

# НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Департамент общего образования Томской области  
ОГБУ «Региональный центр развития образования»  
АНО ДО «Детский технопарк «Кванториум»  
Департамент образования администрации г. Томска  
МБОУ лицей при ТПУ г. Томска

## **СБОРНИК ТРУДОВ**

XXIV Всероссийской конференции-конкурса  
исследовательских работ старшеклассников  
«Юные исследователи – науке и технике»

27 – 29 марта 2023 г.

Издательство  
Томского политехнического университета  
Томск 2023

**УДК 371.388.6(063)**

**ББК 74.202.7л0**

**Ю751**

Юные исследователи – науке и технике: сборник трудов XXIV Всероссийской конференции-конкурса Исследовательских работ старшеклассников «Юные исследователи – науке и технике»; Томский политехнический университет. – Томск: Изд-во Томского политехнического университета, 2023

В сборнике трудов представлены материалы работ школьников.

Сборник представляет интерес для школьников, занимающихся исследовательской и проектной деятельностью.

В сборник включены статьи, представленные в Оргкомитет конференции и заслушанные на конференции.

## ТЕЛЕГРАМ-БОТ «ТЕОРИЯ ВЕРОЯТНОСТИ И КОМБИНАТОРИКА»

*Абдрахманова Адель*

*Коммунальное государственное учреждение «Средняя многопрофильная школа №37» отдела образования по городу Усть-Каменогорску управления образования ВКО, Казахстан*

Руководитель: Клименко Елена Сергеевна, учитель информатики, категория «педагог-исследователь»

Говоря об информационных и мобильных технологиях, мы обычно подразумеваем всевозможные смартфоны, гаджеты, приложения и тд. Это всё интересно и увлекательно, но давайте задумаемся о таком важном аспекте, как вид и степень влияния мобильных технологий на нашу повседневную жизнь. Еще несколько десятилетий назад мы и не предполагали, что мобильные устройства плотно войдут в нашу жизнь и станут действительно мобильными.

С тех пор технологии продвинулись невероятно далеко. Мобильные устройства стали меньше, мощнее и гораздо полезнее. Они проникли во все сферы нашей жизни, и их роль продолжает расти. Доступность всевозможных смартфонов, планшетов и других гаджетов способствует их быстрому распространению по всему миру. И, само собой, все эти миллиарды мобильных устройств оказывают большое влияние на качество нашей жизни.

Мобильное обучение (m-learning) – современная технология дистанционного образования с использованием портативных устройств. Новая методика позволяет быстро и эффективно обучаться с минимальными затратами. Основные преимущества m-learning – возможность учиться, находясь в любом месте, доступность обучения. Интеграция мобильной технологии в дистанционное образование делает процесс обучения непрерывным, обеспечивает сохранность данных. Обучающий контент представляет оптимальные для быстрого усвоения материалы. Учеба превращается в интересное и увлекательное занятие.

Меня стало интересовать это направление. И пришла идея создать бота для обучения какой-нибудь темы. Почему бот? Это на сегодняшний день это один из популярных механизмов, интерес к которому возрастает все больше. Это самое быстрое средство коммуникации. Статистика подтверждает, что по мнению 87.2% респондентов, потребители имеют нейтральный или благоприятный опыт общения с чат-ботами. Тему для мобильного обучения я выбрала Теория вероятности и комбинаторика, т.к. это одна из сложных тем математики, которая тяжело дается учащимся для изучения. Тем самым надеюсь, что мой бот поможет в ее освоении.

Цель исследования – создать телеграм-бот для учащихся по теории вероятности и комбинаторике.

Задачи исследования:

1. изучить назначение ботов, их функции и возможности;
2. выбрать программу-конструктор для создания бота;

3. подобрать теоретический материал и математические задания для телеграм бота;
4. создать бота по математике;
5. провести апробацию бота для учащихся.

Объект исследования – возможности использования мобильных приложений для обучения.

Предмет исследования – использование телеграм-бота для изучения тем по математике.

Гипотеза - если использовать телеграм-бот для мобильного обучения, то повышаются знания учащихся, т.к. обучение становится доступным и оптимальным для самостоятельного изучения.

