов SketchUp for Schools

- Плагин SketchUp Blog - <http://sketchuptips.blogspot.com/>
- Российский ресурс о SketchUp - <http://sketchup.ru/>.

## Модуль 4. 3D моделирование и 3D печать

SketchUp (Google SketchUp) - программа для быстрого создания и редактирования трехмерной графики, разрабатываемая компанией @Last Software с 1999 г., поглощенной в 2006 г. Основной идеей SketchUp является простота интерфейса, что позволяет освоить работу с программой даже непрофессиональному пользователю.

Программа реализует концепцию прямого моделирования геометрии, в рамках которой пользователь сначала стоит плоский контур из имеющихся примитивов, затем вытягивает его с целью создания или вычитания объема, после чего придает модели нужную форму посредством перетаскивания ее элементов (вершин, ребер и граней) с помощью указателя мыши.

SketchUp поддерживает экспорт и импорт различных форматов трехмерной и растровой графики. В программе имеются библиотеки компонентов, которые можно пополнять своими элементами, и библиотека материалов (Рисунок 4.1).

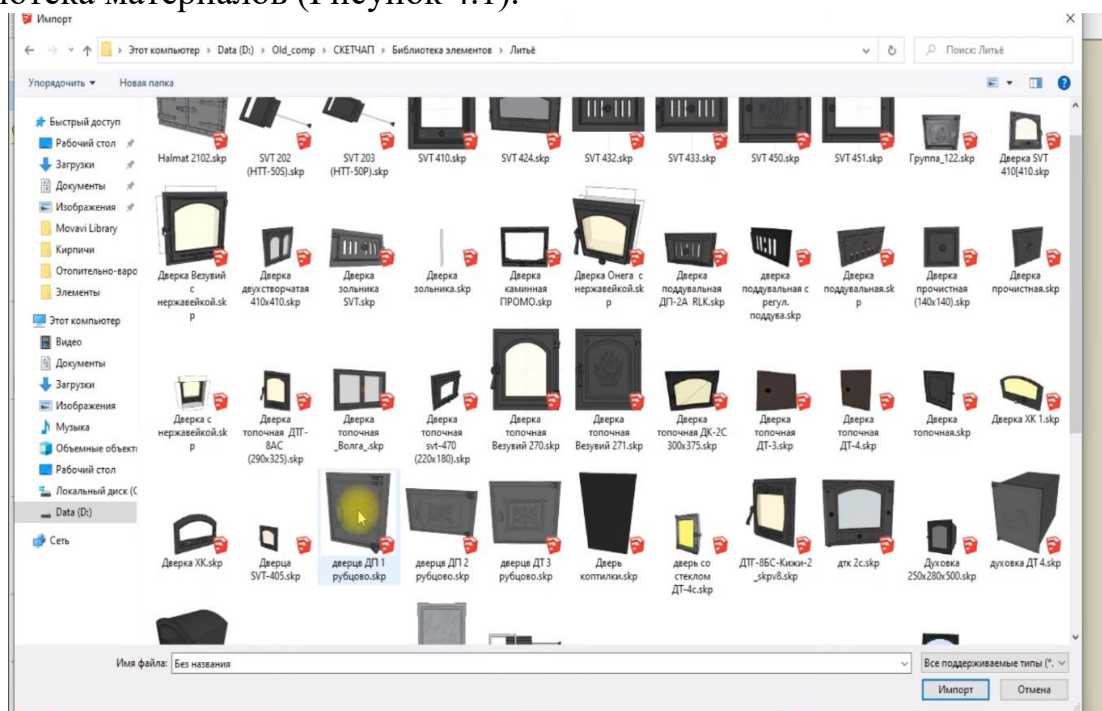


Рисунок 4.1. Библиотека элементов SketchUp

Компания Google поддерживает библиотеку трехмерных моделей (зданий, мостов, машин, мебели, людей, животных, вымышленных персонажей и проч.) с бесплатным доступом к поиску и добавлению новых моделей.

Преимуществом программы SketchUp является то, что на данный момент она поддерживает огромное количество подключаемых модулей (плагинов) от других разработчиков, созданных с помощью программного интерфейса на языке Ruby. Программы на Ruby являются платформенно независимыми и интерпретируются во время исполнения. Существует

возможность интеграции в них бинарного кода, скомпилированного для конкретной целевой платформы. Именно относительная простота изучения и широкие горизонты перспектив изучения программы SketchUp делают ее крайне удобным инструментом изучения 3D-моделирования в школе [19].

Цель изучения 3D-моделирования на основе SketchUp в начальных классах основной школы - создание условий для использования учащимися современных информационных технологий при моделировании конструкторских изделий с проектированием и изготовлением деталей на 3D принтере.

Учебный курс основной школы по 3D-моделированию ставит перед собой следующие цели:

Образовательные:

- Освоить базовые компетенции в области проектирования, моделирования и конструирования.
- Владеть умением представлять форму проектируемых объектов.
- Приобрести навыки моделирования с помощью современных программных средств.
- Освоить навыки 3D печати.

Личностные:

- Формирование творческой инициативы при разработке технических устройств.
- Развитие таких важных личностных компетенций как память, внимание, способность логически мыслить и анализировать, концентрировать внимание на главном при работе над проектами.
- Расширение круга интересов, развитие самостоятельности, аккуратности, ответственности, активности, критического и творческого мышления при работе в команде, проведении исследований, выполнении индивидуальных и групповых заданий при конструировании и моделировании механизмов и устройств.
- Выявление одаренных детей, обеспечение соответствующих условий для их образования и творческого развития.

Метапредметные:

- Создание и обеспечение необходимых условий для личностного развития, профессионального самоопределения и творческой реализации в инженерной сфере.
- Формирование способности задавать вопросы о применимости привычных законов для решения конкретной инженерной задачи, развитие критического отношения к готовым рецептам и образцам, стремления к улучшению уже существующих устройств и создания улучшенных аналогов.

Формы организации деятельности учащихся на занятии:

- индивидуальная;

- групповая;
- фронтальная;
- индивидуально-групповая;
- работа по подгруппам (по звеньям).

Рекомендуемые форматы занятий: круглый стол, семинар, лабораторное занятие, мастер-класс, соревнование, викторина, «мозговой штурм», выставка, занятие-игра, турнир, защита проектов, практическое занятие, презентация, конкурс, консультация, конференция, ярмарка и др. [20]

### **Работа на среде SketchUp. Технические и методические рекомендации**

SketchUp имеет несколько версий, платную Pro и бесплатную Make – полнофункциональную в плане моделирования, но с ограничениями по экспорту-импорту в другие форматы. SketchUp Make - 3D-редактор, который подходит для создания зданий и простых объектов. (Рисунок .4.2)

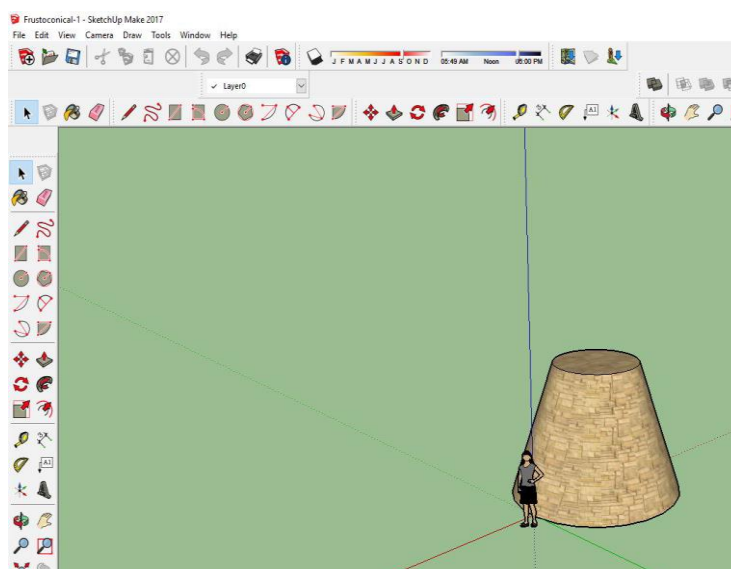


Рисунок 4.2. Основное окно программы SketchUp

Редактор поддерживает наложение текстур, материалы, а также интеграцию моделей в программу Google Планета Земля. Это бесплатный редактор аналогичный 3Ds Max. Все нарисованные детали могут в будущем быть распечатаны на 3D принтере. Редактор знакомит школьников с координатами на плоскости, масштабированием объектов, фигурами вращения. Редактор позволяет построить интересные трехмерные объекты, надеть, например, текстуру «афиша» на объект «тумба». (Рисунок 4.3)





Рисунок 4.3. Афиша на тумбе на SketchUp

Для учащихся часто бывает интересно создавать видео с визуальной подвижной демонстрацией полученных в ходе моделирования объектов. Выбирая подходящий ракурс объектов при помощи создания нескольких сцен для детального рассмотрения объектов проекта, уточнив формат экспорта, пользователь может получить анимацию. Файл имеет формат \*.avi.

Другой очень важной особенностью SketchUp является доступность двух репозиторий: 3D Warehouse и Extension Warehouse. Первый (3dwarehouse.sketchup.com) содержит огромную базу 3D моделей, созданных другими пользователями, в который можно загружать и свои смоделированные объекты. Второй – репозиторий расширений (extensions.sketchup.com), в котором содержится огромное количество инструментов, безгранично расширяющих и дополняющих стандартные инструменты для построения, входящие в базовый набор программы [21].

#### **Итоговый контроль учащихся**

Предлагаемые формы итогового контроля:

1. Презентация и защита мини-проектов с применением методики использования 3D-прототипирования и моделирования на учебных занятиях в инженерных классах.

2. Итоговое тестирование.

Презентация и защита мини-проектов с применением методики использования 3D-прототипирования и моделирования на учебных занятиях в инженерных классах

В процессе презентации и защиты мини-проекта должны быть раскрыты следующие структурные компоненты (требования):

а) тема разрабатываемого мини-проекта;

б) место урока/занятия или фрагментов уроков в контексте образовательной деятельности педагога, работающего в инженерном классе (методический комментарий);

в) цели, задачи урока/занятия или фрагментов уроков;

г) этапы и формы организации деятельности учащихся;

д) планируемые результаты;

е) возможные технологические и методические трудности и пути их решения;

ж) дополнительные методические комментарии к представленному мини-проекту, в т.ч. подборка приложений и презентация.

Презентация и защита мини-проекта – это его представление (регламент – 8 минут) 1–2 докладчиками от каждой группы (всего 5 групп) и последующие ответы на вопросы. Участники группы отвечают на вопросы своих коллег из других групп и модератора семинара (регламент – до 10 минут). В рамках представления и защиты мини-проекта слушателями курсов должны быть продемонстрированы:

- владение методикой использования 3D-прототипирования и моделирования на учебных занятиях в инженерных классах;
- рефлексия способов и результатов собственных профессиональных действий;
- способность к критическому осмыслению работ, представленных коллегами.

### **Опыт зарубежных и отечественных преподавателей**

Оригинальной идеей, побуждающей познавательный интерес у учащихся, является изучение основ построения 3D графики, начиная с моделирования оригами – объемных поделок из бумаги. Само по себе создание оригами из бумаги – это интересный творческий процесс, развивающий пространственное мышление, алгоритмическое и логическое мышление, творческий подход к решению задач, память и внимание. 3D моделирование фигурок оригами помогает дополнительно развить навыки работы в 3D графических редакторах, а также способность строить пространственные схемы с помощью информационных технологий, разрабатывать новые схемы, учиться переносить и воспроизводить информацию о реальных объектах в электронном виде в формате 3D моделей.

При реализации любой идеи приходится сталкиваться с рядом сложностей и противоречий, которые могут существенно повлиять на конечный результат. Так, при освоении 3D графики могут возникнуть следующие противоречия:

- 3D моделирование кажется школьникам достаточно интересным и увлекательным занятием, однако практические приёмы 3D моделирования могут быть достаточно сложны для восприятия и понимания;

- уровень технической и интеллектуальной подготовки различных учащихся может сильно отличаться, что может повлиять как на познавательный интерес, так и на процесс изучения трёхмерной графики и на его результаты;
- единые стандарты и строгие требования к уровню освоения 3D графики, которые могут осилить далеко не все учащиеся.

В связи с вышеперечисленными противоречиями встаёт исследовательский вопрос: какие педагогические технологии применять для оценки условий обучения 3D графике и дальнейшего применения методов и приемов обучения с целью повышения, развития и укрепления познавательного интереса учащихся к этой деятельности. Педагогические методы должны быть подобраны так, чтобы вызвать первоначальный интерес, затем закрепить его на практике и усилить, побуждая школьников самостоятельно переходить к более сложным задачам 3D моделирования, в результате решения которых у них будут вырабатываться необходимые познавательные и творческие навыки [22].

В работе [23] приведен пример использования программы SketchUp для реализации проектной деятельности учащихся основной школы. Учащиеся использовали программу SketchUp для моделирования различных физических явлений - от вращения планет по орбите до измерения жесткости пружины (Рисунок 4.4).

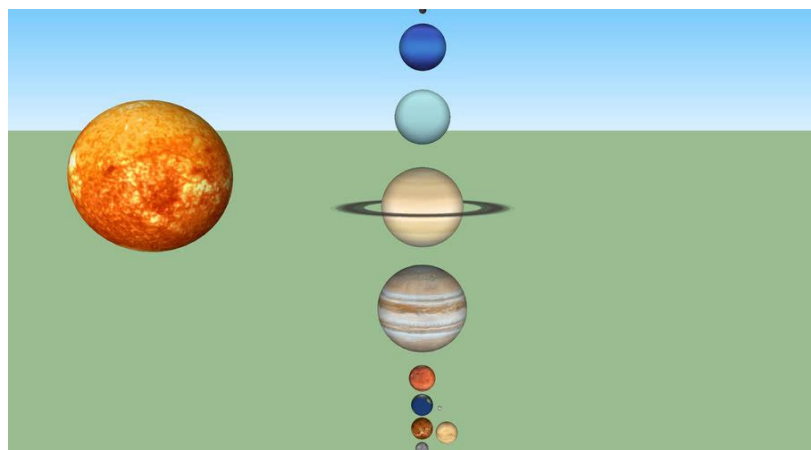


Рисунок 4.4. Модели планет из библиотеки SketchUp

Реализация проектной деятельности учащихся состояла из следующих этапов:

**I. Подготовительный этап:**

1. Разработка тематики проектов.
2. Подбор учителем технологий под тематику проектов.
3. Поиск методических рекомендаций по применению 3D-технологий.

4. Постановка проблемы, целей и задач проекта, возможного результата и способа его достижения, составление плана в форме беседы, обсуждения.

### **II. Основной этап:**

1. Подбор учеником технологий под тематику индивидуального проекта.

2. Изучение методических рекомендаций по применению 3D-технологий.

3. Консультации по овладению 3D-технологиями.

4. Консультации по самому итоговому проекту.

### **III. Заключительный этап.**

1. Индивидуальное сопровождение по оформлению и защите индивидуального проекта в форме консультаций.

2. Педагогическая поддержка обучающегося в ходе защиты проекта, обсуждение перспектив развития.

Таблица 4.1. Система сопровождения проектной деятельности учащихся в процессе обучения с использованием 3D технологий

Этап	Содержание	Цели	Методы	Результат
1	1. Разработка тематики проектов. 2. Подбор учителем технологий под тематику проектов. 3. Поиск методических рекомендаций по применению 3D-технологий. 4. Постановка проблемы, целей и задач проекта, возможного результата и способа его достижения, составление плана	Заинтересовать учащихся, подобрать тему в соответствии с его уровнем подготовки и развития	Анализ (тем, технологий, ресурсов интернета)	Тематика проектов, база методических рекомендаций, выбор учащимися тем проектов и разработка индивидуального плана проектной деятельности
2	1. Помощь в подборе технологий под тематику	Стимулировать обучающихся к самостоятельному изучению	Сравнение, анализ	Оптимально подобранная 3D-технология и реализация

	<p>индивидуального проекта.</p> <p>2. Изучение методических рекомендаций по применению 3D технологий.</p> <p>3. Индивидуальное сопровождение по овладению 3D-технологиям.</p> <p>4. Индивидуальное сопровождение по выполнению итогового проекта</p>	<p>рекомендаций по применению 3D-технологий и выполнению индивидуального проекта</p>		<p>плана</p>
3	<p>1. Индивидуальное сопровождение по оформлению и защите индивидуального проекта.</p> <p>2. Педагогическая поддержка обучающегося в ходе защиты проекта, обсуждение перспектив развития</p>	<p>Научить верному оформлению итогового проекта, подготовить к яркой и запоминающейся защите индивидуального проекта</p>	<p>Сравнение, анализ</p>	<p>Защита итогового проекта</p>

В работе [24] приведен пример лабораторных работ, посвященных изучению моделирования на SketchUp. Лабораторный практикум позволит развить познавательную активность школьников, творческое и операционное мышление, а также повысит интерес к информатике и к профессиям, связанных с компьютерной графикой и дизайном.

Данный практикум:

- 1) знакомит учащихся с основными инструментами и приемами создания и обработки изображений в программе Google SketchUp;
- 2) дает представления о принципах работы трехмерного редактора;
- 3) научит создавать 3D-изображения путем использования набора инструментов, имеющихся в Google SketchUp;

- 4) способствует повышению интереса к информатике и компьютерной графике;
- 5) стимулирует учеников к дальнейшему изучению путем включения соревновательных элементов (проектная индивидуальная и командная деятельность)

Предполагаемые результаты обучения:

После выполнения лабораторного практикума по трехмерной графике учащиеся освоят современный программный продукт эскизного 3D- моделирования, который позволяет в дальнейшем перейти к любому профессиональному графическому программному пакету (Рисунок 4.5).

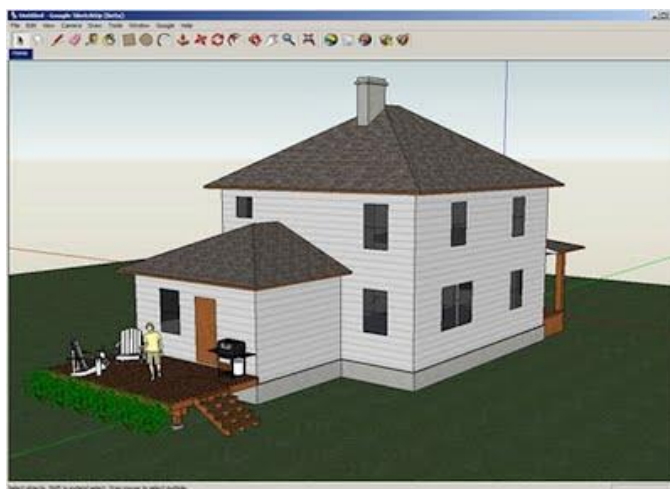


Рисунок 4.5. Дом с ландшафтом, выполненный на SketchUp

Практикум состоит из 6-ти лабораторных работ, выполняемых в компьютерном классе. Объем лабораторного практикума – 10 часов (таблица 4.2)

Таблица 4.2. Поурочный план лабораторного практикума

№	Название	Количество часов
Лабораторная работа №1	Знакомство со средой графического редактора Google SketchUp (интерфейс и инструментарий, приемы моделирования геометрических фигур)	1
Лабораторная работа №2	Моделирование дома. Создание стен, окон и дверей дом	1
Лабораторная	Моделирование интерьера:	1

работа №3	Работа с цветом; проработка деталей окна	1
	Проработка деталей двери	1
	Моделирование занавесок	1
	Моделирование мебели (стола, стульев)	1
	Построение прочих предметов	1
Лабораторная работа №4	Моделирование экстерьера. Построение крыши и крыльца дома; наложение текстур	1
Лабораторная работа №5	Моделирование ландшафта. Построение поверхности земли и деревьев	1
Лабораторная работа №6	Доработка деталей	1

Использование 3D-принтеров формирует у учащихся целый ряд практических навыков в моделировании, технологии, физике, биологии, математике, программировании. 3D-печать – это мощный образовательный инструмент, который может привить обучающемуся привычку не использовать только готовое, а творить самому.

Трехмерная графика получает все большее применение в различных отраслях и сферах деятельности, знание которой становится все более необходимым для полноценного развития личности.

Освоение технологии трехмерного моделирования является основным стартом для тех обучающихся, кто, возможно, захочет связать свою жизнь со сферой производства, строительством, транспортом, в военных и инженерных профессиях и в рабочих специальностях.

Систематизированный и грамотный подход в обучении детей 3D-моделированию является важным шагом в сторону выбора будущей профессии ребенка. На этих занятиях рождаются будущие конструкторы, мультипликаторы, дизайнеры, инженеры. Все эти профессии очень актуальны на сегодняшний день и будут всегда востребованы в будущем.

#### **Полезные ресурсы для преподавателей:**

1. <https://www.sketchup.com/ru/plans-and-pricing/compare/extend-sketchup-pro-desktop-functionality> - Сайт позволяет пользователям установить различные плагины для расширения функционала SketchUp Pro для ПК.
2. <https://gsketchup.ru/uroki/> - курс для начинающих 3D-моделирование в SketchUp, представляющий собой серию видеоуроков, обучающую



проектно-ориентированным методом, соблюдая принцип “от простого к сложному”: подробно рассматривается пример проектирования дома, с последующим анализом плагинов и библиотек SketchUp.

3. <https://www.sketchup.com/education/sketchup-for-schools/curriculum/app> - пошаговое изучение среды SketchUp на английском языке (Рисунок 4.6). Занятия представлены в виде презентаций Google Docs.