«Бот» (сокращение от «робот») – это программа, выполняющая автоматические заранее настроенные повторяющиеся задачи. Боты обычно имитируют поведение пользователя или заменяют его. Они являются автоматизированными, потому работают намного быстрее, чем пользователи. Роботы выполняют полезные функции, например, обслуживание клиентов или индексация поисковых систем.

Про бот можно сказать и другими словами, например. Боты – это маленькие умные помощники. Их прелесть в том, что вам не нужно покидать экосистему мессенджера, чтобы решить элементарную задачу: спросить, уточнить, узнать, посчитать. Каждый бот сам по себе, но вместе они образуют вашу личную армию ассистентов, каждому из которых вы даете поручение, а они безропотно и четко выполняют ваши пожелания.

Самих помощников и отличают по их функционалу, например, *чат-бот*, который состоит из обычного чата, имитирующего общение на определенную тему; *боты-информаторы*, которые доставляют до своих читателей определенные новости: от погоды за окном, до мероприятий, происходящих в другом конце страны; *игровые боты*, которые представляют собой текстовые версии самых разных игр; *боты-ассистенты* – это помощники, которые привязаны к веб-версии сайта или приложения, и являются лишь дополнением к основной части.

Функции ботов также разнообразны как и их виды: поиск и обмен файлов (поиск различных файлов с множества источников, например, книг, фильмов); актуальные новости (свежая информация о происходящем в мире, о погоде за окном, и курсе доллара в данную секунду); интеграция с другими сервисами (бот поможет управлять умным домом, например); развлечение (скрашивание досуга с помощью смешных картинок).

Для начала работы, нужно выбрать бот из каталога и перейти по его ссылке, или же, найти по имени через поиск. Чтобы завести переписку, нужно зайти в диалог, и отправить стартовое сообщение. В основном, это команда /start. Во многих ботах данное кодовое слово спрятано за кнопкой виртуальной клавиатуры. После чего, робот пришлет вам информацию о себе, инструкции, свои возможности и список функций.

На сегодняшний день существует множество возможностей для создания телеграм бота.

Ботов можно программировать. Среди разработчиков ботов лучшей библиотекой для ботов на Python считается aiogram. Она асинхронная, использует декораторы и содержит удобные инструменты для разработки.

Также ботов часто пишут на JavaScript, для этого обычно используется Telegraf. Библиотеки есть и для многих других языков, но используют их реже.

А нужны ли навыки программирования для создания ботов? Вовсе не обязательно. Существуют различные сервисы и конструкторы, которые позволяют создать собственного бота без знаний специальных кодов. Множество сервисов предлагают различные возможности, но и характеристики у них различны.

Во-первых, есть сервисы бесплатные или платные. А также условно бесплатные, т.е. ограниченным функционалом можно пользоваться, а для полноценного использования нужно оплатить. Либо бесплатная версия распространяется на достаточно короткий срок, что нерационально для долгого применения.

Во-вторых, есть сервисы с собственной рекламой, которая появляется в непредсказуемый момент и отвлекает от работы. К тому же реклама может быть не всегда этичной. При чем отключить ее невозможно.

В-третьих, не у всех конструкторов есть мобильная версия, что является достаточно неудобным фактом при разработке.

После поисков нужного конструктора и проб различных сервисов мой выбор остановился на Manybot.

Manybot — это: создание собственных команд; создание меню и подменю; рассылка новостей подписчикам; 6 языков интерфейса; формы для заказа и обратной связи.

Текст, картинки, видео, файлы — можно использовать любой формат. Здесь нет ограничений на количество сообщений в одной рассылке, так что вы сможете составлять сложные посты и быть уверены, что они дойдут быстро и в нужном порядке.

Пользовательские команды позволяют боту отвечать заранее заданным образом. Можно использовать пользовательские команды, чтобы добавить интерактивные функции в бот. Меню позволяет пользователям использовать бот без ввода команд. Также очень быстро можно создать красивые меню и подменю.

Созданный бот по теории вероятности и комбинаторике содержит три основных раздела: комбинаторика, теория вероятности, тесты.

Комбинаторика состоит из теории, которая включает основные понятия, размещение, перестановку, сочетание. Также имеется возможность отдельно просмотреть формулы. Для повторения и закрепления теоретического материала встроены ссылки на видео youtube. Задания на закрепление представлены тремя видами. Имеется описание “Зачем нужна комбинаторика”.