Рисунок 4.6. Курсы на сайте “SketchUp Education”

## Модуль 5. Программирование микроконтроллера MicroBit (MicroBit и средой программирования MakeCode; )

Микробит (англ. BBC micro bit или micro:bit) - одноплатный компьютер, разработанный по инициативе корпорации Би-би-си совместно с крупными технологическими компаниями, сообществами и образовательными организациями для ведения образовательной деятельности в области программирования и стимулирования технического творчества учащихся. Компьютер был разработан в комплекте со средами программирования. По данным BBC Media Centre, в 2018 году всего было продано более 2 млн экземпляров по всему миру.

### Возможности среды MakeCode.

Печатная плата содержит в себе программируемый микроконтроллер, с возможностью подключения различных навесных оборудования (Рисунок. 5.1).



Рисунок 5.1. Платы BBC micro:bit

Автономная версия онлайн-редактора MakeCode доступна бесплатно в Microsoft Store. Среде MakeCode представляет собой платформу на блочном языке программирования. Пользователи, ознакомившиеся с блоками и принципами их работы, могут переключиться на кодирование более сложных программ на языке JavaScript, который широко используется для программирования веб-страниц или MicroPython. На рис. 5.2 показано блочное программирование в MakeCode. На рисунке 5.3 показано программирование на языке JavaScript.

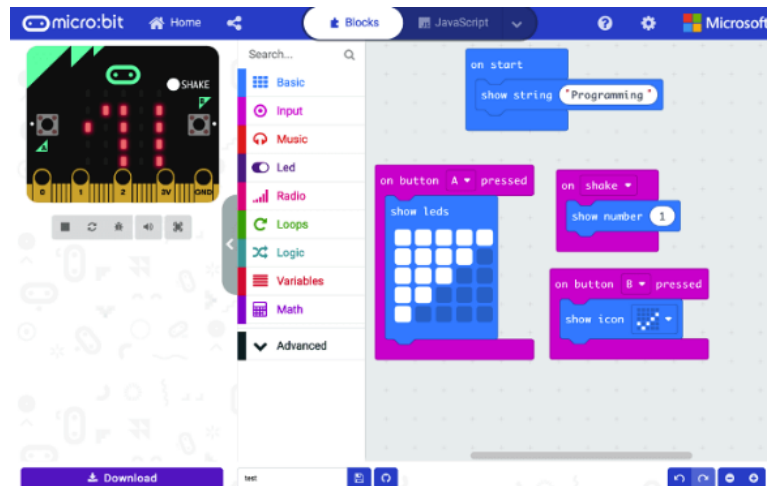


Рисунок 5.2. Блочное программирование на среде MakeCode

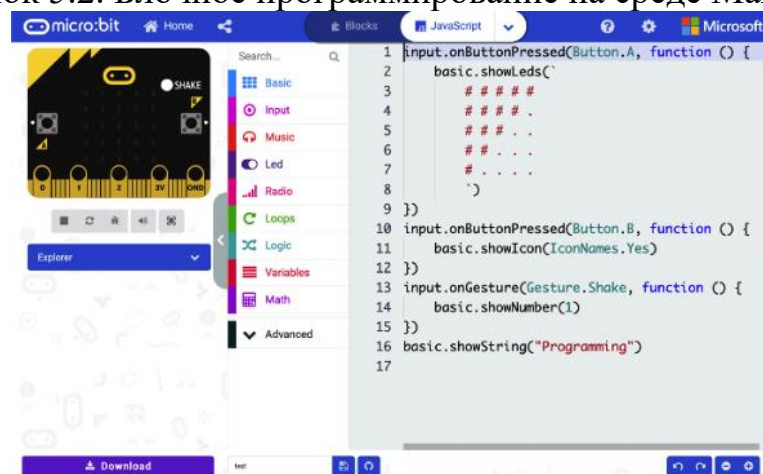


Рисунок 5.3. Использование JavaScript в редакторе MakeCode

На рис. 5.4 показано последнее нововведение в платформе MakeCode – программирование в MicroPython. Платформа, как было отмечено ранее, представляет пользователям возможность переключаться между языками и способами программирования. Редакторы MakeCode содержат симулятор, который можно использовать для отображения результата кода, не требуя для этого самой платы Microbit. Пользователь также имеет возможность «перевода» кода с одного языка на другой: код, написанный на блочном языке, можно «перевести» на JavaScript или MicroPython.

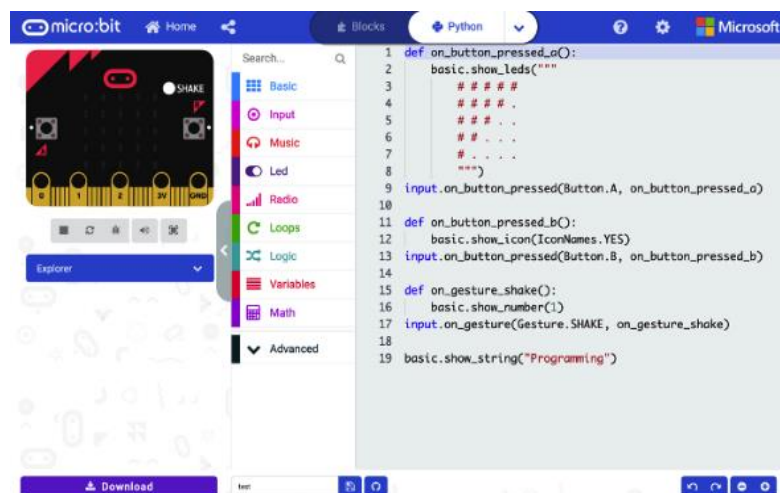


Рисунок 5.4. Использование MicroPython в редакторе MakeCode

### Использование MicroBit в образовании.

В процессе проведения занятий рекомендуется практико-ориентированный метод обучения по принципу «от практики к теории», который заключается в том, что инструменты среды программирования MakeCode, а также основы алгоритмизации, программирования и технологии разработки проектов на занятиях изучаются на практических примерах реализации отдельных функциональных возможностей разработанного проекта или программы, без детальных теоретических объяснений.

Таким образом, у учащихся формируются практические навыки и доступные для понимания представления о требуемом результате. В зависимости от года обучения, уровня подготовки, интересов и способностей учащихся, учителю предлагается использовать одну или несколько стратегий проектно-ориентированного обучения.

Коллектив авторов из университета Матея Бела предлагает ряд задач для учащихся, имеющих предыдущий опыт работы со схожими принципами работы программными средами и языками.

Примеры задач:

- Событийно-ориентированное программирование: после нажатия кнопок А, В, А+В, встряхивания платы microbit и т.д., светодиодная матрица microbit (5x5) отображает различные значки (сердце, да/нет, счастье, футболка и т.д.), тексты, числа, стрелки.
- Математические операторы – после встряхивания microbit отобразит случайное число (кости), чтобы использовать мин/макс. значение, округление и т. д.
- Простые игры с хранением данных в переменной или массиве (увеличение, уменьшение числа после некоторых событий) «камень-ножницы-бумага».

- Светодиодная подсветка – подключение светодиода через зажимы типа «крокодил» и резисторы.

- Беспроводная связь – отправка текста/номера с одной платы microbit на другую плату.

В качестве дополнительных инструментов при обучении можно использовать такие расширения как воспроизведение звуков, подключение светодиодных лент (Рисунок 5.5), набор инструментов stop:bit.



Рис. 5.5. Подключение светодиодов Neopixel

Одним из главных преимуществ использования светодиодов является возможность использовать выходы данных с microbit. Учащиеся знакомятся с принципами передачи данных от платы к Neopixel, контактами питания и контактами-основами.

Примеры задач на использование светодиодных лент:

- Создайте программу, меняющую цвет светодиода в следующем порядке, с интервалом ожидания перед изменением в одну секунду - красный, синий, фиолетовый, индиго, оранжевый.

- Для каждого светодиода в ленте задайте свой цвет (используйте циклы и массивы данных).

- После встряхивания microbit отображает случайный цвет для каждого светодиода в ленте.

- В блоке “Всегда” светодиод меняет 5 цветов, между показом цветов устанавливается блок ожидания (используйте переменную, с инициализирующим значением в 100 мс). Если кнопка A нажата, добавьте к



этому времени +100 мс. Если кнопка В нажата, сделайте вычитание этого времени до -100 мс (Рисунок 5.6).

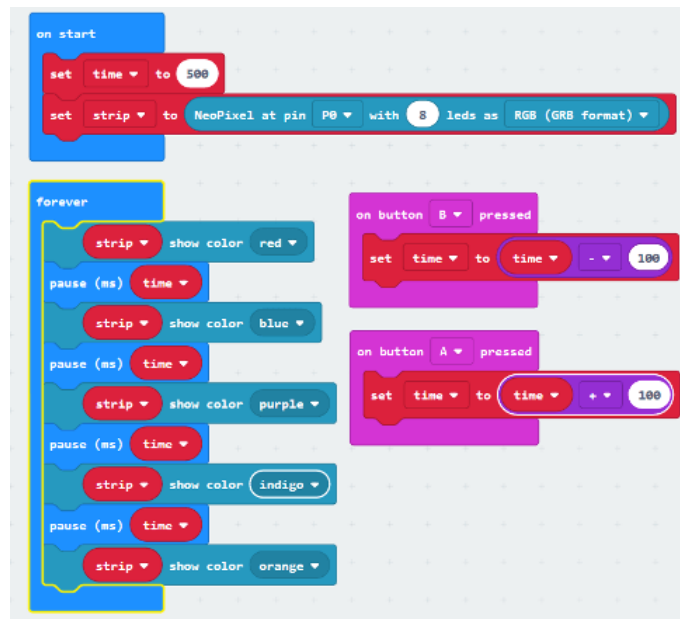


Рис. 5.6. Код для расширения NeoPixel

Еще одним доступным, и в то же время интересным для учащихся дополнением к плате являются динамики (динамик должен иметь наконечники типа 3,5 мм JАСК и две клипсы типа «крокодил»). Разъем динамика имеет два металлических контакта для левого и правого динамика, и третий – для заземления (Рисунок 5.7).



Рис. 5.7. Плата MicroBit с динамиками

Пример заданий с использованием динамиков:

- Создать мелодию по требованиям задания (с учетом тонов, нот, высоты и длительности воспроизведения).
- Создайте свою собственную музыку (пример указан на рисунке 5.8).
- Импортируйте звуковой файл.

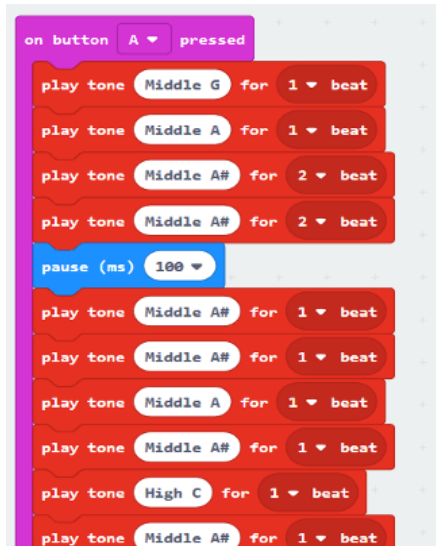


Рис. 5.8. Код для задания с использованием динамиков

Stop:bit — расширение для MicroBit, имитирующее поведение светофора (Рисунок 5.9). Состоит из двух основных частей, первая из которых соединена с MicroBit с помощью четырех винтов, а вторая часть — это три светодиода, каждый из которых управляется отдельным контактом MicroBit.



Рисунок 5.9. Расширение Stop:bit

Для упрощения настройки к среде программирования Microsoft Makecode можно добавить расширение “Kitronik-stopbit”, где устройство имеет собственные блоки <https://novator.team/post/1344>.

Несколько рекомендаций для использования расширения Stop:bit:

- В качестве единого светофора рядом с пешеходным переходом;
- Использование нескольких таких светофоров в качестве сети светофоров для управления движением в городе.
- В качестве индикации любых процессов, имеющих до трех вариантов состояния, например “вывод случайного числа от одного до трех”.



- В качестве визуального сопровождения к музыкальному ритму.

Пример заданий с использованием расширения Stop:bit:

Задание 1. После установки радиосвязи загорается зеленый либо красный свет: учащийся использует функцию радио и отправляет текстовую строку... Stop:bit включает на светофоре цвет в соответствии с принятой строкой. В случае получения сообщения с текстом «green» загорается зеленый свет, после получения текста «red» загорается красный. Также, MicroBit при нажатии (Рисунок 5.10):

- Кнопки А – отправляет текстовое сообщение «green».
- Кнопки В – отправляет текстовое сообщение «red».

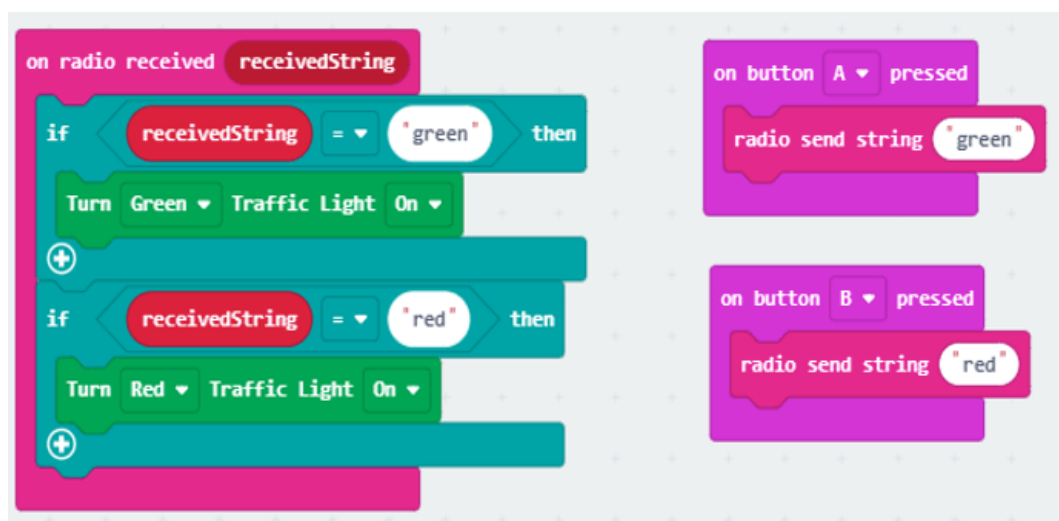


Рис. 5.10. Включение зеленого и красного цветов по сигналу радио

Задание 2 – викторина по математике: учащийся отвечает на вопросы, говорит или показывает на доске. Учитель устанавливает правильный ответ через свой MicroBit. Используя функцию радио в MicroBit, студент отправляет ответ учителю. В зависимости от правильности ответа, Stop:bit реагирует соответствующим цветом (Рисунок 5.11).

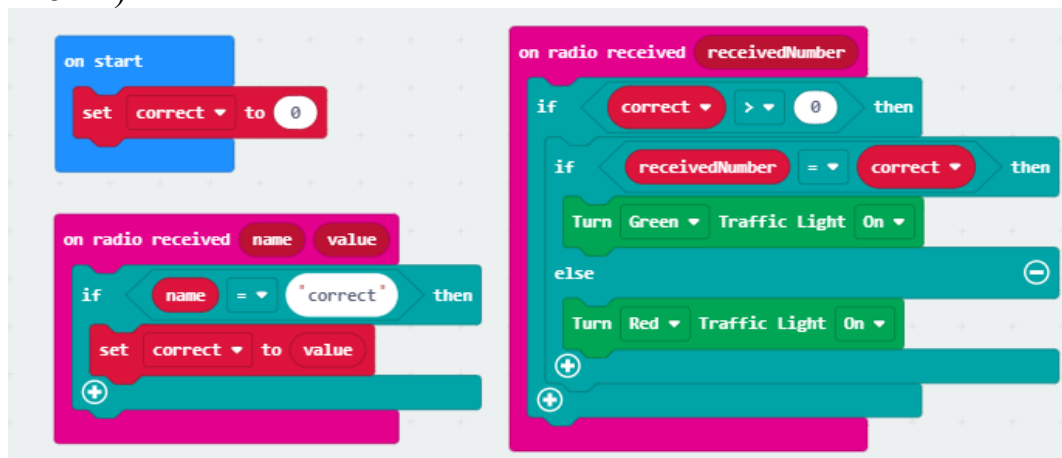


Рисунок 5.11. Пример кода для математической викторины

### Проектная деятельность с использованием MicroBit.

Несмотря на то, что плата MicroBit ориентирована на учащихся школьного возраста, она широко используется для создания проектов разного уровня сложности. На базе платы можно собрать и умные часы с фитнес-трекером, и мобильных роботов на колёсной платформе. Приведем примеры использования платы MicroBit в разработке проектов.

Проект “Музыкальная перчатка”. Интересной идеей использования MicroBit может быть реализация проектной деятельности. В 2020 году в киберфестивале “Rukami” был представлен проект “Muse Gloves” - музыкальная перчатка, способная управлять midi-клавиатурой, звуковыми и световыми эффектами на сцене (Рисунок 5.12). Аппаратная часть состоит из BBC: MicroBit, резисторов изгиба, датчиков давления (нажатия), гироскопа. Программная часть реализована на языке Java и в стандартном приложении «Microsoft Make Code» [25].



Рисунок 5.12. Музыкальная перчатка “Muse Gloves”

Проект “Шагомер” (Рисунок 5.13). Принцип работы основан на «считывании» жеста «встряхивания», получаемого во время выполнения шага, который использует несколько показаний датчика акселерометра. На сайте <http://stem.ort.org/microbit/> представлены примеры проектов и пошаговые инструкции их реализации.



Рисунок 5.13. Шагомер.

**Microbit в Казахстане.** Казахстанская образовательная компания Stemkit предоставляет своим учащимся возможность проектирования и программирования роботов с использованием наборов Qbit, созданных на основе платы Microbit. В настоящий момент курс содержит более 30 готовых уроков, 3 уровня сложности и более 100 экспериментов [26].

#### **Полезные ресурсы для преподавателей**

1. ORT-STEM (<http://stem.ort.org/microbit/>) - сборник учебно-методических материалов по использованию микроконтроллера BBC micro:bit в STEAM-образовании. Сборник содержит в себе планы уроков, рекомендации по проведению проектов, интересные идеи и материалы.
2. M-ROBOT (<https://mrobot.by/>) - сайт содержит серию уроков по основам программирования на среде MakeCode с использованием Microbit.
3. Microsoft-MakeCode (<https://makecode.microbit.org/courses>) - сборник руководств, созданных для MicroBit.
4. Книга “BBC micro:bit. Официальное руководство пользователя” - полное официальное руководство по среде Microbit, предназначенное для самостоятельного обучения.

## Источники

- 1) Методика обучения учащихся решению задач по робототехнике с использованием сборника задач. Маслакова Е.А., - вып. квал. раб. по напр. «44.03.01 – Педагогическое образование», профиль «Информатика». – Екатеринбург, 2017. – 63 с.
- 2) RobotC — кросс-робототехнический язык программирования [Электронный ресурс]. – Режим доступа: <http://news.scienceland.ru/2019/04/21/robotc-%D0%BA%D1%80%D0%BE%D1%81%D1%81-%D1%80%D0%BE%D0%B1%D0%BE%D1%82%D0%BE%D1%82%D0%B5%D1%85%D0%BD%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9-%D1%8F%D0%B7%D1%8B%D0%BA-%D0%BF%D1%80%D0%BE%D0%B3%D1%80/> (дата обращения: 28.07.2022)
- 3) Виртуальная робототехника в реальности: большая подборка программ [Электронный ресурс]. – Режим доступа: <http://edurobots.ru/2020/05/virtual-toolkits/> (дата обращения: 28.07.2022)
- 4) Электронная книга “Большая книга Lego Mindstorms EV3”. Режим предпросмотра. [https://books.google.kz/books/about/%D0%91%D0%BE%D0%BB%D1%8C%D1%88%D0%B0%D1%8F\\_%D0%BA%D0%BD%D0%B8%D0%B3%D0%B0\\_LEGO\\_MINDSTORM.html?id=IUY4DwAAQBAJ&printsec=frontcover&source=kp\\_read\\_button&hl=ru&redir\\_esc=y#v=onepage&q&f=false](https://books.google.kz/books/about/%D0%91%D0%BE%D0%BB%D1%8C%D1%88%D0%B0%D1%8F_%D0%BA%D0%BD%D0%B8%D0%B3%D0%B0_LEGO_MINDSTORM.html?id=IUY4DwAAQBAJ&printsec=frontcover&source=kp_read_button&hl=ru&redir_esc=y#v=onepage&q&f=false)
- 5) Методические рекомендации по образовательной робототехнике. Сборник 1. /Ассоциация инженерного образования детей; Томский физико-технический лицей. - Томск: Изд-во Томского физико-технического лицея, 2017. - 25 с.
- 6) С.Н. Родин. - Олимпиадные задачи как одно из направлений обучения детей соревновательной робототехнике. - Новомосковск, 2021 г.
- 7) Daniil Filimonov, Elina Pak - [Прелесть робототехники] Олимпиада по робототехнике: вдоль и поперек/ Статья из сайта <https://blog.bc-pf.org/>
- 8) Овсяницкая Л.Ю. Алгоритмы и программы движения по линии робота Lego Mindstorms EV3 / Овсяницкая Лариса Юрьевна, Овсяницкий Д.Н., Овсяницкий А.Д. - М: Издательство “Перо”, 2015. - 168 с.
- 9) ПИД-регулятор доступно и просто [Электронный ресурс]. – Режим доступа: <http://edurobots.ru/2020/02/pid-robo/> (дата обращения: 10.08.2022)

- 10) Михаил Д. Робототехника Диченко. Курс на сайте <https://stepik.org/learn/> - 2021 г.
- 11) ПРОХОЖДЕНИЕ ЛАБИРИНТА: ПРАВИЛА И АЛГОРИТМЫ [Электронный ресурс]. – Режим доступа: [https://myrobot.ru/articles/logo\\_mazesolving.php](https://myrobot.ru/articles/logo_mazesolving.php) (дата обращения: 28.07.2022)
- 12) Сумин Д.В. ПОДГОТОВКА УЧАЩИХСЯ 5-7 КЛАССОВ К ОЛИМПИАДАМ ПО РОБОТОТЕХНИКЕ // Материалы VI Международной студенческой научной конференции «Студенческий научный форум» URL: <https://scienceforum.ru/2014/article/2014005050> (дата обращения: 28.07.2022).
- 13) Нарымская О. Е. - Использование трехмерного моделирования на уроках технологии. Санкт-Петербург, - 2019.
- 14) Бахмисова, М. А. Программа SketchUp для 3D-моделирования и дизайн-проектирования объектов интерьера / М. А. Бахмисова // Развитие науки и образования: монография. – Чебоксары: Общество с ограниченной ответственностью «Издательский дом «Среда», 2019. – С. 80-90.
- 15) Кусакин И.И. РАБОЧАЯ ПРОГРАММА учебного курса ЦТПО «Основы проектирования в Google SketchUp 8» - Москва, 2018.
- 16) Кончаков Р. Б. Учебно-методическое пособие "3D моделирование средствами Google Sketchup в прикладных и гуманитарных исследованиях" / Р. Б. Кончаков, М. Ю. Сидляр // Информационный бюллетень ассоциации История и компьютер. – 2014. – № 42. – С. 190-191.
- 17) Применение программы SketchUp на уроках математики. [Электронный ресурс]. – Режим доступа: <https://infourok.ru/primenenie-programmi-sketchup-na-urokah-matematiki-3974586.html> (дата обращения: 28.07.2022).
- 18) Газимагомедова А.О. Методические рекомендации использования программы SketchUp на уроках геометрии. Махачкала, 2018.
- 19) Мир 3D: SketchUp. [Электронный ресурс]. – Режим доступа: <https://sites.google.com/site/mir3dsketchup/> (дата обращения: 28.07.2022)
- 20) Шмидт А.А. Дополнительная общеобразовательная общеразвивающая программа технической направленности «3D моделирование». - 2020.
- 21) Зенкина, С. В. Использование 3D редакторов в урочной и во внеурочной деятельности / С. В. Зенкина, М. В. Савченкова // Инфоком. – 2018. – № 1(2). – С. 45-53.

- 22) Сырицына, В. Н. 3D-моделирование как средство развития познавательного интереса к информатике в средней школе / В. Н. Сырицына, О. Е. Кадеева // *Фундаментальная наука и технологии - перспективные разработки : материалы XXI международной научно-практической конференции, North Charleston, USA, 23–24 декабря 2019 года.* – North Charleston, USA: LuluPress, Inc., 2020. – С. 19-22.
- 23) Сенькина, Г. Е. 3D-моделирование в проектной деятельности обучающихся: построение системы сопровождения в процессе изучения физики в общеобразовательной школе / Г. Е. Сенькина, А. А. Базылева // *Системы компьютерной математики и их приложения.* – 2019. – № 20-2. – С. 209-214.
- 24) Резцова А. И. Обучение трехмерной компьютерной графике в средней школе / А. И. Резцова // *Инновации. Наука. Образование.* – 2019. – № 10(11). – С. 6.
- 25) Черёмухин П. С., Шумейко А. А., Комиссарова О.В., Рожкова Е.А. - Организация инновационной проектной деятельности учащихся основного общего и среднего общего образования: методические рекомендации. Комсомольск-на-Амуре: изд-во “Инженерная школа города Комсомольска-на-Амуре”, 2020. - 100с.
- 26) Сайт Stem Solutions [Электронный ресурс]. – Режим доступа: <https://stemkit.kz/qbit> (дата обращения: 28.07.2022)

## Программирование робота на языке программирование Python

Для программирования в робототехнике необходимо в начале углубиться в понимание языка Python (Рисунок 1) и основы синтаксиса.



Рисунок №1. Логотип Python

Python — это активно развивающийся скриптовый язык, который используют для решения большого объема самых разноплановых проблем и задач. Python пригодится в создании компьютерных и мобильных приложений, его применяют в работе с большим объемом информации, при разработке web-сайтов и других разнообразных проектов, используют в машинном обучении.

Python применяют в различных сферах. Разработчики со всего мира полюбили этот язык программирования за множество возможностей.

1. Объектно-ориентированное программирование (ООП). Реализация ООП в Python хоть и специфичный по сравнению с иными объектно-ориентированными языками, но одновременно является неплохо продуманным.
2. Обобщенное и функциональное программирование.
3. Модули и пакеты. Программное обеспечение (ПО) на Питоне оформляется в виде модулей, которые могут быть собраны в пакеты.
4. Интроспекция. Это возможность запросить тип и структуру объекта во время выполнения программы. В Питоне можно узнать всю информацию о внутренней структуре любого объекта.
5. Обработка исключений, итераторы и генераторы.
6. Управление контекстом выполнения.
7. Декораторы, регулярные выражения.
8. Робототехника.

### **Приготовление среды для изучения основ языка Python:**

Для начала комфортного изучения Python языка лучше использовать ОС на основе Linux, так как благодаря архитектуре Linux (Линукс) (Рисунок 2) рабочую среду для программирования здесь настроить легче по сравнению с другими ОС. В последние годы появился Docker и WSL, упрощающие настройку среды для программирования и проверку работы



программ в Windows, но в Linux это сделать легче. Например, если ученик хочет программировать на Python, то установить все необходимые инструменты можно с помощью одной команды и после этого больше ничего настраивать не нужно. В Windows же этот процесс сложнее, так как необходимо найти и установить инструменты компиляции, а затем настроить операционную систему, чтобы их видеть.

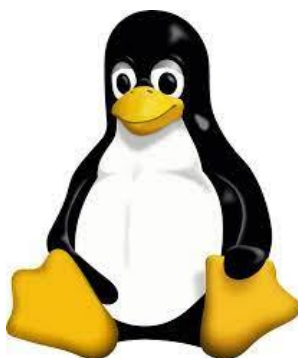


Рисунок №2. Логотип Linux.

Для веб-программирования рационально будет установить и настроить веб-сервер и интерпретатор PHP в Linux с помощью пакетного менеджера дистрибутива. Расположение конфигурационных файлов в файловой системе стандартизировано, и ученик может без труда их найти и настроить всё, что ему нужно. В Windows придется скачивать и настраивать все вручную.

Для использования системы Linux рекомендуем использовать его дистрибутивы, такие как самые простые и популярные: Ubuntu (Рисунок 3), Debian, Mint, Arch. Их рабочие столы и функционалы похожи на Windows [1].

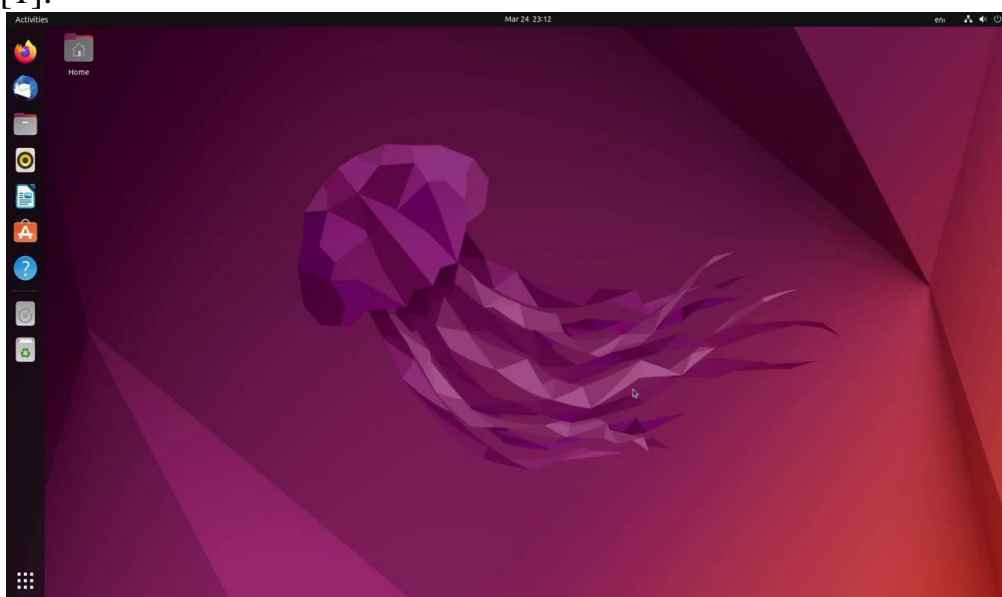


Рисунок №3. Рабочий стол Ubuntu.

Для примера, почему Ubuntu (Убунту)?

Ubuntu (Убунту) — это дистрибутив операционной системы (ОС) Linux, основанный на Debian GNU/Linux. Дистрибутив — это форма распространения программного обеспечения. Под дистрибутивом GNU/Linux понимают общее определение ОС, которые используют ядро Линукс и готовы для конечной установки на клиентское оборудование. В настоящий момент Убунту активно развивается и поддерживается свободным сообществом.

Где используется Ubuntu:

- Ubuntu используется на виртуальных выделенных серверах (VDS/VDS) и на выделенных серверах (Dedicated Server).
- В качестве операционной системы на домашнем компьютере.
- Убунту используется в качестве основной ОС на серверах проекта «Wikipedia».
- В учебных заведениях. Например, еще в 2010 году более ста финских школ перевели компьютерные классы на использование свободного ПО. В качестве дистрибутива была выбрана Убунту.
- В банковской сфере, государственных органах и других крупных областях.

Плюсы Убунту:

1. Безопасность. Убунту — самый безопасный среди дистрибутивов Линукс. Разработчики уделяют пристальное внимание отслеживанию уязвимостей, публикации CVE и быстрому закрытию уязвимых точек.
2. Бесплатность. Ubuntu бесплатен, что привлекает множество пользователей по всему миру.
3. Большое дружелюбное сообщество. Как мы писали ранее, данный дистрибутив является одним из самых популярных во всем мире. За счет этого образовалось огромное сообщество пользователей и разработчиков, готовых помочь коллегам и оперативно решить возникший вопрос.
4. Простота в использовании. Помните, выше мы рассказывали об идеях, которые были заложены при разработке Ubuntu. Одна из этих идей — понятность и простота в работе.
5. Резервное копирование. Ни один другой дистрибутив Линукс не содержит столько качественных инструментов для создания точек восстановления.
6. Удобный центр приложений, содержащий более 20 000 программ. С его помощью можно просто и быстро устанавливать необходимый для работы софт. Все программы имеют подробное описание, помогающее с выбором.

7. Быстрая установка. С каждой новой версией установка Ubuntu все больше упрощается, что привлекает неопытных пользователей.

Минусы Убунту:

1. Если ученики всю жизнь использовали Windows, то первое время будет трудно привыкнуть к новой операционной системе.
2. Отсутствие популярных программ. Пожалуй, главная причина, по которой многие пользователи не могут полностью перейти на Линукс. С помощью специального эмулятора этой проблемы можно избежать [2].

### **Среды разработки на языке Python:**

1) Переходя к самому программированию языком Python, можно отметить Jupyter-ноутбук (Рисунок 4), являющийся средой разработки, где сразу можно видеть результат выполнения кода и его отдельных фрагментов. Отличие от традиционной среды разработки в том, что код можно разбить на куски и выполнять их в произвольном порядке [3].



Рисунок №4. Логотип Jupyter.

В такой среде разработки можно, например, написать функцию и сразу проверить ее работу, без запуска программы целиком. Можно поменять порядок выполнения кода. Можно отдельно загрузить файл в память, отдельно проверить его содержимое, отдельно обработать содержимое.

В Jupyter-ноутбуках есть вывод результата сразу после фрагмента кода. Например, можно в середине кода увидеть построенный график, получить предварительные цифры или любую другую визуализацию.

Плюсы:

- Лучшая платформа для начала работы с наукой о данных.
- Легко делиться файлами и визуализациями.
- Разметка и другие дополнительные функции.

Минусы:

- Нет мощных функций из некоторых IDE.

Jupyter Notebook подходит для начала программирования и его изучения на языке Python. Также, в самой платформе для Jupyter Notebook есть курсы по изучению этого языка, что может помочь вам в обучении детей. Jupyter Notebook можно использовать через Anaconda Navigator (Рисунок 5), в котором ученик может найти не только эту среду разработки, но и другие подходящие инструменты программирования.

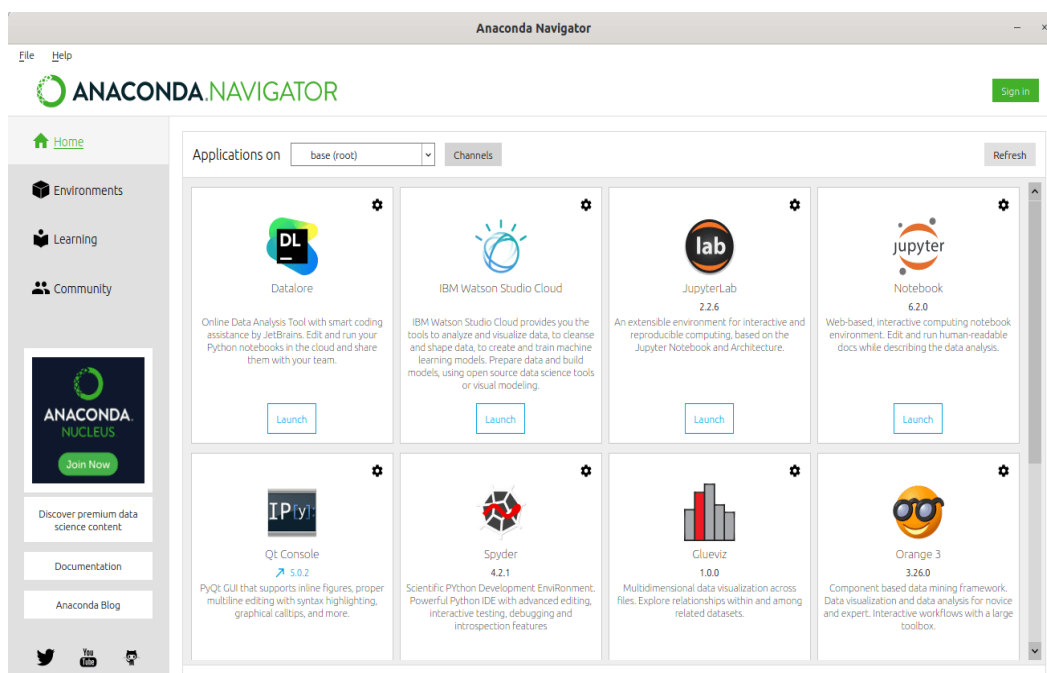


Рисунок №5. Навигация между приложениями от Anaconda.

2) Также для начинающих можно использовать Sublime Text (Рисунок 6), свободное программное обеспечение с широкой поддержкой сообщества. Редактор работает с несколькими языками программирования. В большинстве случаев можно использовать незарегистрированный Sublime Text, но время от времени ученик будет получать всплывающее окно с просьбой зарегистрировать продукт и приобрести лицензию. Sublime Text тонко настраивается и дополняется пакетами для отладки, автозавершения кода, линтинга и т.д.

Плюсы:

- Простой и по большей части бесплатный.
- Тонко настраивается.
- Компактный и эффективный.

Минусы:

- Для удобства требует дополнительных пакетов.

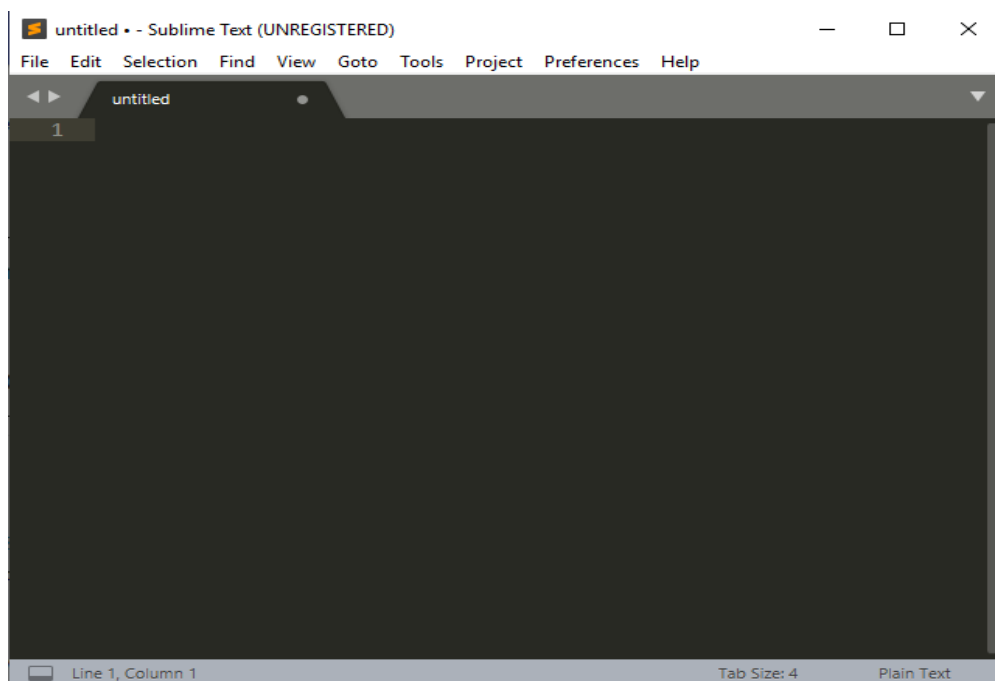


Рисунок №6. Приложения Sublime Text.

3) Для более продвинутого изучения можно пройтись по Visual Studio Code, бесплатный редактор кода от Microsoft для Windows, Linux и MacOS. Его возможности: отладка, подсветка синтаксиса, интеллектуальное завершение кода, predefined фрагменты кода, рефакторинг и интеграция с Git. Поддерживаются различные языки программирования. Для начала работы с Python может понадобиться несколько дополнительных пакетов. Редактор постоянно обновляется.

Плюсы:

- Это платформа с непрерывными обновлениями.
- Потребляет меньше памяти по сравнению с другими инструментами разработки.

- Имеет встроенный терминал.

Минусы:

- Терминал может работать со сбоями.

4) Для профессионального использования языка на уровне программистов высших должностей можно использовать PyCharm (Рисунок 7), это интегрированная среда разработки специально для Python. Разработана чешской компанией JetBrains. Редактор разработан специально для Python, так что имеет широкий набор возможностей, таких

как автозавершение и инспекции кода, подсветка ошибок, исправления, отладка, система контроля версий и рефакторинг. IDE доступна на Microsoft Windows, Linux и MacOS. Есть бесплатная и платная профессиональная версии. Профессиональная IDE имеет несколько дополнительных функций, но бесплатной версии достаточно для большинства задач.

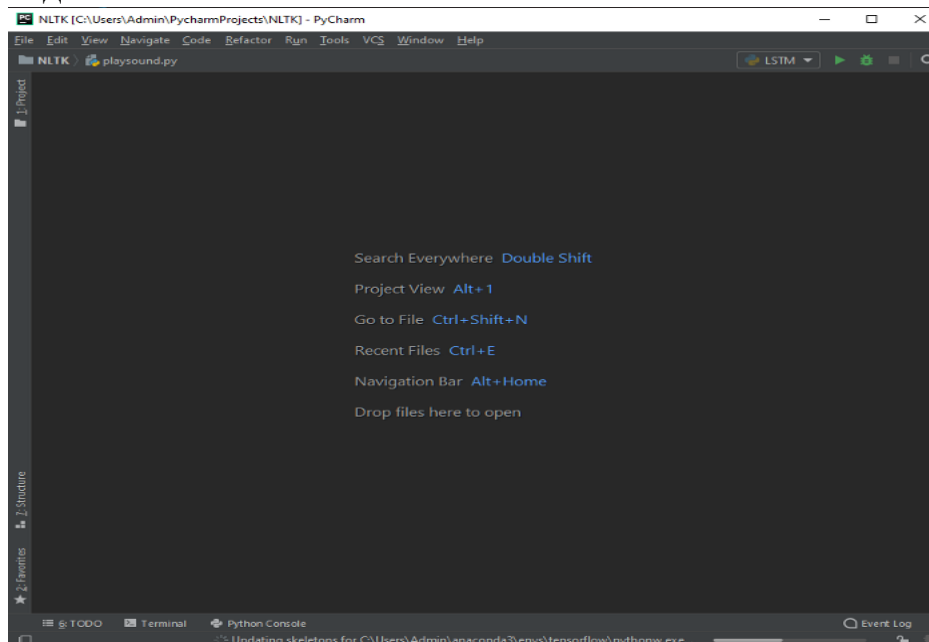


Рисунок №7. Pycharm.

Плюсы IDE:

- Имеет важные встроенные функции.
- Разработана профессионалами специально для Python.
- Поддерживает виртуальные среды Anaconda.

Минусы:

- Основная проблема PyCharm: если у вас недорогой ПК или ноутбук и в нем нет 8 Гб оперативной памяти, то IDE немного притормаживает и работает довольно медленно [3].