Раздел “Теория вероятности” в теоретической части отражает классическое определение вероятности, события, совместные и независимые события, сложение и умножение событий. Также имеется возможность отдельно просмотреть формулы и выполнить задания на закрепление.

Третий раздел “Тесты” позволяет проверить свои знания по комбинаторике и теории вероятности, выполняя различные тестовые задания. Для выполнения заданий добавлены ссылки, с помощью которых можно перейти на выполнение.

Тема Теория вероятности и комбинаторика является одной из сложных тем математики, которая тяжело дается учащимся для изучения. К тому же данная тема изучается по несколько уроков с 9 по 11 класс и чаще всего учениками забывается к следующему году.

Созданный телеграм-бот был апробирован для учащихся 9-11 классов. В результате чего получил положительные отзывы старшеклассников. Учащиеся проявили интерес к данному приложению.

Считаем, что гипотеза подтвердилась, использование телеграм-бота повышает знания учащихся, т.к. обучение становится доступным и оптимальным для самостоятельного изучения.

## Приложение

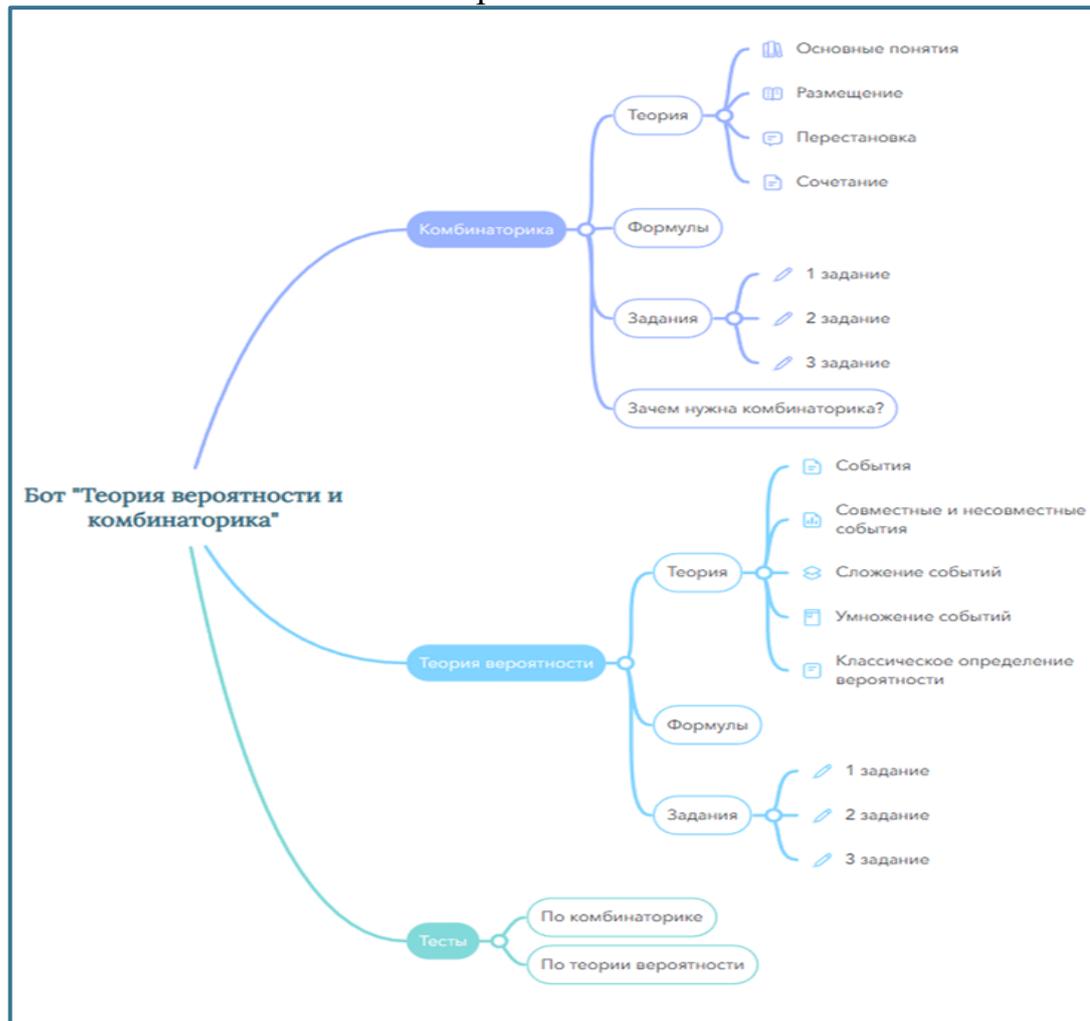


Рисунок 1. Схема телеграм-бота



Рисунок 2. QR-код для подключения к телеграм-боту

**МОДЕЛИРОВАНИЕ И ПРАКТИЧЕСКАЯ РАЗРАБОТКА НОВОГО ЛОГОТИПА ТПУ С ПОМОЩЬЮ ЭЛЕКТРОНИКИ И СВАРКИ**

*Ажермачёв Роман, Наставко Владислав, Плавинский Григорий,  
Прибытков Данил*

*Муниципальное бюджетное общеобразовательное учреждение лицей при  
ТПУ, 10 класс*

*г. Томск*

Руководитель: Скрипко Степан Игоревич, Ассистент ОЭИ ИШНКБ ТПУ