### **Среды разработки и библиотеки для программирования роботов на языке Python:**

1) Переходя к программированию роботов EV3 на Python, можно использовать MicroPython (Рисунок 8). Среда EV3 построена на базе процессора ARM9, и разработчики специально оставили архитектуру открытой. Это решение позволило накатывать альтернативные прошивки, одной из которых стал образ для работы с MicroPython. Он позволяет использовать Python для программирования EV3, что делает работу с набором еще ближе к задачам из реальной жизни.

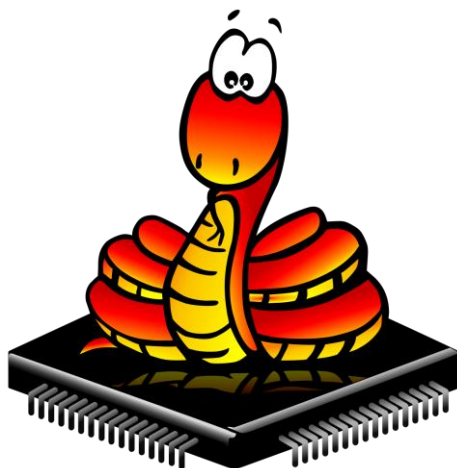


Рисунок №8. Логотип MicroPython.

Чтобы начать работать, нужно скачать образ EV3 MicroPython на microSD-карту, установить ее в микрокомпьютер EV3 и включить его. Затем нужно установить бесплатное расширение для Visual Studio. И можно приступить к работе. MicroPython можно также использовать и для других микроконтроллеров, таких как ESP 32 на среде программирования Thonny IDE. Сам по себе MicroPython универсален и может использоваться в разных связках.

2) Также для программирования роботов на языке Python можно использовать библиотеку PySerial (Рисунок 9). Данная библиотека обеспечивает все функции и методы, которые понадобятся для общения с Arduino [4].

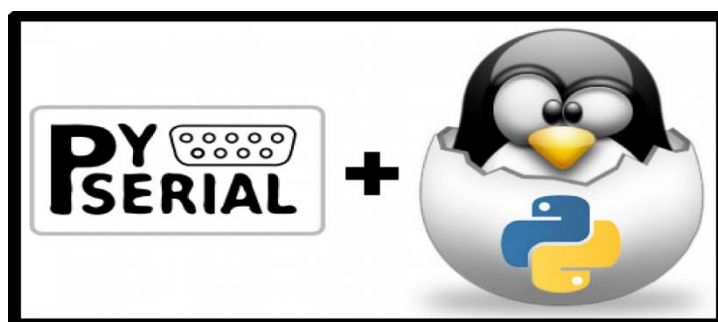



Рисунок № 9. Логотип PySerial.



Примеры задач на MicroPython школы робототехники “Роботек” (Рисунок 10, 11):

### Задача №1

➤ Робот должен проехать круг (с разными радиусами)



### Ответ 1

```
while True:
    m1.run(500)
    m2.run(-500)
    time.sleep(10)
```

Рисунок № 10. Задачи по MicroPython №1.

### Задача №2

Напишите функцию которая выводит на экран блока яркость отраженного цвета , при этом проигрываю любую ноту.

### Ответ 2

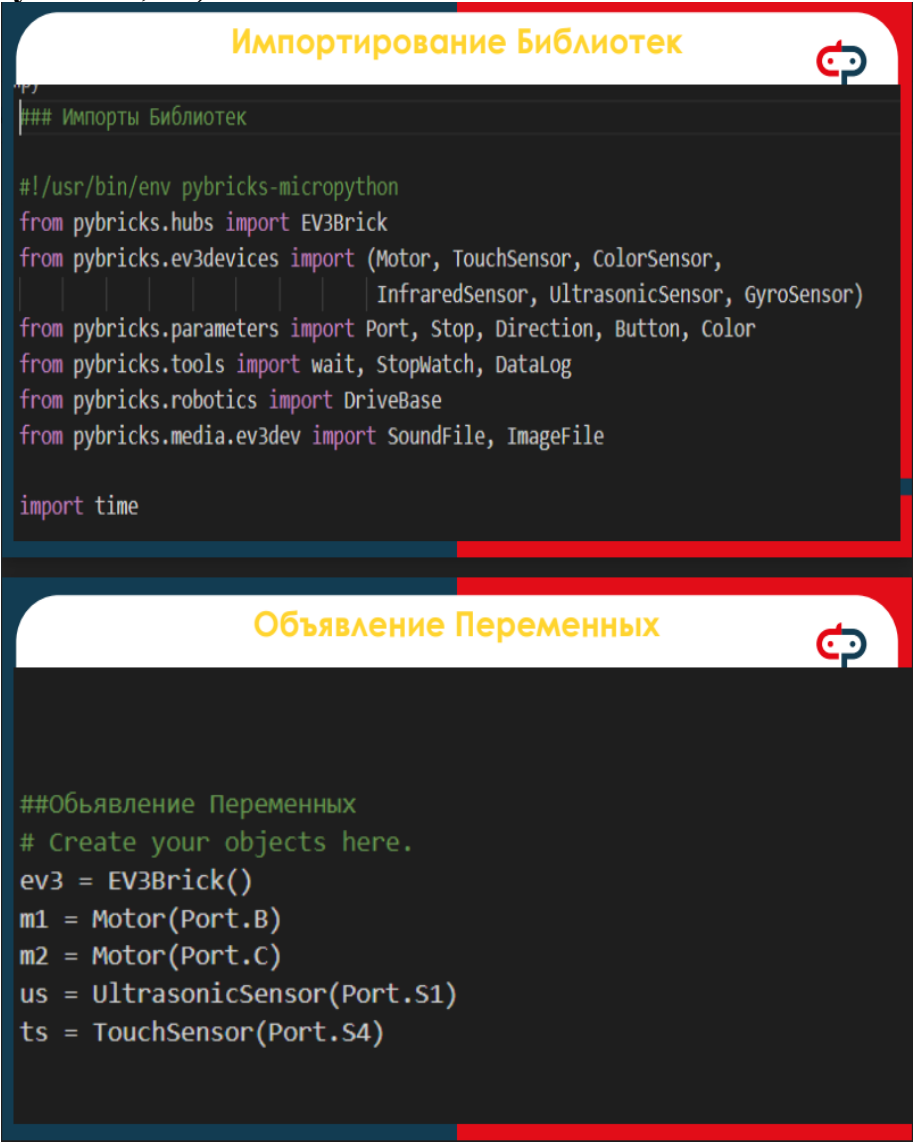
```
ev3 = EV3Brick()
cs = ColorSensor(Port.S1)

def note():
    ev3.screen.draw_text(10,90,int(cs.reflection()))
    time.sleep(2)
    ev3.speaker.play_notes(['C4/4'],100)

while True:
    note()
```

Рисунок № 11. Задачи по MicroPython №2.

**Пример обучения MicroPython в школе робототехники “Роботек” (Рисунок 12, 13):**



The image shows two screenshots of a MicroPython IDE. The top screenshot is titled "Импортирование Библиотек" (Importing Libraries) and contains the following code:

```
### Импорты Библиотек

#!/usr/bin/env pybricks-micropython
from pybricks.hubs import EV3Brick
from pybricks.ev3devices import (Motor, TouchSensor, ColorSensor,
                                  InfraredSensor, UltrasonicSensor, GyroSensor)
from pybricks.parameters import Port, Stop, Direction, Button, Color
from pybricks.tools import wait, Stopwatch, DataLog
from pybricks.robotics import DriveBase
from pybricks.media.ev3dev import SoundFile, ImageFile

import time
```

The bottom screenshot is titled "Объявление Переменных" (Variable Declaration) and contains the following code:

```
##Объявление Переменных
# Create your objects here.
ev3 = EV3Brick()
m1 = Motor(Port.B)
m2 = Motor(Port.C)
us = UltrasonicSensor(Port.S1)
ts = TouchSensor(Port.S4)
```

Рисунок № 12. Обучения MicroPython №1.

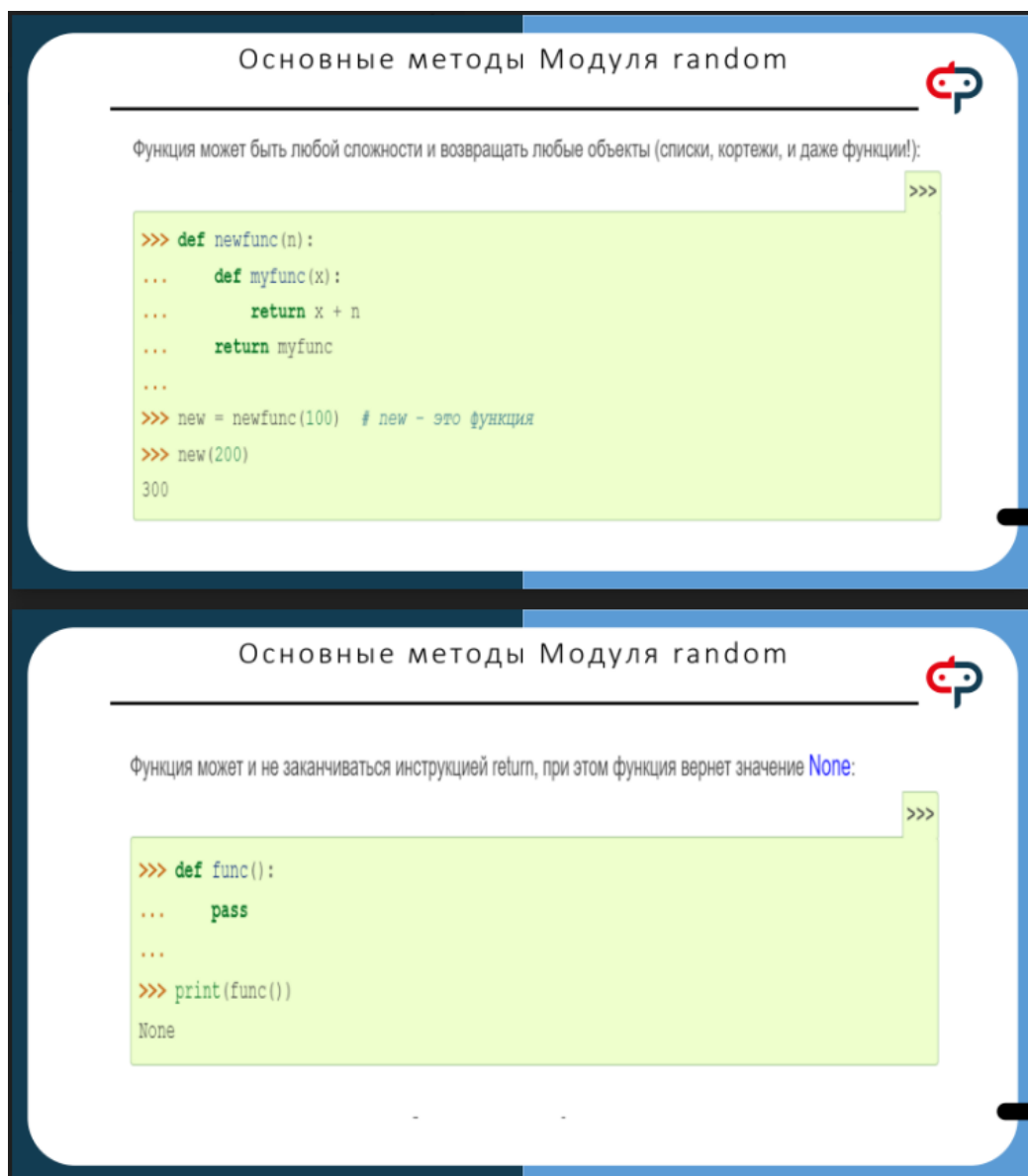


Рисунок № 13. Обучения MicroPython №2.

**Таблица 1. Примеры методики обучения MicroPython в школе робототехники “Роботек”:**

№	Тип урока	Тема	Цель
1	Теория	День 1. Знакомство с EV3 MicroPython	1. Введение в библиотеку EV3Python 2. Язык программирования Python 3. Введение в ИИ 4. Работа на VScode 5. Создание своих собственных решений
2			Повторение материала: 1) color() 2) ambient()

	Теория, повторение	День 2. Знакомство с EV3 MicroPython. Часть 2.	3) reflection() 4) rgb() 5) distance() 6) presence() 7) speed() 8) angle() 9) reset_angle(angle) Узнать ответы на эти вопросы: Что такое комментирование? Какие виды комментариев вы знаете? Для чего оставляют комментарии? Для чего нужен модуль(библиотека) random? Что такое методы в программировании?
3	Теория	День 3. Функции	Изучить функции Python (def, return, continue, pass, break, while, for, int, float, string)
4	Теория	День 4. Массивы	Изучить массивы Python (list, numpy, двумерные, трехмерные, dict)

### **Открытые ресурсы по изучению языка Python для начинающих:**

Изучить программирование на Python можно по бесплатным ресурсам, которые позволят ученикам стать востребованными программистами. Команда школы робототехники “Роботек” рекомендует подборку русскоязычных и англоязычных сайтов для изучения популярного языка программирования.

### **Бесплатные курсы и пособия для программистов на Python:**

1) <https://stepik.org/course/67/promo>

Материал подойдет тем, кто не имеет опыта программирования. Курс содержит 28 уроков, разъясняющих базовые понятия и элементы синтаксиса Python: списки, операторы, циклы, условия, строковые и числовые переменные. Для проверки знаний доступно 27 тестов. В продолжении курса есть еще 19 уроков и 25 тестов с задачами и советами по каждой теме.

2) [https://ru.hexlet.io/courses/python\\_101](https://ru.hexlet.io/courses/python_101) (Рисунок 14)

Теоретическая информация по синтаксису, типам коллекций, функциям, операторам и юникоду с проверкой пройденного материала после каждой лекции. Для каждого урока доступно видео с текстовой базой (основные моменты теории и код).

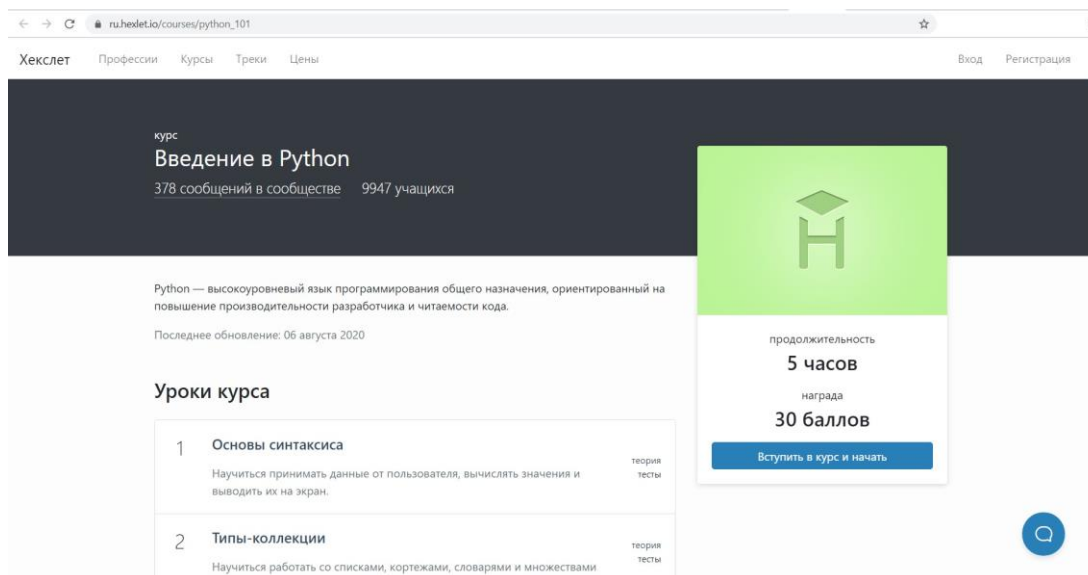


Рисунок № 14. Хекслет.

3) <https://intuit.ru/studies/courses/49/49/info>

В курсе много теоретической информации в виде текста. Рассмотрены библиотеки и форматы данных, а главное – применение Python для создания программ в разных предметных областях.

4) <https://python.swaroopch.com/>

В вольном переводе название электронной книги звучит как «Укус питона» – это одно из наиболее известных учебных пособий по популярному языку программирования. В курсе расскажут про принципы, базовые понятия, основы синтаксиса, инструкции по программированию.

5) <https://www.learnpython.org/> (Рисунок 15)

Информация для новичков и опытных программистов: основы (функции, классы, операции и т.д.), руководство по базам данных, материалы для продвинутых разработчиков (специализации, карты, многопользовательские аргументы, частичные функции и пр.). Здесь можно проверить знания на практике: написать код, запустить программу и получить отчет по ошибкам.

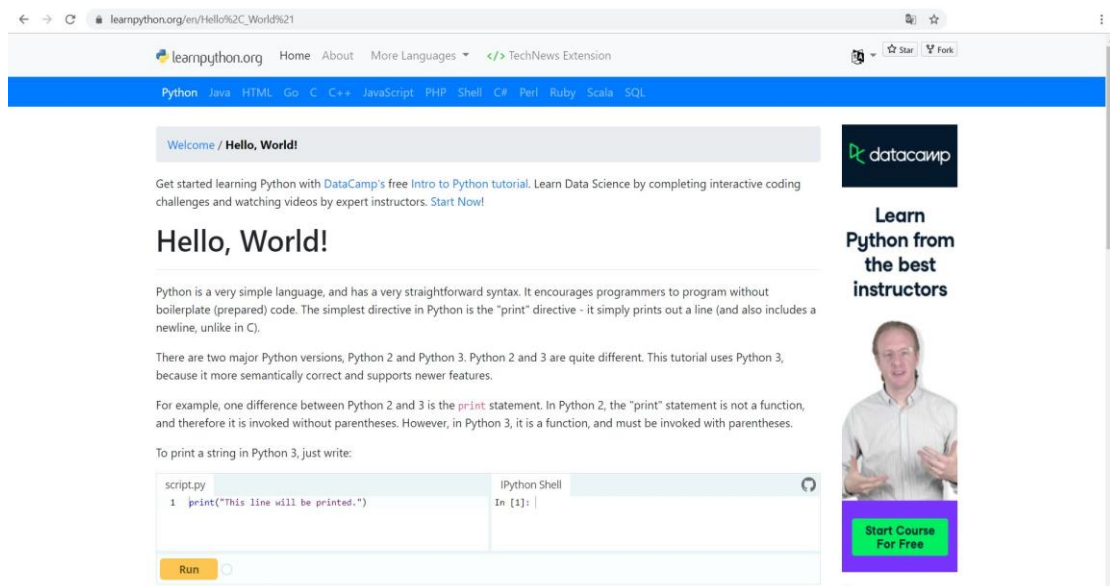


Рисунок № 15. Learnpython.

6) <https://www.codecademy.com/learn/learn-python>

Изучение Python 2 от фундаментальных знаний до написания кода. Интерактивные занятия с возможностью запустить и проверить код через браузер без дополнительных настроек. Преимущество платформы в отсутствии потребности писать код целиком – необходимо только вносить небольшие изменения и запускать программы.

7) <https://www.udemy.com/>

На этом сайте собрано несколько бесплатных курсов с видеоматериалами. Наиболее востребованы следующие:

- Основы Python 3.
- Python Практика для начинающих.
- Learn Python 3 From Scratch.
- Learn Python Language Fundamentals In Simple Way.
- Python for Absolute Beginners Part 1 and Part 2.

8) <https://www.educative.io/courses/learn-python-3-from-scratch>  
(Рисунок 16)

Интеграционная текстовая платформа, которая позволяет изучать концепцию и писать код по каждому уроку. Новичку/ученику не придется ничего скачивать и настраивать на своем устройстве, все обучение происходит на сайте. В начале курса идет объяснение основ и концепций, по которым придется решать задачи для работы с функциональным программированием и структурами данных.

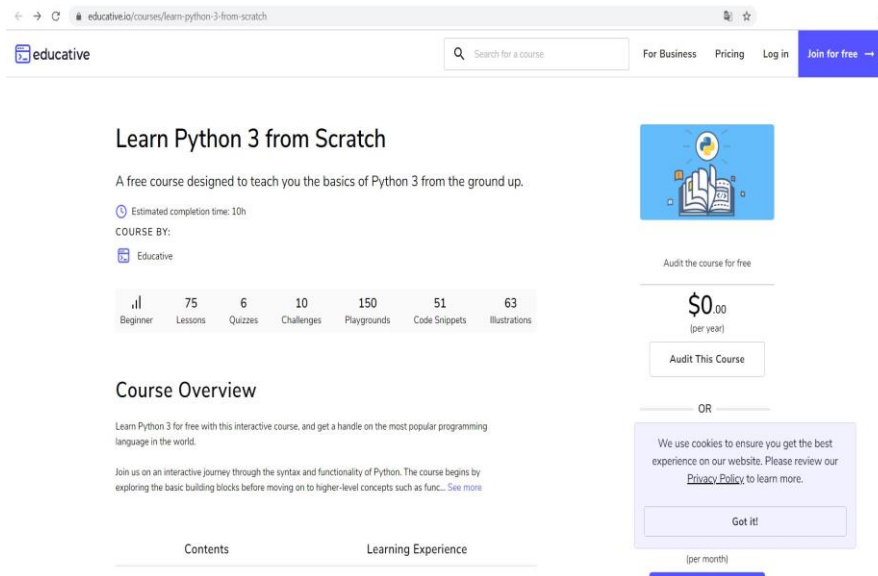


Рисунок № 16. Educative.

9) <https://www.coursera.org/search?query=python&>

Этот популярный ресурс предлагает несколько учебных программ от специалистов ведущих университетов (INSEAD, Stanford, National University of Singapore и прочие). Стоит обратить внимание на курс от преподавателей Мичиганского университета «Специализация Python для всех», который (среди прочего) включает следующие разделы: «Структуры данных в Python», «Capstone: Retrieving, Processing, and Visualizing Data with Python» и «Using Databases with Python».

10) <https://www.codewars.com/?language=python>

Интерактивный ресурс с собранием функциональных задач, интересными тестами, отображением результатов, проведением дискуссий. Игровой процесс мотивирует быстро проходиться по задачам.

### **Бесплатные видео-лекции по программированию на Python:**

1) <https://www.youtube.com/watch?v=rfscVS0vtbw>

Видео англоязычное, но с русскими субтитрами. В длительной (более 4-х часов) лекции собрана информация по установке и настройке Python и PyCharm, показано создание калькулятора и игры. В курсе представлено все – от переменных и списков до модулей и функций.

2) [https://www.youtube.com/watch?v=\\_uQrJ0TkZlc](https://www.youtube.com/watch?v=_uQrJ0TkZlc)

Классы, модули, параметры и другая теория рассмотрена в шестичасовом видео. В конце инструктор Мош разбирает три проекта: по автоматизации, машинному обучению, а также по созданию веб-сайта. Лектор дает материал на хорошем английском, доступны также субтитры на русском языке.



3) <https://www.youtube.com/playlist?list=PLS1QuIWo1RIaJECMeUT4LFwJ-ghgoSH6n>

Плейлист из 220 видео, в котором собраны всевозможные данные для изучающих программирование на Python новичков. Синтаксис, команды, обработка исключений, расширения и все прочее, что нужно для интерактивного обучения.

4) <https://www.youtube.com/playlist?list=PL61E606149255B362>

19 видео от команды Google Developers с информацией о веб-программировании на Python. Опытные разработчики демонстрируют практическое применение материала, представленного на теоретических курсах Google's Python Class [5].

### **Робототехника на основе Arduino**

Arduino — это электронный конструктор и удобная платформа быстрой разработки электронных устройств для новичков и профессионалов. Платформа пользуется огромной популярностью во всем мире благодаря удобству и простоте языка программирования, а также открытой архитектуре и программному коду. Устройство программируется через USB без использования программаторов.

Arduino позволяет компьютеру выйти за рамки виртуального мира в физический и взаимодействовать с ним. Устройства на базе Arduino могут получать информацию об окружающей среде посредством различных датчиков, а также могут управлять различными исполнительными устройствами.

Микроконтроллер на плате программируется при помощи языка Arduino (основан на языке Wiring) и среды разработки Arduino (основана на среде Processing). Проекты устройств, основанные на Arduino, могут работать самостоятельно либо взаимодействовать с программным обеспечением на компьютере (напр.: Flash, Processing, MaxMSP).

### **Среды разработки для работы с Arduino:**

1) ScratchDuino (Рисунок 17)

Для работы с Ардуино и ее среды разработки, мы рассмотрим ScratchDuino, S4A (возрастная категория учащихся 5-6 классов, первый год обучения.)

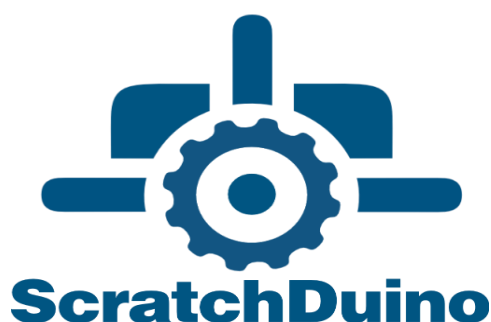


Рисунок № 17. Логотип ScratchDuino.

Таблица 2. Преимущества и недостатки ScratchDuino.

Преимущества	Недостатки
Язык программирования Scratch.	Плохая расширяемость конструктора (подключение экранов, других контроллеров, организация взаимодействия между ними невозможна)
Полностью укомплектованные наборы	Потеря автономности. Робот требует постоянного соединения с компьютером.

Arduino – открытая платформа. Наиболее известными в России решения предоставлены компаниями Амперка и РОББО (ScratchDuino). РОББО предоставляет два решения – Роббо-лаборатория и Роббо-платформа. ScratchDuino. Лаборатория — цифровая лаборатория, устройство, позволяющее собирать данные об окружающей среде с различных датчиков и обрабатывать полученную информацию в программе на компьютере. ScratchDuino/Робоплатформа — это робототехнический конструктор, созданный для обучения как детей, так и взрослых программированию и робототехнике с помощью программирования микроконтроллеров и обработки информации с различных датчиков. Для работы с предложенными решениями используется среда программирования на основе Scratch. Управление осуществляется с компьютера.

Среда позволяет составлять алгоритмы на языке визуального программирования Scratch, что позволяет познакомить обучающихся с исполнителями и наглядно продемонстрировать выполнение им команд.

## 2) Ardublock

Также, мы можем использовать Ardublock (Рисунок 18) (возрастная категория учащихся 6-7 класс, второй год обучения). Компания Амперка

пошла иным путем. Она предоставляет широкий выбор комплектующих деталей или готовых наборов деталей, предлагая программировать микроконтроллеры в среде разработки Arduino IDE. Однако существуют различные языки – интерпретаторы, такие как google Blockly или ArduBlock , которые позволяют переводить программу, написанную на визуальном языке на язык Wiring (небольшую модификацию языка C++)

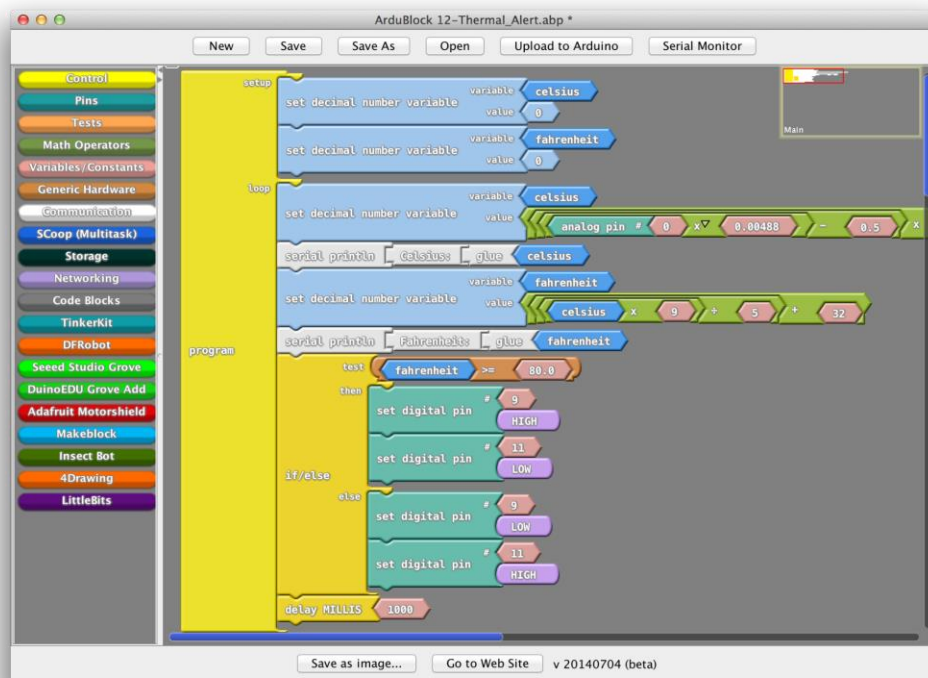


Рисунок № 18. ArduBlock.

Таблица №3. Преимущества и недостатки ArduBlock.

Преимущества	Недостатки
Визуальный язык программирования.	ArduBlock – более не поддерживается разработчиками.
Хорошая расширяемость конструктора. Позволяют полностью использовать возможности, заложенные в микроконтроллер.	При сборке конструктора требуются некоторые начальные знания электротехники.
ПО является код-генератором. Возможен постепенный переход на программирование на языке Wiring.	

Простота предыдущей среды программирования несет в себе серьезные ограничения как на количество используемых в работе

электронных компонентов, так и на возможность создания автономных электронных устройств. Языки-интерпретаторы позволяют в более полной мере реализовать потенциал платформы. На их основе можно создавать полностью автономные проекты, расширять их за счет подключения сторонних устройств, таких как LSD/TFT – экраны, Wifi модули и др.

### 3) Arduino IDE

Далее можно рассмотреть более профессиональную среду программирования. Arduino IDE (Рисунок 19) (8кл., третий год обучения) – позволяет программировать на C-подобном языке.

В конечном итоге обучающиеся должны перейти от изучения визуальных языков программирования к реальным. Начиная работу в среде Arduino IDE, учащиеся знакомятся с C-подобным языком программирования.



Рисунок № 19. Arduino IDE

Таблица №4. Преимущества и недостатки Arduino IDE.

Преимущества	Недостатки
Подходит для обучения программированию на языке C.	При сборке конструктора требуются начальные знания электротехники.
Хорошая расширяемость конструктора.	Имеет относительно высокий порог вхождения. У детей должны быть сформированы первоначальные

	знания из области алгоритмизации
Позволяют полностью использовать возможности, заложенные в микроконтроллер [6].	

### **Симуляторы для работы с Arduino:**

Если под рукой нет набора по Ардуино, можно использовать симуляторы Ардуино. В наши дни **симуляторы Arduino** позволяют любому - и начинающим, и профессиональным разработчикам схем изучать, программировать и тестировать идеи, не беспокоясь о том, чтобы тратить время и деньги.

Симуляторы Arduino являются платформой для программистов и дизайнеров, которые хотят изучить основы схемотехники и проектирования.

С помощью симулятора Arduino ученик получит возможность учиться, не боясь, что ученик повредит свою доску и дизайнерское оборудование.

Студенты, которые сталкиваются с трудностями при покупке электрооборудования, не имея представления о том, как они будут работать, могут устранить ошибки проб и ошибок с помощью симуляторов Arduino, сэкономив при этом время и деньги.

Еще одним преимуществом симуляторов Arduino является тот факт, что они поддерживают отладку от линии к линии, и пользователь точно определяет место, где что-то пошло не так.

Симуляторы Arduino бывают самых разных форм, и они были разработаны таким образом, чтобы быть совместимыми с основными ОС.

#### 1) TinkerCad

Первый в списке - это Тинкеркад (Рисунок 20) (Tinkercad Circuits Arduino) – бесплатный, простой и одновременно мощный эмулятор Arduino, с которого можно начинать обучение электронике и робототехнике. Он предоставляет очень удобную среду для написания своих проектов.

Tinkercad – это онлайн сервис, который сейчас принадлежит САД-системе – компании Autodesk. Тинкеркад уже давно известен многим как простая и бесплатная среда для обучения 3D-моделированию. С ее помощью можно достаточно легко создавать свои модели и отправлять их на 3D-печать. Единственным ограничением для русскоязычного сегмента интернета долгое время являлось отсутствие русскоязычного интерфейса, сейчас эта ситуация исправляется.

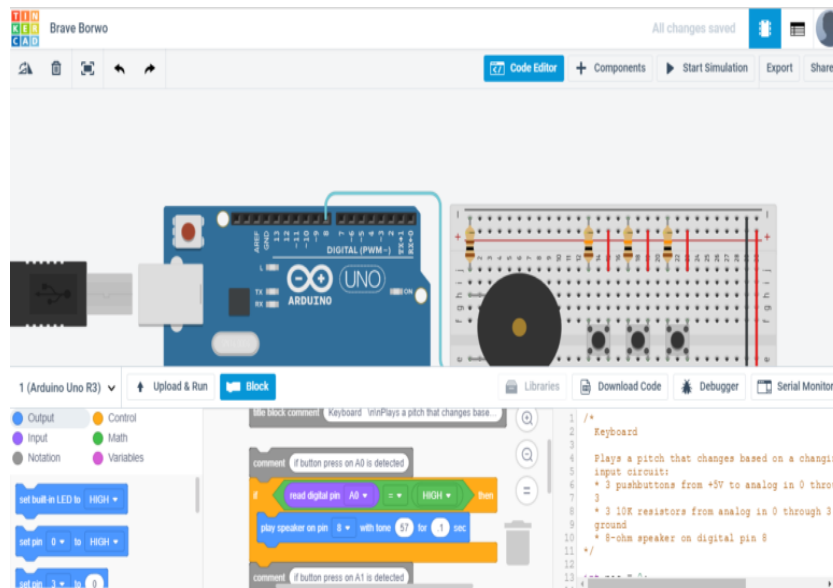


Рисунок № 20. Tinkercad.

Совсем недавно Тинкеркад получил возможность создания электронных схем и подключения их к симулятору виртуальной платы ардуино. Эти крайне важные и мощные инструменты способны существенно облегчить начинающим разработчикам Arduino процессы обучения, проектирования и программирования новых схем.

Список основного функционала и полезных возможностей Tinkercad Circuits:

- Онлайн платформа, для работы не нужно ничего кроме браузера и устойчивого интернета.
- Удобный графический редактор для визуального построения электронных схем.
- Предустановленный набор моделей большинства популярных электронных компонентов, отсортированный по типам компонентов.
- Симулятор электронных схем, с помощью которого можно подключить созданное виртуальное устройство к виртуальному источнику питания и проследить, как оно будет работать.
- Симуляторы датчиков и инструментов внешнего воздействия. Ученик может менять показания датчиков, следя за тем, как на них реагирует система.
- Встроенный редактор Arduino с монитором порта и возможностью пошаговой отладки.
- Готовые для развертывания проекты Arduino со схемами и кодом.
- Визуальный редактор кода Arduino.
- Возможность интеграции с остальной функциональностью Tinkercad и быстрого создания корпуса и других конструктивных

элементов – отрисованная модель может быть сразу же отправлена на 3D-принтер.

- Встроенные учебники и огромное сообщество с коллекцией готовых проектов [7].

**Пример работы на TinkerCad школы робототехники “Роботек” (Рисунок 21):**

7 - сегментный дисплей (Схема подключения)

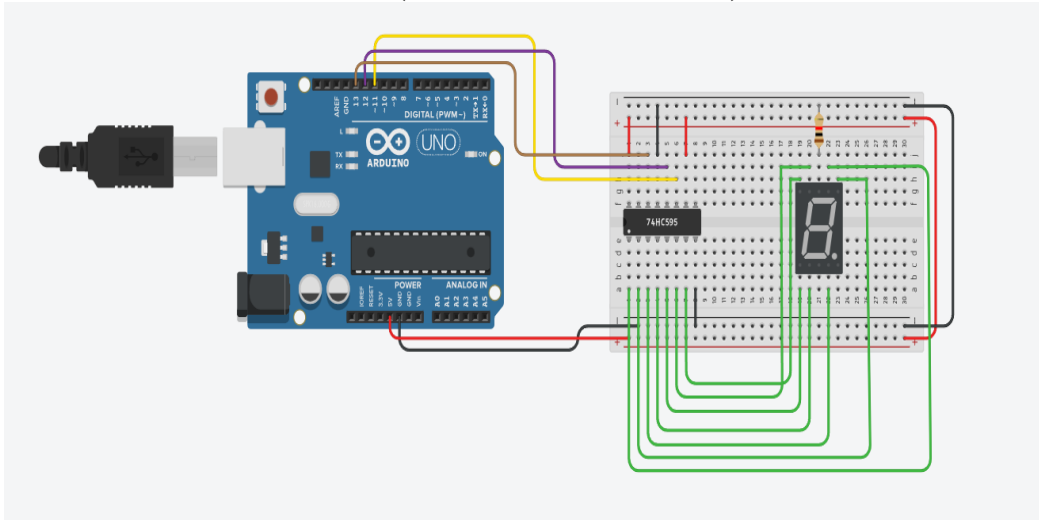


Рисунок № 21. Пример работы на TinkerCad.

2) Autodesk Eagle

Второй в списке **Autodesk Eagle** (Рисунок 22) предоставляет мощные и простые в использовании инструменты для студента.

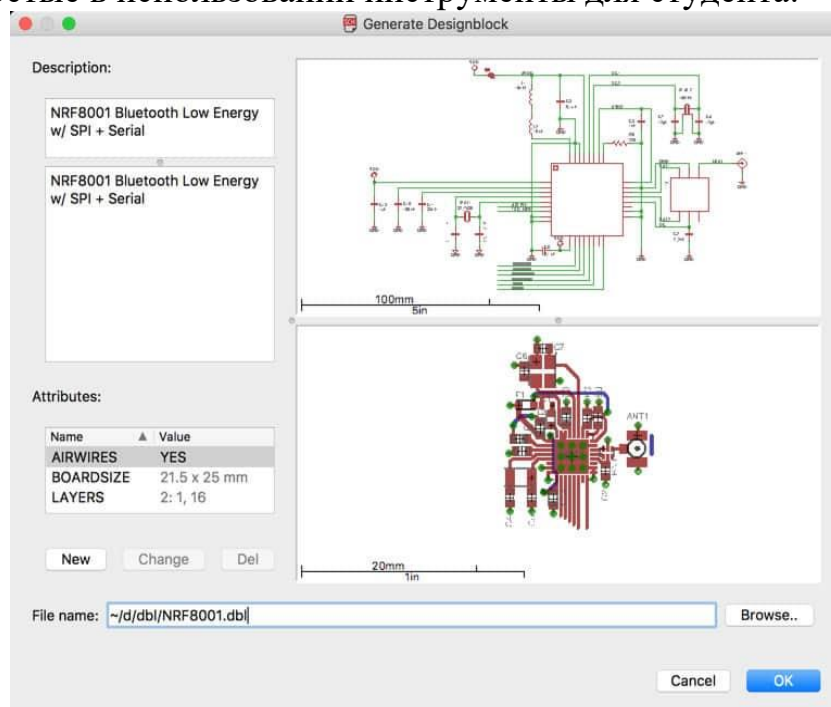


Рисунок № 22. Autodesk Eagle.



Теперь ученики могут воплотить свои электронные изобретения в жизнь с помощью полного набора макетов печатных плат и инструментов для редактирования схем, функций, управляемых сообществом, и содержимого библиотеки.

Ключевые особенности Eagle включают следующее:

- Схематический редактор
- Модульная конструкция блоков — ученик может повторно использовать существующие блоки схемотехники.
- Многостраничные схемы — ученик может сохранить дизайн любого размера организованным.
- Проверка электрических правил — ученик, наконец, может быть уверен в своей схеме.
- Синхронизация дизайна в реальном времени — ученик может синхронизировать схему и схему печатной платы.
- Редактор макетов печатных плат
- Высокоскоростное проектирование — ученик может проектировать с использованием новейших технологий, включая DDR4, PCI Express или USB-C.
- 3D-модели для размещения печатных плат — ученик может без проблем объединить вашу печатную плату и корпус.
- Комплектные компоненты — это универсальный магазин для анализа потребностей ваших компонентов.
- Пользовательские языковые программы (ULP) — ученик может улучшить свой инструмент проектирования.

### 3) Proteus

Третий в списке **Proteus** — отличный симулятор Arduino, который сочетает в себе простоту с различными функциями и позволяет сделать симуляцию Arduino проще.

Этому симулятору удалось проникнуть во все виды промышленности, такие как образование, автомобилестроение и IoT.

Он совместим как с Linux, так и с Windows, и есть много вещей, которые нужно знать о его предложениях и динамичном сообществе.

Наиболее заметной и интересной особенностью Proteus является его способность моделировать взаимодействие между программным обеспечением, работающим на микроконтроллере, и любой цифровой / аналоговой электроникой, подключенной к нему.

Модель микроконтроллера находится на схеме вместе с другими элементами. Как и настоящий чип, он имитирует выполнение объектного кода.

Proteus включает в себя различные виртуальные инструменты, такие как осциллограф, логический анализатор, генератор функций, генератор

шаблонов, таймер счетчика и виртуальный терминал, а также простые вольтметры и амперметры.

Более того, Proteus предоставляет специализированные анализаторы протоколов в режиме Master/Slave/Monitor для SPI и I2C — ученику нужно подключить их к последовательным линиям и контролировать/взаимодействовать с данными в реальном времени в процессе моделирования.

Proteus предлагает способ получить программное обеспечение для связи прямо перед аппаратным прототипированием.

Proteus также позволяет выполнять следующие операции:

- **Отладка:** он может выполнять симуляции в одношаговых режимах и работает как отладчик.
- **Диагностика:** Proteus оснащен комплексной диагностикой или отслеживанием сообщений.
- **Совместное моделирование кода микроконтроллера как низкого, так и высокого уровня** в контексте смешанной схемы SPICE-моделирования.
- Proteus повышает эффективность, гибкость и качество на протяжении всего процесса проектирования.

#### 4) Victronics Arduino

Еще один в списке **симулятор Victronics Arduino** (Рисунок 23) - полнофункциональный из доступных на данный момент.