В настоящее время у каждой компании и организации существует своя отличительная символика (логотип или бренд). У ТПУ также есть логотип, но 19 сентября 2022 года он официально изменился благодаря известному российскому дизайнеру Артемию Лебедеву и его студии [рисунок 1], а с 10 декабря 2014 года до этой даты логотип выглядел иначе [рисунок 2]. На данный момент существует только реальное представление логотипа 2014 года, поэтому мы решили воплотить в реальности логотип 2022 года.



*Рисунок 1. Новый логотип ТПУ, использующейся после 19 сентября 2022 года*



*Рисунок 2. Логотип ТПУ, использовавшийся с 10 декабря 2014 года по 19 сентября 2022 года*

Актуальность проекта. Так как на данный момент наяву существует только логотип 2014 года можно сделать вывод, что логотип от 2022 года никто не смог воплотить в жизнь. Исходя из этого можно понять, что на данный момент практическая разработка нового логотипа ТПУ является актуальной задачей.

Цель проекта. Практическая разработка нового логотипа ТПУ при помощи электроники и сварки.

Задачи:

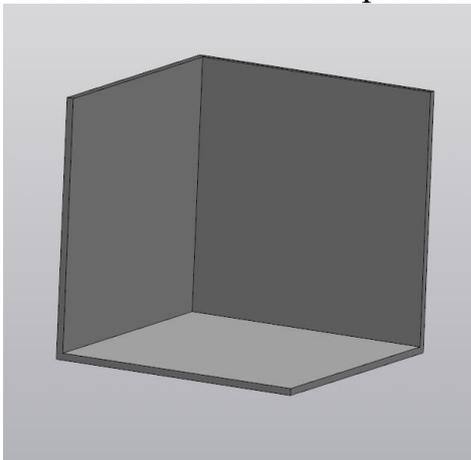
1. Разработать трёхмерную модель логотипа с помощью программы для моделирования.
2. Определиться с материалом и методом работы с ним для создания логотипа.
3. Создать макет логотипа для проверки модели.
4. Разработать подставку для модели.

5. Сварить детали логотипа и выровнять возможные неровности.
6. Провести финальную обработку модели и вставить детали модели в подставку.

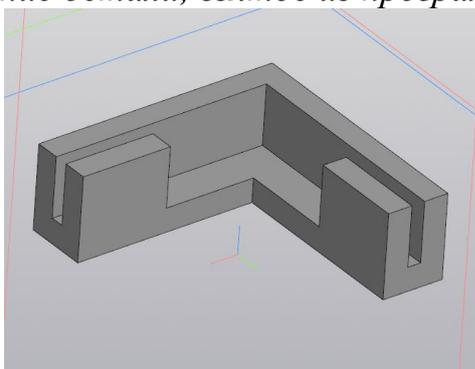
Разбор аналогов. На данный момент не существует полных аналогов нашей работы, чтобы мы смогли использовать его в качестве референса для выполнения работы.

Целевая аудитория. Тема нашего проекта предоставлена нам от ТПУ, значит заказчиком также является ТПУ.