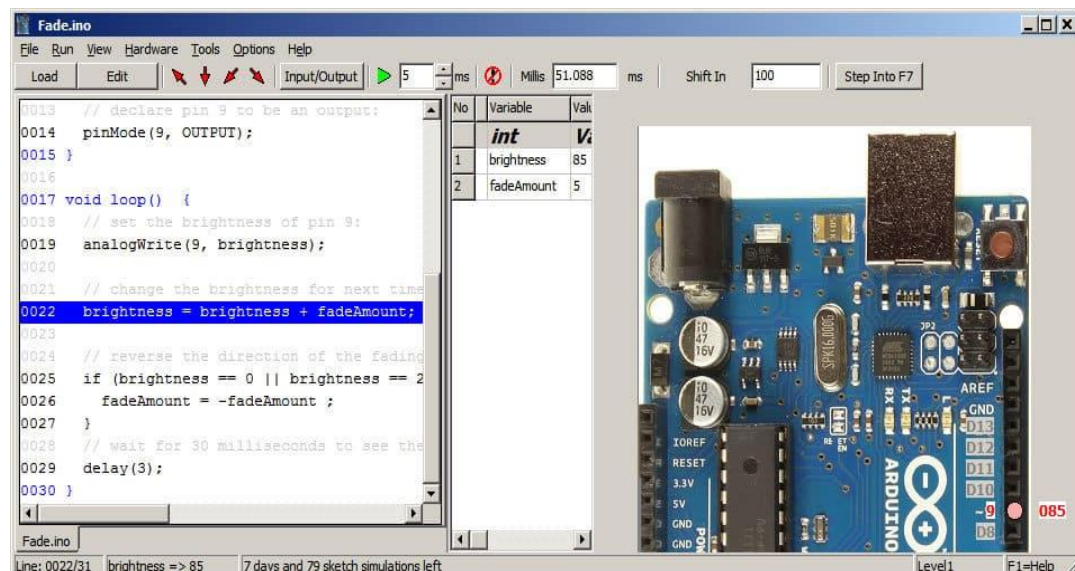


Рисунок № 23. Victronics Arduino.

Вот его самые важные преимущества:

- Victronics Arduino показывает способность преподавать и демонстрировать внутреннюю работу эскиза Arduino.

- Ученик может проверить эскиз без оборудования или перед покупкой оборудования.

- Ученик может отладить эскиз.

- Используя его, ученик может продемонстрировать проект потенциальным клиентам.

- Ученик может разработать сложный эскиз быстрее, чем при использовании аппаратного обеспечения.

Можно скачать бесплатную версию, попробовать ее, и когда ученик будет готов, можно обновить до версии Pro.

Программное обеспечение предназначено для плат Arduino Uno, Mega и других стандартных плат Arduino. Он выполняет множество процессов, включая следующие:

- Он шаг за шагом проходит через программу. Если выбрана новая строка, программа продолжит с этой точки.

- Выполняет цифровую запись, цифровое чтение и PinMode для контактов 0-53.

- Аналоговое считывание для выводов 0-16 и аналоговое считывание для цифровых выводов 0-53.

- Эмулирует последовательный, ЖК-выход, Ethernet, серво, SD-карту, EEPROM, SoftSerial, SPI, Wire.

- Подпрограммы (многоуровневые) с аргументами.

- Просмотр переменных в режиме реального времени.

- Step Into, Step Over, Шаг Out или Режим Run.

- Возможность редактировать эскиз или открывать в Arduino IDE.

- Поддержка 2- и 4-строчного ЖК-дисплея только с импровизированной CGRAM.

- 2-мерные массивы (без инициализации).

У вас также есть возможность изменить шрифт, стиль и размер симулятора.

Режим минимизации идеально подходит для демонстрации/обучения. Программное обеспечение предлагает ограниченную поддержку пользовательских библиотек, указателей и структур.

Ученик может проверить больше информации об этом симуляторе на его официальной странице.

## 5) Virtual Breadboard for Arduino

**Виртуальный макет для Arduino** (Рисунок 24) создан, чтобы помочь пользователям сделать первые шаги в изучении физических вычислений с помощью микроконтроллера Arduino и моделей электронных схем на основе макета.

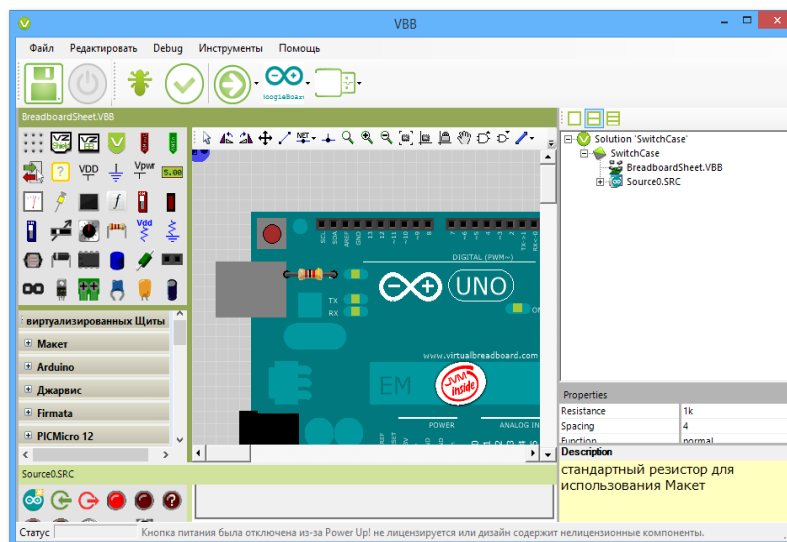


Рисунок № 24. Virtual Breadboard.

Вот основные вещи, которые ученики смогут изучить с помощью Virtual Breadboard for Arduino:

- Ученик может узнать все о физических вычислениях в безопасной среде виртуальной песочницы.
- Ученик может узнать из встроенных в Arduino примеров.
- Ученик может исследовать десятки различных типов датчиков, источников света и двигателей.
- Ученик сможет узнать команды Arduino и что они делают.
- Ученик может использовать макеты в качестве эталона для создания реальных схем.

Вот ключевые особенности Virtual Breadboard:

- Ученик может создавать прикладные схемы виртуальных макетов без пайки.
- Ученик может экспериментировать с электронными схемами на основе микроконтроллеров.
- Ученик сможет протестировать макеты схем перед выполнением сборки.
- Программное обеспечение имеет встроенный редактор встроенных программ для разработки приложений для микроконтроллера.
- Программа предлагает вам интерактивную виртуализацию для изучения, основанного на обучении и схемотехнического тестирования.
- У вас будет достаточно проводника примеров для навигации по многочисленным документированным обучающим примерам.
- Программа имеет интегрированную систему Wiki для разработки документации и курсовых работ.
- Ученик сможет проверить анимированные записи GIF-скриншотов для документации.

- Один клик компилирует и разворачивает встроенные приложения на реальных микроконтроллерах Vbb4UNO.

- Кроссплатформенная разработка с 8-битными микросхемами и Raspberry Pi 2 с целями Windows 10 IoT Core.

VBB4Arduino — это упрощенная версия полного продукта VBB. Он предназначен в качестве самостоятельной версии для начинающих с меньшим количеством опций и модулей, которые могут запутать стартеров.

Лучше всего использовать его как «песочницу» для изучения физических понятий. Опытные пользователи могут предпочесть использовать полную версию VBB с модулем расширения ArduinoToolkit [8].

Также для помощи ученикам в усвоении учебного материала нужно предложить им курсы по изучению Arduino:

### **Бесплатные курсы и пособия для программистов на Arduino на русском:**

1) <https://alexgyver.ru> (Рисунок 25)

Сайт предоставляет бесплатные текстовые уроки — подробные уроки по программированию Arduino с разбором всех тонкостей и особенностей языка.

Есть также бесплатные видеоуроки. Цикл охватывает все стандартные операторы и функции Ардуино и построен таким образом, что от выпуска к выпуску у зрителя идёт плавное формирование “базы”, каждый последующий урок (видео урок) содержит в себе информацию из предыдущих, то есть уроки усложняются и становятся комплексными.

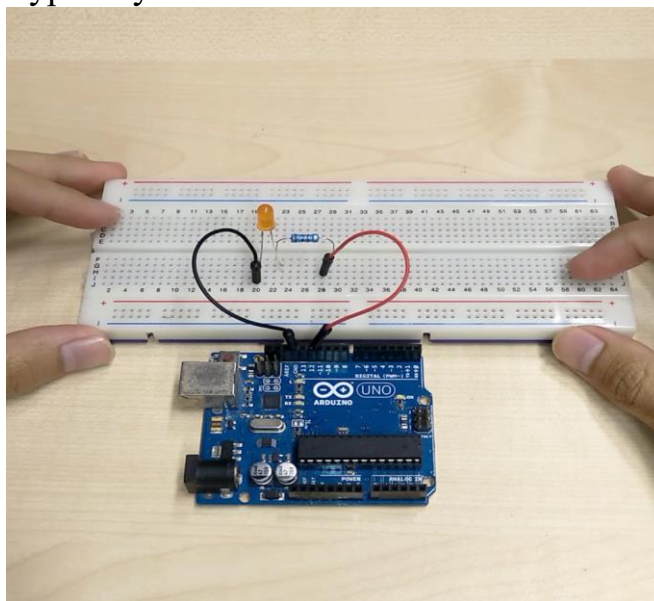


Рисунок № 25. AlexGyver.

Что узнает ученик:

- Что такое Arduino и зачем она нужна?
- Что умеет Arduino и что можно сделать на её основе?
- Подключение датчиков к Arduino.
- Питание Arduino от различных источников электричества.
- Математические операторы для работы с переменными.
- Особенности переменных и констант.
- Общение между компьютером и Arduino через COM порт.
- Какие существуют типы реле? В чём достоинства и недостатки?
- Как подключить реле и как им управлять?

Также, 16 видеоуроков расскажут все нюансы о программировании на Arduino.

2) <https://all-arduino.ru>

Программирование микроконтроллеров Arduino осуществляется на языке программирования C++. Этот язык является низкоуровневым, поэтому считается сложным и имеет высокий порог вхождения. Но для программирования Arduino используется упрощенная версия этого языка программирования.

Этот сайт содержит уроки для обучения новичков. Здесь собраны уроки по Arduino для начинающих. Также будут уроки и по более сложным аспектам программирования Arduino.

Почти все уроки содержат видео для наглядного получения информации, а также текстовую интерпретацию, ссылки на необходимые компоненты и архив с исходниками урока. Так ученик сможет не только посмотреть и послушать урок, но и повторить его для приобретения практических навыков.

После прохождения курса вы сможете реализовать проекты любой сложности. Курс по работе с Arduino для новичков рассчитан на 8 часов.

3) <http://edurobots.ru>

Проект «Занимательная робототехника» представляет учебный курс «Arduino для начинающих». Серия представлена 10 уроками, а также дополнительным материалом. Уроки включают текстовые инструкции, фотографии и обучающие видео.

В каждом уроке ученик найдет список необходимых компонентов, листинг программы и схему подключения. Изучив эти 10 базовых уроков, ученик сможет приступить к более интересным моделям и сборке роботов на основе Arduino.

Курс ориентирован на новичков, и чтобы к нему приступить, не нужны дополнительные сведения из электротехники или робототехники.

4) <https://academiait.ru> (Рисунок 26)

Бесплатный и доступный онлайн-курс «Уроки Ардуино/Arduino». Пройдя данный курс, ученик сделает первый шаг к обучению и сможет четко определиться с направлением интересов.

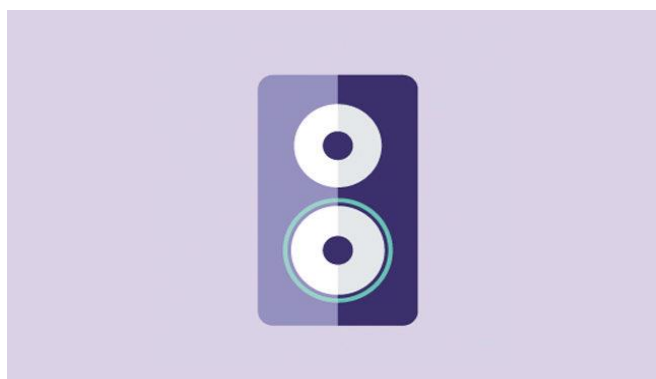


Рисунок № 26. Academia IT

На уроках ученик познакомится с Ардуино:

- для чего используется
- как программировать
- как подключать датчики
- как заставить робота выполнять команды и многое другое.

На платформе Академии можно найти и другие курсы по программированию Ардуино, а также радиотехнике и электронике.

5) <https://arduinoplus.ru>

На сайте представлены текстовые и видеоуроки по Ардуино. Большая часть из них бесплатная.

Автор подробно и понятно рассказывает, что такое Ардуино, как его использовать и как программировать. Из уроков также можно узнать, как подключать и управлять датчиками, как подключить микроконтроллер к компьютеру.

Из следующего курса учащийся узнает, как программировать на Python, как выглядит программное обеспечение Ардуино, как использовать внешние компоненты и макетную плату.

6) <https://arduinomaster.ru>

Бесплатные онлайн уроки научат учеников собирать роботов и управлять ими с помощью Ардуино. Для этого ученику понадобится сама плата и программа для написания кода.

За время прохождения уроков ученик научится:

- подключать светодиоды к плате и управлять ими
- подключать питание
- работать с графическими LCD дисплеями
- собирать схему и управлять ею
- устанавливать различные программы для Ардуино и так далее.

7) <https://роботехника18.рф>



Уроки по Ардуино для начинающих — это развитие творческого и конструкторского мышления, вовлечение в технические кружки. Программа уроков Arduino разбита на модули по 16 занятий и рассчитана на учеников и взрослых с нулевыми знаниями в электротехнике и программировании. По окончании каждого модуля остается робот или «умное» устройство, полностью сделанное своими руками.

8) <http://itrobo.ru>

Курс по Ардуино включает в себя:

Основы работы с Arduino: установка, структура программы, подключение светодиодов, проверка датчиков, создание сигнализации.

Мобильные роботы с Ардуино: motor shield, повороты, движение по линии, управление с телефона, перевороты.

Подключение датчиков и устройств: сервопривод, реле, радиомодуль, клавиатура, ЖК-дисплей, фоторезистор, датчики движения, влажности и света.

Уроки предоставляются бесплатно. Оформлены доступным текстом и иллюстрациями — уроки будут понятны всем.

9) <http://eschool.by>

Это начальный курс по программированию микроконтроллеров Arduino. Рекомендуются к изучению в 6-8 классах. В качестве программной среды будет выступать Arduino IDE с установленным плагином ArduBlock.

Перед прохождением данного курса настоятельно рекомендуется изучить курсы по программированию в Scratch, так как программирование с использованием визуальной оболочки ArduBlock очень схоже с программированием в среде Scratch.

Для самостоятельного изучения курса понадобится набор Arduino Upgraded Learning KIT.

Курс состоит из 13 базовых занятий. Внутри курса есть памятки по каждому из компонентов используемого набора Arduino и рекомендации по работе с ними. В конце курса приведены полезные и интересные проекты для самостоятельной реализации. Раздел самостоятельных проектов постоянно пополняется [9].

**Бесплатные курсы и пособия для программистов на Arduino на английском:**

10) <https://learn.adafruit.com/> (Рисунок 27)

Adafruit — отличный вариант, если ученику нужен визуально привлекательный способ изучения Arduino.

Учебники яркие и веселые, в них используются визуальные и письменные указания для каждого этапа проекта. Существуют учебные пособия, подходящие для различных уровней навыков и типов проектов, а регулярные обновления с новыми руководствами означают, что у

учеников никогда не закончатся новые и уникальные проекты для оттачивания навыков.



Рисунок № 27. Логотип Adafruit.

11) <https://docs.arduino.cc/tutorials/>

Официальный сайт Arduino содержит массу информации доступной как профессионалам, так и студентам и любителям.

Учебники представлены в виде документов, шпаргалок и инструкций, предполагают знакомство с основами и решением трудностей, с которыми могут столкнуться все.

12) <https://techexplorations.com/arduino/>

Содержание Tech Explorations разбито на пять разделов, каждый из которых состоит из нескольких подразделов. Здесь нужно много читать, и многие подразделы отрываются от других статей или веб-страниц.

Существуют основные сводные изображения (например, показанное выше), поэтому, если ученики ищут что-то близкое к большому количеству примеров изображений, это руководство не для них. Тем не менее, существует значительное количество подробного контента для изучения, в котором рассматриваются более мелкие детали, поэтому это определенно хорошее начало, если ученик любит глубоко погружаться в справочную информацию о своих проектах.

13) <https://toptechboy.com/> (Рисунок 28)

Веб-сайт Пола Уортон Technology Tutorials — отличный вариант для всех. Эти видеоролики, созданные бывшим инженером-электриком, ставшим учителем естественных наук, разбивают каждый шаг на простые для понимания, но подробные этапы. Это идеально подходит для тех, кто хочет, чтобы кто-то дал больше указаний.



Рисунок № 28. Technology Tutorials.

14) <https://www.freecodecamp.org/>

Еще один отличный видеокурс, который стоит рассмотреть — это FreeCodeCamp. С учетом того, что первое видео длится впечатляющие 4 часа и 4 минуты, это безусловно выбор для очень увлеченных новичков с большим количеством времени, а не для тех, кто просто хочет окунуться в Arduino. Эти видеоролики действительно хорошо структурированы и отлично подходят для того, чтобы лучше понять все тонкости и тонкости возможностей Arduino!

15) <https://thestempedia.com/tutorial-hub/arduino> (Рисунок 29)

STEMpedia — отличный веб-сайт, ориентированный на пользователей Arduino всех уровней. Их четко обозначенные учебные пособия и огромный выбор контента означают, что есть что-то для каждого уровня пользователя. Более того, веселые и разнообразные учебные пособия означают, что у учеников есть возможность по-настоящему адаптировать то, что ученик хочет изучить, и погрузиться во все, что может предложить Arduino.

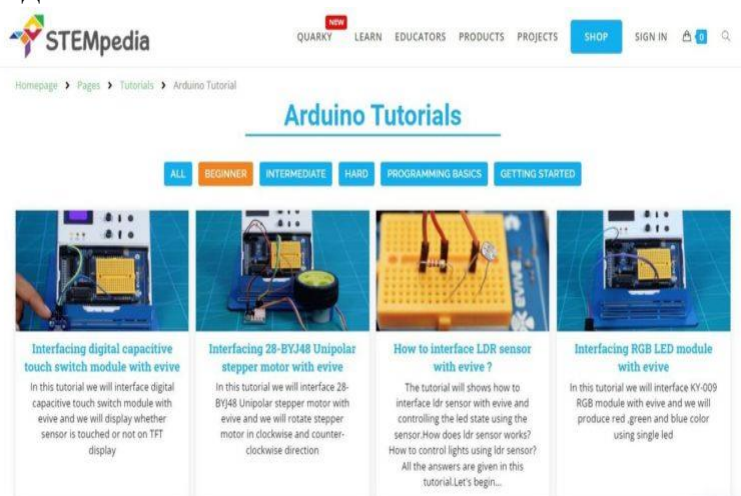


Рисунок № 29. STEMpedia.

16) <https://www.programmingelectronics.com/arduino-crash-course/>  
(Рисунок 30)

Если у ученика мало времени, то Академия программирования электроники — отличный вариант. Их ускоренный курс по Arduino предназначен для того, чтобы ученики как можно быстрее освоили Arduino. 12 видеоуроков познакомят ученика с основами программирования и кодирования Arduino, двумя его наиболее важными функциями.

В дополнение к ускоренному курсу Programming Electronics Academy предлагает целый ряд письменных и видеоуроков по различным способам использования Arduino, а также библиотеку проектов, представленных участниками [10].

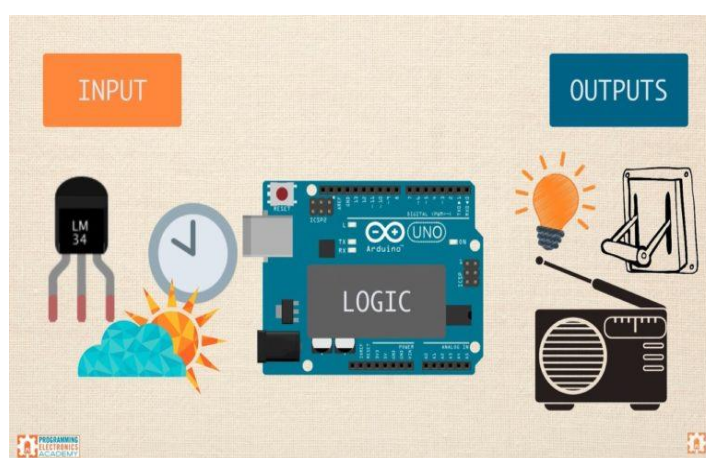


Рисунок № 30. Часть урока от Академия программирования электроники.