Используемые материалы, методы, оборудование. Наша модель состоит из девяти деталей и подставки. Подставка, представляющая собой фигуру сложной формы со множеством пазов под детали, смоделирована в программе КОМПАС-3D v21 и напечатана на 3D принтере Voron v2.4r2. Каждая из девяти деталей представляет собой три грани куба, прилегающие к одной вершине, с длиной ребра равной три сантиметра. Детали состоят из трёх пластин нержавеющей стали марки 12х18, сваренные аргонно-дуговой сваркой при помощи сварочного аппарата модели Orion mPulse 100с перпендикулярно друг другу [рисунок 3] при помощи специального крепления [рисунок 4].



*Рисунок 3. Изображение детали, взятое из программы КОМПАС-3D v21*



*Рисунок 4. Изображение приспособления для сварки, взятое из программы КОМПАС-3D v21*

Этапы выполнения проекта:

1. Создание 3D-модели нового логотипа ТПУ.
2. Создание 3D-модели нового логотипа ТПУ производилось в программе КОМПАС-3D v21. На данном этапе было перепробовано множество вариантов деталей и их расположения. Так как логотип не предоставляет

нам полную информацию о форме деталей и об однозначном расположении их в пространстве, поэтому нам пришлось обратиться к референсам логотипа, представленным на сайте Артемия Лебедева [1]. Также нам пригодились собственные знания в геометрии.

3. Выбор материала и метода работы с ним.
4. Выбор материала и метода работы с ним был очевиден, так как тема нашего проекта ограничивала нас в материалах и методах их обработки. Из-за того, что нам нужен был стойкий к коррозии металл, который будет просто сваривать и, при этом итоговые детали должны быть достаточно прочными при физических взаимодействиях (например, при случайном падении детали), поэтому мы выбрали нержавеющую сталь марки 12х18. Так как нам нужно было сваривать достаточно тонкие и небольшие пластинки нержавеющей стали, то нам подошёл аргоно-дуговой сварочный аппарат модели
5. Создание макета для проверки модели.
6. Создание макета 3D-модели логотипа показало недочёты в нашей конструкции и после этого 3D-модель пришлось изменить и разработать подставку.
7. Разработка подставки.
8. Подставка была разработана в программе КОМПАС-3D v21 и впоследствии распечатана на 3D принтере Voron v2.4r2. Она представляет собой фигуру сложной формы со множеством пазов под детали.
9. Сварка деталей и устранение возможных неровностей.
10. Сварка деталей проходила достаточно просто благодаря приспособлению [рисунок 4], разработанному специально для сварки двух пластин под прямым углом. Устранять возможные неровности не потребовалось потому, что мы сварили избыточное количество деталей и выбрали из них наилучшие.
11. Финальная обработка деталей и вставка их в подставку.

Финальная обработка деталей заключалась в чистке деталей от царапин и отпечатков пальцев. Со вставкой деталей в подставку не возникло никаких сложностей, так как она была рассчитана с учётом погрешности 3D печати и усадки пластика.

Предоставление результата. 3D-модель продукта нашего проекта представлен на изображении ниже [рисунок 5].

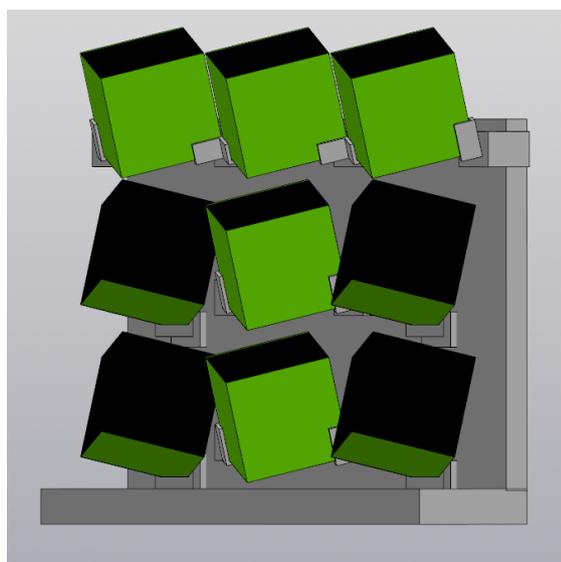


Рисунок 5. Изображение 3D-модели нового логотипа с подставкой, взятое из программы КОМПАС-3D v21

Заключение.