**Таблица 5. Примеры методики обучения Ардуино школы робототехники “Роботек”:**

№	Тип урока	Тема	Цель	Сборка	Программирование
1	Теория	День 1. Arduino. Что такое Ардуино?	Изучить что такое Arduino и зачем она нужна, что умеет Arduino и что можно сделать на ее основе. Изучить подключение датчиков к Arduino и изучить питание Arduino		1. Понять интерфейс Arduino IDE. 2. Виды комментариев (//, /*, */) 3. Стандартные функции (#include, void setup() {}, void loop() {})

			<p>от различных источников электричества. Также изучить как запускать скаченные скетчи с интернета, устанавливать библиотеки и прошивать Arduino. Понять начало программирование в Arduino. TinkerCard.</p>	<p>4. Типы данных Ардуино (int, float, string, boolean, char, byte, unsigned int, long, double)  5. Особенности разных типов данных и как их использовать.  6. Математические операторы (pow, sqrt, sqrt и т.д).  7. Укорачивание разных операции.  8. Константы (#define)</p>
2	Теория	<p>День 2 . Кнопка, резисторы и основы программирование</p>	<p>Изучить что такое кнопка и резисторы, motherboard (немного физики: что такое вольт, амперы, земля), как их подключать. Дальше познавать основы программирование и выполнить пару задач с кнопкой.</p>	<p>1. Serial порты (Serial.begin(), Serial print(), Serial println()) и вывод чисел и надписей в COM порту.  2. Системы исчисления  3. Буфер обмена (Serial.available(), Serial.read(), ASCII таблица).  4. Цифровые и аналоговые порты.  5. Настройка порта (pinMode (pin, mode), INPUT, OUTPUT, INPUT_PULLUP).  6. digitalWrite (pin, signal) (LOW, HIGH,</p>

					0,1). 7. Вывод на экран сигналы портов используя кнопку.
3	Теория	День 3. Основы программирования, расширенные возможности кнопки.	Изучить условные операторы, операторы вывода, функции времени, флажки Arduino. Изучить расширенные возможности кнопки и выполнить пару задач связанные с операторами и кнопкой.		1. Условные операторы (if() {}, else {}, else if () {}) 2. Операторы сравнения (a==b, a!=b, <,>,>=,<=) 3. Логические операторы (&&,   , !) 4. Операторы выбора (switch, case, break, default) 5. Временные функции (delay(), delayMicroseconds (), millis(), micros()). 6 Система флажков и как работать с кнопкой. 7. Отладка программы и дребезжания
4	Теория	День 4. Светодиоды, пьезодинамик, аналоговые сигналы	Изучить работу светодиода и пьезодинамик, понять основы физики светодиода и пьезодинамика, подключить пьезодинамик и светодиод с резистором. Выполнить задачи с светодиодами,		1. Подключение светодиода и пищалки 2. Аналоговые сигналы(analogRead (pin), analogWrite(pin), map, constrain)

			пъезодинамик и кнопкой. . Изучить Шим сигнал.		
5	Повторение и теория	День 5. Повторить, Servo привод, фотодатчик	Повторить всё что прошли до этого. Изучить Servo мотор, фотодатчик и подключить эти датчики. Сделать задачи с использованием новых и уже изученными датчиками.		<ol style="list-style-type: none"> <li>1. Подключить библиотеку Servo привода</li> <li>2. Команды Servo привода (Servo, Servo.attach, Servo.write, map)</li> <li>3. Разобрать программы фотодатчика</li> </ol>
6	Тест и теория	День 6. Тест. Датчик расстояния, основы программирования.	Пройти тест. Изучить датчик расстояния (немного физики датчика) и изучить циклы for, while и выполнить задачи с циклами и с датчиком расстояния.		<ol style="list-style-type: none"> <li>1. Циклы (for (counter; condition, change) {}, break, continue, while (condition) {}, do {} while {})</li> <li>2. Скачать библиотеку Датчика расстояния.</li> <li>3. Команды датчика (Ultrasonic, ultrasonic.distance Read)</li> </ol>
7	Мини-проект и Теория	День 7. Ультразвуковой радар, функции.	Изучить функции Arduino и построить ультразвуковой радар на основе Servo привода и датчика расстояния. Выполнить задачи по функциям Arduino	<a href="https://iarduino.kz/blog/povorotnyj-radar-na-osnove-hcsr04-i-arduino.html">https://iarduino.kz/blog/povorotnyj-radar-na-osnove-hcsr04-i-arduino.html</a>	<ol style="list-style-type: none"> <li>1. Функции Ардуино (void, int (return), byte, fload, и т.д )</li> </ol>
8	Теория	День 8. Массивы и их виды.	Изучить массивы и их виды.		<ol style="list-style-type: none"> <li>1. Объявление массива</li> </ol>



			Выполнить задачи с массивами и светодиодами и другими датчиками.		(<тип><имя>[<число элементов>]). 2. Примеры массивов и строение массивов. 3. Виды массивов (двухмерные и т.д, массивы с битами).
9	Теория	День 9. Сдвиговый регистр и сегментный дисплей	9. Изучить сдвиговый регистр и 7 сегментный дисплей. Узнать как они работают и подключить их с друг другом, выполнив пару задач с использование массивов.		1. Команды и библиотеки для работы регистра.
10	Теория	День 10. Дальше изучать сдвиговый регистр и сегментный дисплей	7. Дальше продолжить изучения сдвигового регистра и 7 сегментный дисплея. Начать изучать методы паяния.		1. Команды и библиотеки для работы регистра.
11		День 11. Паяние, жидкокристаллический дисплей и потенциометр	Изучить паяние, жидкокристаллический дисплей и потенциометр. Спаять дисплей и вывести на нем значения потенциометр, датчика расстояния. (можно сделать мини-проект с		1. Библиотека для дисплея 2. Команды для дисплея (lcd.begin, lcd.backlight, lcd.print и т.д)

			дисплеем, сервоприводом, датчиком расстояния и потенциометром)		
12	Теория	День 12. ИК-порт	Изучить ИК-порт, сделать задачи по ИК-порту с дополнением жидкокристаллического дисплея.		1. Установка библиотеки IRremote.h. 2. Изучить команды ИК-порта (irrecv.decode, irrecv.enableIRIn, irrecv.results)
13	Повторение, тест и подготовка к проекту	День 13. Повторение изученного материала, тест и подготовка	Повторить пройденный материал с задачками по разным датчикам и пройти тест. Подготовиться к проекту (написать план, выбрать проект (если дети сами захотят) и изучить компоненты для проекта).		-
14	Проект	День 14. Проектное занятие	Подготовиться к проекту, выбрать проект самому или же сделать проект "Автоматическая мусорка". Подготовить корпус и начать писать код.	<a href="https://alexgyver.ru/arduino-trashcan-2/">https://alexgyver.ru/arduino-trashcan-2/</a>	<a href="https://github.com/AlexGyver/automaticTrashcan_v2">https://github.com/AlexGyver/automaticTrashcan_v2</a>
15	Проект	День 15. Проектное занятие	Написать проект	Проект	-
16	Проект	День 16.	Сделать	Проект	

	Проектное занятие	презентация в Google-презентации и защитить проекты	
--	-------------------	---	--

**Пример задач школы робототехники “Роботек” с использованием TinkerCad (Рисунок 31, 32):**

1) Давайте попробуем сделать простой радар, используя Arduino, ультразвуковой датчик HC-SR04 (HC-SR05) и сервопривод (Радар должен сканировать свое окружение и выводить результаты в COM порт).

2)

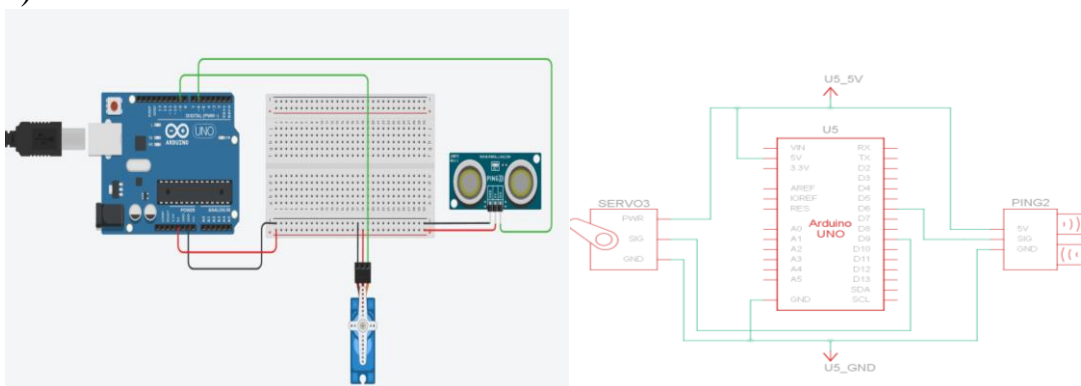


Рисунок № 31. Задача для Arduino №1.

3) При нажатии кнопки поворачиваем привод на 180 градусов и включаем светодиод. При отпускании кнопки ставим привод на 0 градусов и включаем светодиод.

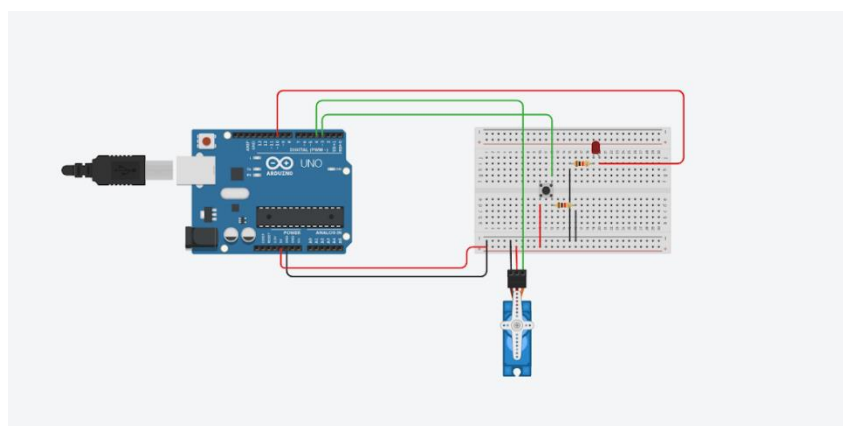


Рисунок № 32. Задача для Arduino №2.

34): Пример ответа школы робототехники “Роботек” (Рисунок 33,

```
1 #include "Servo.h" //Подключение библиотеки управления сервоприводом
2 int trigPin = 6;
3 int echoPin = 5;
4 long duration;
5 int distance; //переменная для вычисления расстояния
6 Servo servo; //создаем объект типа Servo
7
8 void setup()
9 {
10 //устанавливает режим работы портов на ардуино
11 // - выход
12 pinMode(trigPin, OUTPUT);
13 // - вход
14 pinMode(echoPin, INPUT);
15 //определяем скорость передачи по последовательному порту
16 Serial.begin(9600);
17 // Указываем порт для управления сервоприводом
18 servo.attach(9);
19 }
20
21 void loop()
22 {
23 //Функция поворота сервопривода налево
24 left();
25 //Функция поворота сервопривода направо
26 right();
27 }
28
29 void left()
30 {
31 for (int pos = 0; pos <= 180; pos += 1)
32 {
33 servo.write(pos);
34 Serial.print(pos);
35 Serial.print(" ");
36 distance = echoloop();
37 Serial.println(distance);
38 delay(50);
39 }
40 }
41
42 void right()
43 {
44 for (int pos = 180; pos >= 0; pos -= 1)
45 {
46 servo.write(pos);
47 Serial.print(pos);
48 Serial.print(" ");
49 distance = echoloop();
50 Serial.println(distance);
51 delay(50);
52 }
53 }
54 // функция для определения расстояние до объекта
55 int echoloop()
56 {
57 long duration, cm;
58 // для большей точности установим значение LOW на пине Trig
59 digitalWrite(trigPin, LOW);
60 delayMicroseconds(2);
61 // Теперь установим высокий уровень на пине Trig
62 digitalWrite(trigPin, HIGH);
63 // Ждем 10 микросекунд
64 delayMicroseconds(10);
65 digitalWrite(trigPin, LOW);
66 // Узнаем длительность высокого сигнала на пине Echo
67 duration = pulseIn(echoPin, HIGH);
68 //вычисляем расстояние до объекта
69 cm = duration / 58;
70 return cm;
71 }
```

Рисунок № 33. Ответ для задачи по Arduino №1.

```

1 #include <Servo.h> // подключаем библиотеку для работы с сервоприводом
2 Servo servo; // объявляем переменную servo типа "servo"
3 int led_pin=3; // пин подключения
4 int button_pin = 4; // пин кнопки
5 // переменные
6 int buttonState = 0; // переменная для хранения состояния кнопки
7 void setup() {
8   pinMode(led_pin, OUTPUT); // Инициализируем цифровой вход/выход в режиме выхода.
9   pinMode(button_pin, INPUT); // Инициализируем цифровой вход/выход в режиме входа.
10  servo.attach(5); // привязываем сервопривод к аналоговому выходу 10
11 }
12 void loop() {
13   buttonState = digitalRead(button_pin); // считываем значения с входа кнопки
14   if (buttonState == HIGH) {
15     digitalWrite(led_pin, HIGH); // зажигаем светодиод
16     servo.write(0); // ставим вал на 180
17     delay (1000); // задержка в 1 секунду
18   }
19   else {
20     digitalWrite(led_pin, LOW); // выключаем светодиод
21     servo.write(180); // ставим вал на 0
22     delay (1000); // задержка в 1 секунду
23   }
24 }

```

Рисунок № 34. Ответ для задачи по Arduino №2.

### Интернет вещей (IoT) на основе Arduino:

Платы Arduino обладают простотой программирования и легкостью интегрирования в различные встраиваемые системы платы Arduino.

Ранние версии плат Arduino были в основном микроконтроллерами общего назначения, которые подключались к Интернету с помощью модулей GSM и Wi-Fi, но по мере того, как концепция интернета вещей стала получать все большее развитие, стали разрабатываться платы со специальными функциями, поддерживающими интернет вещей (IoT). К ним можно отнести такие платы как Arduino 101 (Рисунок 35) (разработанная совместно с Intel), MKR1000, Arduino WiFi Rev 2 и MKR Vidor 4000, которая является первой платой Arduino на основе чипа FPGA.

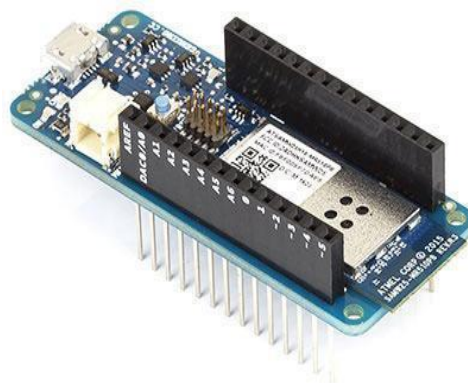


Рисунок № 35. Arduino 101.

Каждая из этих плат была создана с учетом технологий интернета вещей, и все они имеют различающиеся функции, которые делают их более подходящими для конкретных решений, то есть каждая из этих плат подходит для использования в своей конкретной нише. Например, плата Arduino WiFi Rev 2 (Рисунок 36) поставляется с IMU (инерциальный

измерительный блок), что делает ее удобной для создания приложений на базе дронов.



Рисунок № 36. Arduino WiFi Rev 2.

Как и компании Adafruit и Particle.io, Arduino также имеет облачный сервис, предназначенный для использования некоторыми платами Arduino, включая MKR1000, Arduino Yun/Yun Shield и Arduino 101/WiFi Shield 101. Облако устройств Arduino ([cloud.arduino.cc](http://cloud.arduino.cc)) предлагает производителям простой инструмент для подключения своих устройств к сети Интернет и требует минимально короткого процесса настройки для доступа к облаку Arduino.

### **Инструменты для работы с IoT:**

#### 1) Eclipse IoT (Рисунок 37)

Один из наиболее полезных инструментов разработки для IoT. Он является продуктом совместных усилий нескольких организаций из разных отраслей. С его помощью можно создавать фреймворки, получать реализации с открытым исходным кодом, предоставлять сервисы, помогающие в использовании IoT-решений, а также создавать инструменты для самих разработчиков в этой области.



Рисунок № 37. Eclipse IoT.

#### 2) Node - Red (Рисунок 38)

**Node-Red** – это гибкий и полезный визуальный редактор с открытым кодом, который позволяет разработчикам любого уровня налаживать взаимосвязь между облачными системами, базами данных, API.

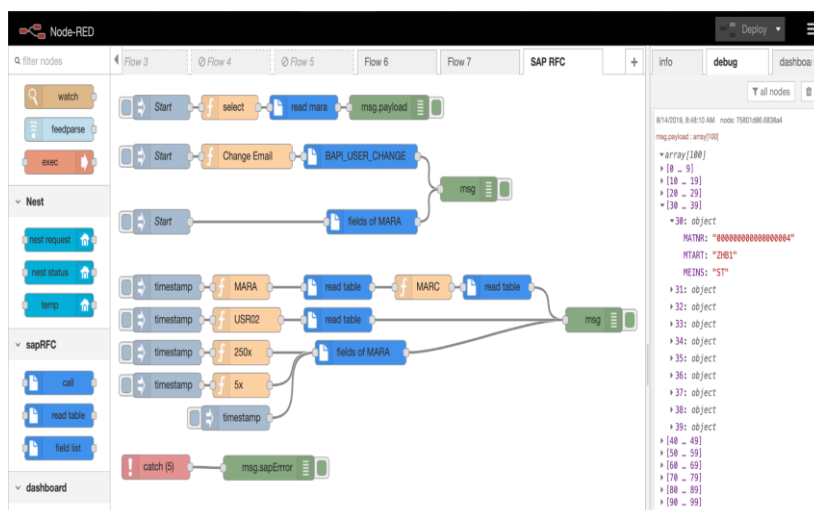


Рисунок № 38. Node - Red.

Работать он может в том числе и на низкобюджетных машинах, включая недорогие облачные решения и Raspberry Pi. Состоит данный инструмент из 225,000 модулей, что облегчает расширение палитры узлов для добавления новых возможностей. Node-Red разработан IBM, поэтому с помощью данного редактора можно создавать Java-функции, которые в итоге можно сохранять для последующего использования, также как шаблоны и потоки.

### 3) Particle (Рисунок 39)

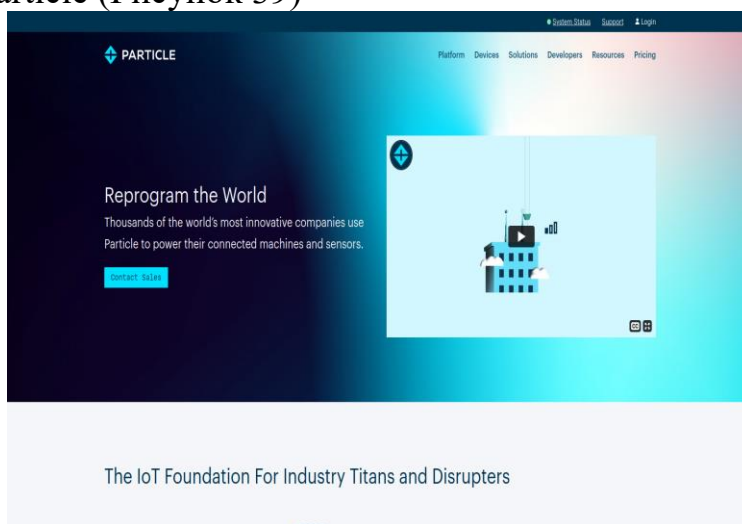


Рисунок № 39. Particle.

Данная платформа характеризуется надежностью и защищенностью, позволяя программистам разрабатывать, подключать, объединять и администрировать IoT-решения быстро и без особых усилий. Многие инженеры из разных стран используют этот сервис для организации и управления разработкой IoT-продуктов.



#### 4) Каа (Рисунок 40)

Каа относится к числу наиболее гибких и расширяемых платформ, позволяющих как создавать IoT-решения, так и управлять подключенными устройствами. Она предлагает богатый набор возможностей, включая сбор и обработку данных, аналитику, управление конфигурацией, визуализацию данных и многое другое. Каа поддерживает открытые протоколы IoT и обеспечивает полноценную безопасность пользовательской информации. Она дает ученику полный контроль над устройствами, при этом ученик может выбирать удобный для него вариант хостинга, будь то публичное облако или частные локальные серверы.

С ее помощью можно подключать и управлять устройствами, используя графический UPI или REST API. Кроме того, она проста в использовании и имеет высокую совместимость.



Рисунок № 40. Каа.

Создатели Каа, имея богатый опыт в этой области, вложили в нее возможность обслуживания всех известных сфер жизнедеятельности, включая различные корпорации и более мелкие стартапы.

#### 5) ThingsBoard (Рисунок 41)

Эта IoT-платформа поможет ученикам организовать для своих проектов сбор и обработку данных, управление устройствами и визуализацию. При ее использовании вывод продукта на рынок занимает минимум времени, а возможные технические сложности решаются очень эффективно. Она позволяет отслеживать миллионы устройств, осуществляя взаимосвязь посредством таких протоколов как HTTP, MQTT и ряда других.

Используя **ThingsBoard**, ученик может мониторить стек технологий и свободно выбирать провайдеров облачных услуг. Этот сервис позволяет моделировать файлы проектов, настраивать устройства, а также получать на основе их данных нужную информацию, делая эти процессы более эффективными.



Рисунок № 41. ThingsBoard.

## 6) Blynk IoT Platform (Рисунок 42)

IoT-платформа отличается рядом возможностей, таких как отображение, хранение и визуализация данных. С помощью библиотеки **Blynk** можно подключать более 400 моделей оборудования, организуя соединение через Wi-Fi, Ethernet, 2G, 3G, 4G, LTE и т.д.

Вся платформа подразделяется на три главных компонента:

- Приложение Blynk предоставляет виджеты, с помощью которых ученик может создавать интерфейсы для своих проектов.
- Сервер Blynk позволяет управлять тысячами устройств, а также налаживать связь между оборудованием и смартфонами.
- Библиотеки Blynk обеспечивают взаимодействие с сервером и обработку команд.

Это открытая платформа, способная работать в любой среде. Для нее характерна скорость, надежность, а также возможность обработки миллиардов информационных потоков и запросов, получаемых с устройств.

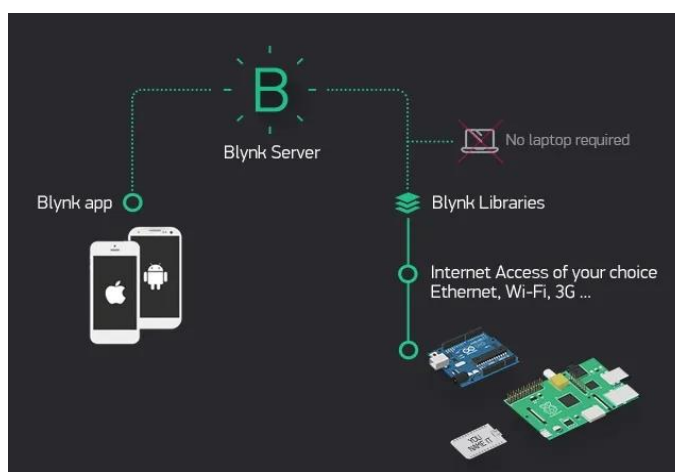


Рисунок № 42. Blynk IoT Platform.