Нам кажется, что наш проект является экономически выгодным, так как затраты на продукт не превышают 5500 рублей [Таблица 1]

Название	Цена, руб.	Итоговая стоимость, руб.
Нержавеющая сталь 12x18, м <sup>2</sup>	3500	5180
Нарезка стали	770	
Газ для сварки (аргон)	360	
Пластик для 3D печати, м <sup>3</sup>	250	
Электроды для сварки	300	

«Таблица 1: Таблица расходов»

В процессе создания логотипа были получены навыки командной работы, изучено 3D-моделирование в программе КОМПАС-3D v21, получены навыки в сварочном деле. Итоговая цель работы была достигнута – создан новый логотип из стали.

#### СПИСОК ЛИТЕРАТУРЫ

1. Логотип Томского политехнического университета [Электронный ресурс] - Режим доступа: <https://www.artlebedev.ru/tpu> (дата обращения 3.03.2023)
2. Юхин Н.А. Ручная дуговая сварка неплавящимся электродом в защитных газах (TIG/WIG) Сварка и сварщик - Страницы15-33. [электронный ресурс] – режим доступа: <https://welding.com/yuhin-ruchnaya-dugovaya-svarka-neplavyashchimsya-elektrodom-zashchitnyh-gazah-tigwig?ysclid=lfbbc57s2e115479556>
3. Рисунок 1: в томском политехе презентовали новый логотип <https://news.tpu.ru/news/2022/09/19/41233> (дата обращения: 24.11.2022)

4. Рисунок 2: томский политехнический университет представил новый логотип [Электронный ресурс] - Режим доступа: <https://news.tpu.ru/news/2014/12/10/22549/> (дата обращения: 13.12.2022)

## **ПРОГРАММА-ТРЕНАЖЁР ПО ТЕМЕ «АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ»**

*Андреев Артемий*

*МБУДО г. Иркутска ЦДТТ, МАОУ Лицей ИГУ г. Иркутска*

*Руководители: Рейнгольд Григорий Борисович, педагог доп. образования*

*МБУДО г. Иркутска ЦДТТ, Медведева Елена Николаевна, учитель информатики МАОУ Лицей ИГУ г. Иркутска*

Профессия учителя содержит большую рутинную составляющую. Это можно отнести к любым предметам: и к математике, и к физике, и к информатике. Среди подобных действий можно отметить составление проверочных работ и задач к ним. Нами было решено облегчить данную задачу для учителей информатики. Отсюда была выведена цель работы.

Цель: создать программу-тренажер по теме алгоритмизация и программирование.

Задачи:

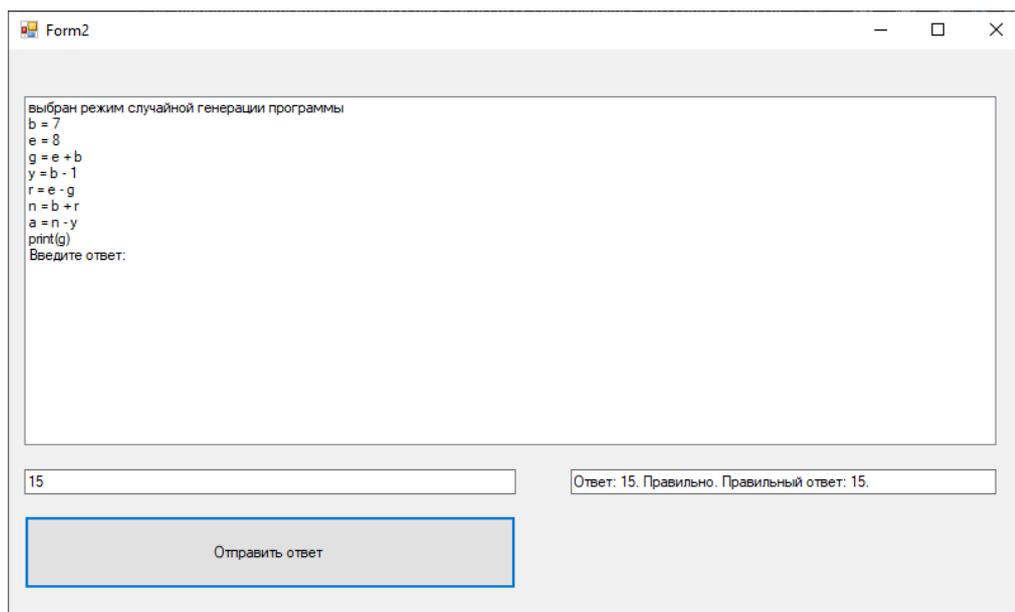
1. Провести библиографический обзор;
2. Найти возможные методы решения поставленной цели;
3. Реализовать метод случайной генерации программного кода по шаблонным алгоритмам;
4. Найти разработкам практическое применение.

В Интернете реализации подобных программ найти не удалось, однако была найдена работа ученика нашего объединения Андрея Степанова “Тестирующая программа по разделу “Алгоритмизация и программирование” школьного курса информатики” [4]. Она датируется 2012 годом, то есть с момента ее реализации прошло более десяти лет. Программа была написана на языке Basic, могла генерировать программы на языках Pascal и Basic, по темам линейные конструкции, ветвления и циклы. К сожалению, продолжения работы не последовало, и она не была доведена до готовности к внедрению в учебный процесс. Саму программу можно назвать устаревшей, так как языки Basic и Pascal во многом отстают от более современных языков и сейчас практически не используются.

В ходе библиографического обзора были найдены два наиболее эффективных метода реализации подобной цели. Первый – метапрограммирование [1]. Если говорить более конкретно, то метапрограммирование шаблонов [2]. При использовании этого метода, компилятор действовал бы как “виртуальная машина”, генерирующая команды по данным шаблонам. Эти команды, в свою очередь, образуют конечную программу. Подобный метод имеет преимуще-

ства в виде малого объема написания программного кода и своего быстродействия, однако его реализация и понимание довольно сложны. В качестве альтернативы метапрограммированию было найдено следующее решение – кодогенерация [3]. Суть его следующая: в программном коде по шаблонам генерируются строки, представляющие из себя команду для программы, в которую программой подставляются требуемые переменные и их значения. Далее эти строки записываются в исполняемый файл новой программы, являющийся продуктом генерации программы. Именно этот метод и был выбран как основной в нашей работе.

После выбора метода реализации цели мы занялись разработкой программы. Изначально она генерировала программу без какого-либо четкого алгоритма. Все команды выбирались случайно, размеры программы были ограничены 10 командами. В конце программы был вывод случайно выбранной из использованных переменных. Требовалось написать значение, полученное при выводе. Генерируемая программа была на языке Python, так как это довольно-таки легкий язык, подходящий для первоначального изучения алгоритмов и программирования. В то же время, генерирующая программа была написана на C#.



*Рисунок 1. Пример работы изначальной версии программы*

Работает программа следующим образом: сначала создаются массивы букв латинского алфавита, а также массивы символов в командах и в ветвлениях, количество возможных команд. Имена переменных, используемых в программе, выбирались случайно из массива букв латинского алфавита. Есть лишь две буквы, которые программа не берет в качестве имен переменных – *i* и *j*, которые были оставлены специально для циклов и ветвлений. Затем в программу задается расположение файла с расширением *.ру*, в котором и будет генерироваться программа. Ввод расположения файла возможен как в самом приложении, так и в тексте программы, а при небольшой модификации программы – из файла. Выбор режима программы был реализован позднее, после реализации иных режимов. В актуальных версиях программы первый режим

именуется как “random”. Первые две команды всегда присваивают двум случайным переменным случайные значения. Это было сделано для того, чтобы избежать возможности генерации программы без значений переменных. После этого начинался цикл, идущий от двух и до количества команд, в котором случайно выбиралось значение от 0 до 3. Каждому из этих значений соответствовала определенная команда. При 0 – присвоение переменной (не обязательно старой, возможно и новой) значения, значение определяется выражением с двумя переменными. При 1 – аналогично, однако значение определяется выражением с одной переменной из ранее использованных и одним числовым выражением. При 2 – ветвление, при этом в самом ветвлении имеется 50% шанс появления второй ветви else. Среди ветвлений были реализованы вложенные ветвления, ветвления со сложным условием (or, and). Количество вложенных ветвлений было ограничено единым экземпляром. При 3 – цикл for. Абсолютно все числовые значения выбирались случайно. Изначально с этим возникали проблемы: ограничение на размер числа было равно 100, а потому в ходе генерируемого алгоритма могли получаться огромные числа, подсчитать которые без калькулятора было довольно сложно. Программа была рассчитана на проверку знания работы языка программирования, а не на проверку умения считать большие числа, а потому ограничение на размер числовых значений было изменено со 100 до 20.