## 7) Device Hive

**Device Hive** тоже является открытым инструментом, помогающим подключать к приложению устройства и добавлять в него объекты. Подключение устройств осуществляется через WebSocket, REST API или MQTT. Платформа поддерживает несколько языков программирования, что делает ее универсальной для всех устройств.

Помимо прочего, она предоставляет участникам возможность обсуждения разработки или бизнеса, а также предлагает подходы к созданию ПО в коммерческих целях. Подключенные устройства можно контролировать без использования сети конфигурации. Ученик также может проводить аналитику удобным для него способом [11].

## Бесплатные курсы и пособия для программистов IoT:

### 1) <https://stepik.org/course/92731> (Рисунок 43)

Интернет вещей. Введение в Arduino (Stepik, 24 урока): ученик может попробовать себя в C++, Arduino (функции цифрового ввода/вывода, ШИМ, датчики).

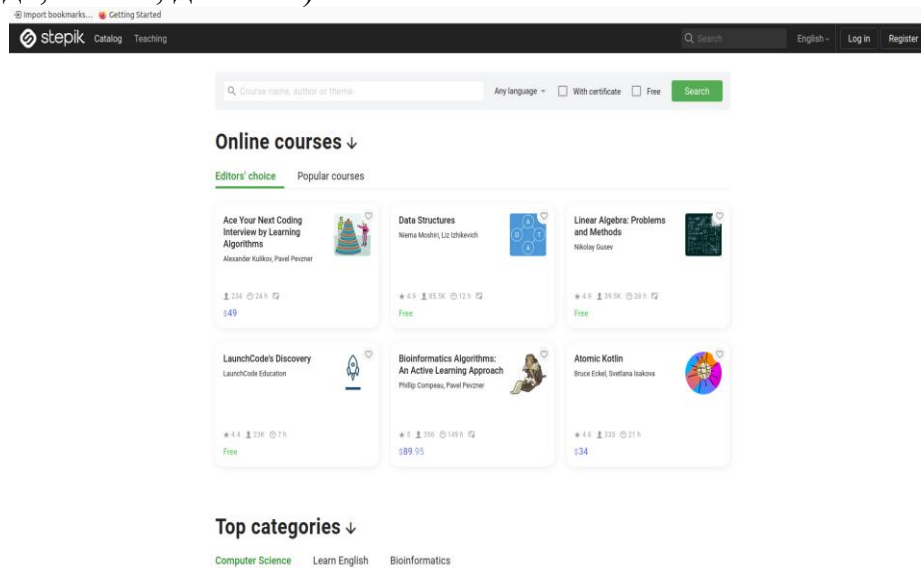


Рисунок № 43. Stepik.

### 2) <https://stepik.org/course/71759>

Введение в Интернет вещей (Stepik, 10 уроков) – ученик может познакомиться с концепцией IoT и ее ключевыми составляющими – датчики, беспроводные технологии, программное обеспечение, облачные технологии и др.

3) <https://stepik.org/course/95890/promo>

Интернет вещей (Stepik, 18 уроков) – ученик может научиться программировать на платформе Arduino, управлять работой электрических схем, датчиков и устройств, а также изучит принципы создания веб-приложений на базе платформы для интернета вещей PTC Thingworx.

4) [https://www.youtube.com/playlist?list=PLMiVLCIzZDbRu\\_oJghv6Yp6TeGNHuhYPU](https://www.youtube.com/playlist?list=PLMiVLCIzZDbRu_oJghv6Yp6TeGNHuhYPU)

IoT / ИIoT: Всеобъемлющий Интернет. Введение в Интернет вещей 2020 (YouTube, 16 роликов) – теоретические и практические стримы по IoT: подключение IoT к сети, Python, Jupyter Notebook и PL-App и Arduino.

5) <https://stepik.org/course/102886/promo>

Разработка умных устройств на базе Arduino – ученик может узнать, что такое Интернет вещей и из чего он состоит. Познакомится с платформой Tinkercad и научится моделировать электрические цепи с её помощью. Выполнит лабораторные работы с использованием Arduino.

6) <https://openedu.ru/course/ITMOUniversity/INTROIOT/> (Рисунок 44)

Введение в Интернет вещей (Открытое образование, ИТМО 80-100 ч.) – ученик может овладеть основными понятиями и терминологией в области Интернета вещей, протоколами стандартов и многоуровневых сетевых моделей. Научится разрабатывать и отлаживать устройства на Arduino.

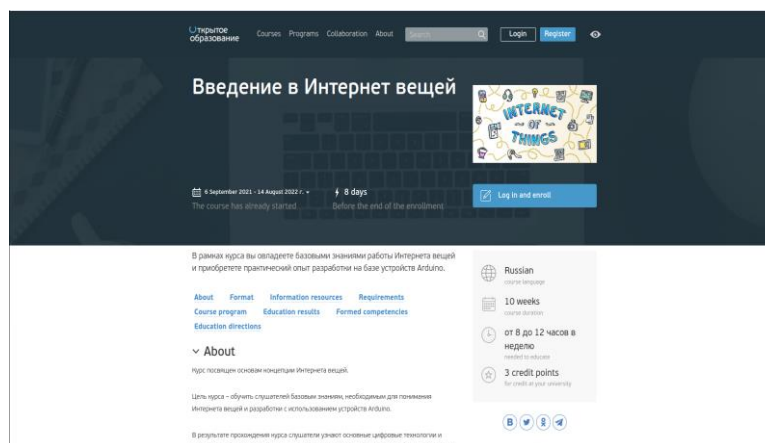


Рисунок № 44. Openedu.

7) <https://openedu.ru/course/ITMOUniversity/EMBSYS/>

Встроенные системы (Открытое образование, ИТМО, 80-100 ч.) – ученик поймет принципы функционирования встроенных систем – база киберфизических систем и Индустрии 4.0, – ученик научится выбирать средства и технологии их разработки и оценивать эффективность их использования в различных прикладных областях.

8) <https://www.coursera.org/learn/proektirovaniya-prilozhenij-interneta-veshchey?ranMID=40328&ranEAID=SAyYsTvLiGQ&ranSiteID=SAyYsTvLiGQ-cO8GUo.QCximqIZysw7xg&siteID=SAyYsTvLiGQ-cO8GUo.QCximqIZysw7xg#about>

Основы проектирования приложений интернета вещей (Coursera, 49 видео) – курс от МФТИ дает общий обзор всей технологической цепочки разработки приложений интернета вещей, организации процесса разработки и распределения ролей в команде проекта.

9) <https://www.coursera.org/learn/roboty-arduino> (Рисунок 45)

Строим роботов и другие устройства на Arduino. От светофора до 3D-принтера (Coursera, 87 видео) – пройдя этот курс МФТИ, ученик сможет проектировать устройства, которые считывают данные о внешнем мире с разнообразных датчиков, обрабатывают информацию, получают и отправляют данные на ПК, в Интернет, на мобильные устройства. Создание устройств будет включать проектирование, изучение компонентов, сборку схем, написание программ, диагностику. Также ученик разберется с устройством и работой FDM 3D-принтера.

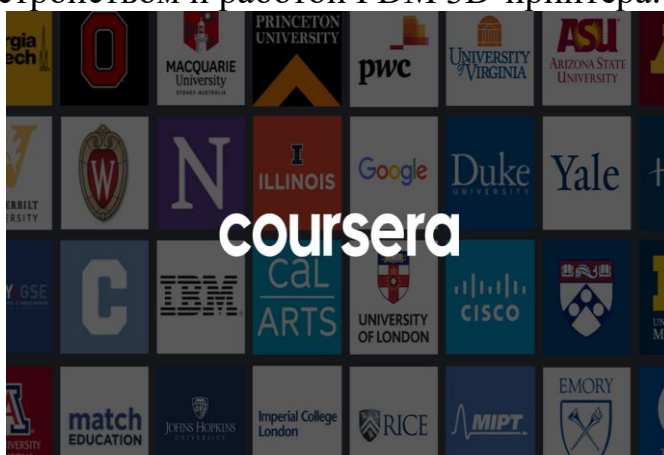


Рисунок № 45. Coursera.

10) <https://www.samsung.com/ru/iotacademy/programm/> (Рисунок 46)

IoT академия Samsung (2 семестра, 80 ч.) – бесплатное обучение продолжительностью 1 год для студентов вузов-партнеров программы. На базе вузов-партнеров оборудованы специализированные IoT лаборатории, где установлены мобильные и носимые устройства Samsung, наборы для быстрого прототипирования и микрокомпьютеры Artik [12].

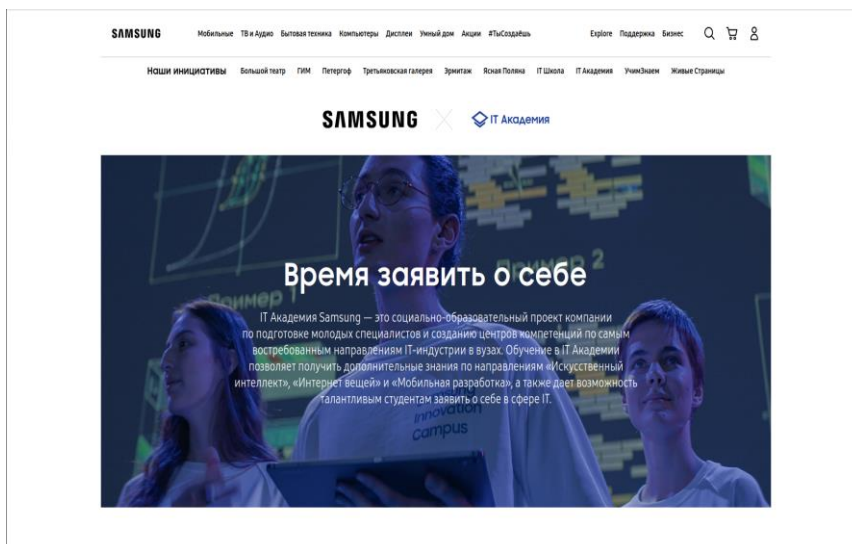


Рисунок № 46. IoT Academy.

1) <https://www.udemy.com/course/raspberry-pi-workshop-become-a-coder-maker-inventor> (Рисунок 47)

Raspberry Pi воркшоп: ученик может стать кодером и изобретателем (Udemy, 29 лекций) – ученик может узнать об экосистеме Raspberry Pi, как кодить на Питоне, использовать GPIO, писать сценарии Shell, планировать задачи, создавать настольные приложения с графическим интерфейсом.

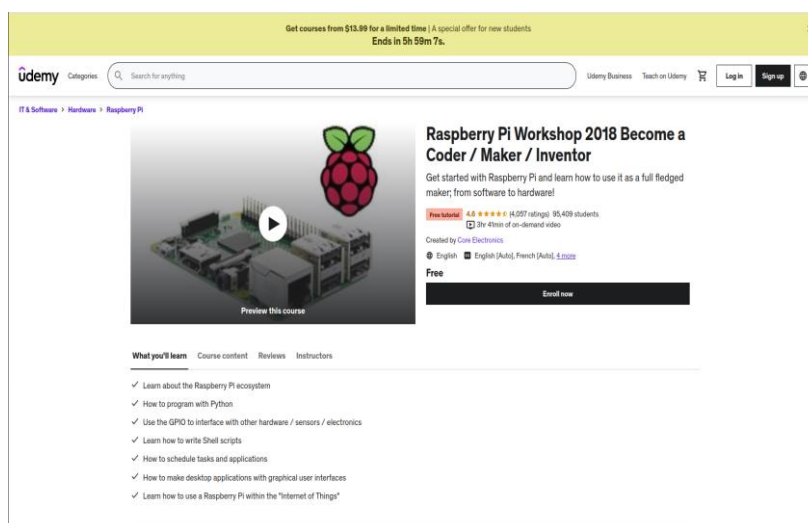


Рисунок № 47. Udemy.

2) <https://www.netacad.com/ru/courses/iot/introduction-iot> (Рисунок 48)

Введение в IoT (Сетевая академия Cisco, 20 часов) – ученики могут познакомиться с IoT, поймут, как он меняет бизнес-процессы, изучит проблемы безопасности, которые нужно принимать во внимание при реализации Интернета вещей.



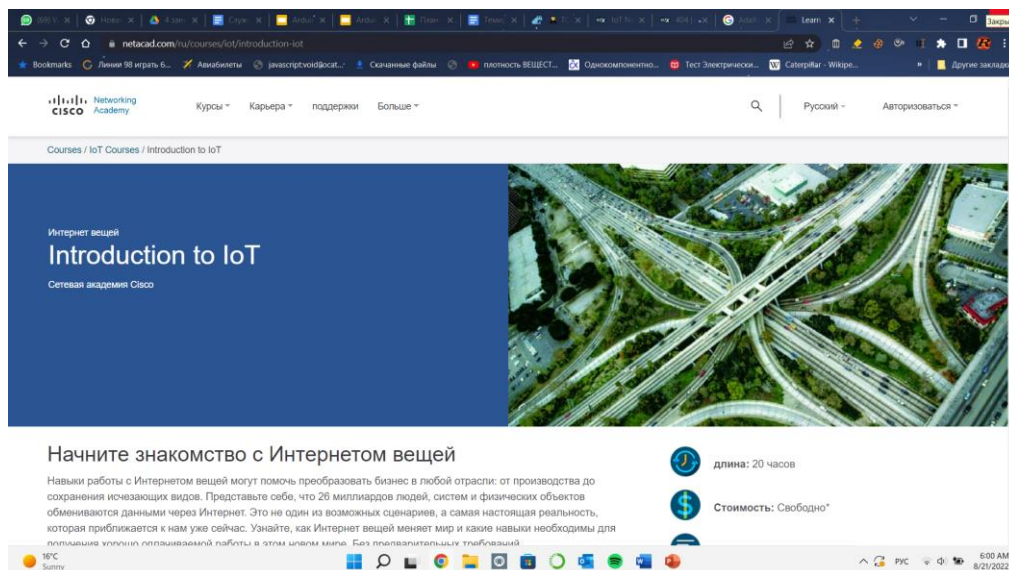


Рисунок № 48. Netacad.

3) <https://www.coursera.org/learn/iot>

Введение в программирование интернета вещей (Coursera, 38 видео) – проектирование, создание и внедрение IoT-устройств с использованием платформ Arduino и Raspberry Pi.

4) <https://www.coursera.org/learn/aws-iot-developing-and-deploying-an-internet-of-things>

AWS IoT: Разработка и развертывание Интернета вещей (Coursera, 29 видео) – ученики могут научиться разворачивать IoT-устройства, интегрированные с такими сервисами, как AWS IoT Core, AWS IoT Device Management и AWS IoT Analytics и как собирать, обрабатывать и анализировать данные с устройств IoT в облаке.

5) <https://www.coursera.org/learn/introduction-iot-boards>

Введение и программирование IoT (Coursera, 14 видео) – введение в IoT, сетевые технологии и IoT, программирование Arduino, Raspberry Pi и ARTIK Board.

6) <https://www.coursera.org/learn/iioot-google-cloud-platform>

Промышленные IoT в облаке Google (Coursera, 29 видео) – ученики могут познакомиться с архитектурой Google IoT. Ученики могут реализовать передачу данных от устройства к BigQuery и Datarprep.

7) <https://www.coursera.org/learn/industrial-internet-of-things>

Промышленный Интернет вещей (IIoT) (Coursera, 35 видео) – на этом курсе ученики познакомятся с концепцией промышленного Интернета вещей (IIoT), узнают, как он применяется в производстве, и что следует учитывать предприятиям при принятии решения о внедрении этой технологии.

8) <https://www.coursera.org/learn/iot-devices-il>

IoT Devices (Coursera, 28 видео) – ученики изучат программирование устройств IoT (Arduino и Raspberry Pi), датчики, стеки протоколов IoT



(Zigbee, 5G, NFC, MQTT и т.д.), обеспечение безопасности, науку о данных для IoT и облачные платформы IoT, такие как AWS IoT.

9) <https://www.coursera.org/learn/iot-wireless-cloud-computing>

IoT (Интернет вещей) беспроводные и облачные вычисления (Coursera, 51 видео) – на этом курсе ученики выполняют проекты, которые учат анализировать беспроводные сети Bluetooth и Wi-Fi, настраивать и использовать облачные сервисы EC2 (Elastic Compute Cloud) в AWS (Amazon Web Service).

10) <https://www.coursera.org/learn/iot-software-architecture>

Архитектура программного обеспечения для Интернета вещей (Coursera, 31 видео) – рассказывают о лучших практиках проектирования сложных IoT-систем.

11) <https://www.coursera.org/learn/iot-cyber-security>

Кибербезопасность и Интернет вещей (Coursera, 15 видео) – ученики узнают, как обеспечить безопасность промышленных, домашних и носимых IoT.

12) <https://www.coursera.org/learn/smart-cities>

Умные города: управление умными городскими инфраструктурами (Coursera, 36 видео) – рассказывают, как перейти от устаревшей городской инфраструктуры к интеллектуальным городским системам.

13) <https://www.edx.org/course/iot-networks-and-protocols>

Сети и протоколы IoT (edX, 6 модулей) – научит определять, какие протоколы лучше всего подходят для различных приложений IoT, проектировать базовую сеть для своих собственных идей IoT.

14) [https://www.edx.org/course/a-subjective-introduction-to-the-iot?source=aw&awc=6798\\_1635685728](https://www.edx.org/course/a-subjective-introduction-to-the-iot?source=aw&awc=6798_1635685728)

Введение в IoT (edX, ИТМО, 7 модулей) – курс университета ИТМО, на которым поможет разобраться в типах IoT-устройств, сценариях их использования, датчиками и программировании на C++.

15) <https://www.edx.org/course/iot-networking-and-fog-layer-devices>

Сетевые технологии IoT и туманные вычисления (edX, ИТМО, 3 модуля) – дают общую информацию о топологиях сетей, аппаратном обеспечении для платформы Raspberry Pi и учат программировать на Raspberry Pi, используя Python (Raspbian OS) и C# (Windows 10 Core IoT OS) [12].

## Список использованной литературы:

1. Почему лучше программировать на Linux. 04.10.2021 //losst.ru: [сайт]. URL: <https://losst.ru/pochemu-luchshe-programmirovat-na-linux>
2. Что такое Ubuntu? 16.11.2020 //eurobyte.ru: [сайт]. URL: <https://eurobyte.ru/articles/chto-takoe-ubuntu/>
3. Выбираем самый удобный редактор кода Python. 06.11.2020 //habr.com: [сайт]. URL: <https://habr.com/ru/company/skillfactory/blog/521838/>
4. LEGO MINDSTORMS Education EV3 + MicroPython: программируем детский конструктор взрослым языком. 07.10.2019 //habr.com: [сайт]. URL: <https://habr.com/ru/company/legoeducation/blog/474562/>
5. Топ-20 бесплатных ресурсов для изучения Python: сохрани это в закладки. 08.01.2021 //proglib.io: [сайт]. URL: <https://proglib.io/p/top-20-besplatnyh-resursov-dlya-izucheniya-python-sohrani-eto-v-zakladki-2021-01-08>
6. ИКТ по работе с Ардуино. URL: [https://nsportal.ru/sites/default/files/2016/11/19/programmirovanie\\_mikrokontrollov\\_na\\_baze\\_arduino\\_vo\\_vneurochnoy\\_deyatelnosti.pdf](https://nsportal.ru/sites/default/files/2016/11/19/programmirovanie_mikrokontrollov_na_baze_arduino_vo_vneurochnoy_deyatelnosti.pdf)
7. Tinkercad Arduino – лучший онлайн симулятор Ардуино на русском. //arduinomaster.ru: [сайт]. URL: <https://arduinomaster.ru/program/simulyator-arduino-tinkercad-circuits/>
8. 7 лучших симуляторов Arduino для ПК на сегодня. 17.09.2018 //gadgets-help.com: [сайт]. URL: <https://gadgets-help.com/windows/7-luchshikh-simulyatorov-arduino-dlia-pk-na-segodnia/>
9. Как научиться программировать на C++ для Arduino: хорошие онлайн-программы обучения и бесплатные ресурсы. 10.05.2020 //howtolearn.ru: [сайт]. URL: <https://howtolearn.ru/online-kursy/arduino.html>
10. Лучшие сайты Arduino 2022 года. 28.03.2022 //3dgram.ru: [сайт]. URL: <https://3dgram.ru/luchshie-sajty-arduino-2022-goda/>
11. Топ-10 инструментов IoT-разработки в 2020. 01.11.2020 //habr.com: [сайт]. URL: <https://habr.com/ru/company/ruvds/blog/525798/>
12. Выбираем самый удобный редактор кода Python. 21.11.2021 //proglib.io: [сайт]. URL: <https://proglib.io/p/top-30-besplatnyh-uchebnyh-kursov-dlya-iot-razrabotchika-ot-novichka-do-professionala-2021-11-21>

**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ  
ПО ИЗУЧЕНИЮ КУРСА «РОБОТОТЕХНИКА»  
В 10-11 КЛАССАХ**

Подписано в печать 21.10. 2022 г. Формат 60×84 1/16.  
Бумага офсетная. Печать офсетная.  
Шрифт Times New Roman. Усл. п.л. 19