После генерации кода и его записи в .ру файл, программа открывала его с помощью процесса python.exe с параметром RedirectStandardOutput = true. Благодаря этому программа могла перехватить значение, подающееся на выход. Это значение сверялось с ответом, вписанном пользователем. Если они совпадают – выводилось “Правильный ответ”, если нет – “Неправильно” и правильный ответ. Подобным режимом мы решили не ограничиваться, а потому вскоре после его создания принялись за разработку иных режимов программы. Они заключались в следующем: по шаблону генерируется определенный алгоритм, в который подставляются случайные переменные и числовые значения, в то время как пользователь программы должен вписать значение, получаемое на выходе сгенерированной программы. По своей сути режимы работы очень схожи, однако новые режимы подходили для тренировки знания конкретных алгоритмов, в то время как старый – знание работы языка программирования на базовом уровне в целом.

Был составлен следующий список алгоритмов, тестирование которых происходит (в дальнейшем он будет пополняться):

1. Алгоритм Евклида
2. Алгоритм нахождения наименьшего общего кратного (в двух вариантах)
3. Алгоритм нахождения остатка от деления
4. Алгоритм суммы по циклу
5. Алгоритм нахождения суммы делителей числа
6. Нахождение количества делителей и проверки на то, является ли число простым

7. Хожжение по отрезку (даются две точки на числовой прямой и количество шагов, в первом варианте при достижении второй точки происходит разворот, во втором перемещение в начало, выводится точка, где оказался по прохождению этого количества шагов)

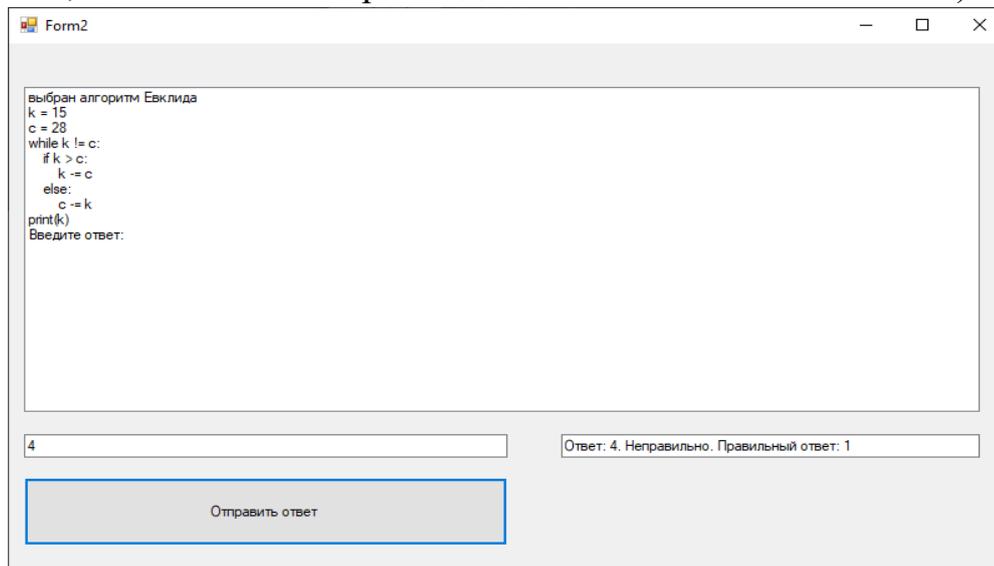


Рисунок 2. Пример работы нового режима программы (алгоритм Евклида)

Все эти алгоритмы были реализованы в отдельной программе на языке Python, из которой впоследствии были сделаны шаблоны для генерируемых программ. После тестирования программы были выявлены следующие преимущества:

1. Быстродействие работы. Генерация программы занимает от одной до трех секунд на среднестатистическом компьютере. На более мощных устройствах – менее секунды.
2. Простое использование. Так как программа на данный момент является консольным приложением, то ее использование очень просто: для выбора ответа, режима и файла нужно просто ввести текст.
3. Выбранный язык генерируемой программы (Python) легок и удобен в обучении алгоритмов.
4. Благодаря выбранному методу генерации программы можно легко добавлять новые алгоритмы для генерации, а также переделывать программу под другие языки.
5. После получения требуемого правильного ответа от сгенерированной программы она удаляется, что позволяет избежать подделки генерируемой программы с целью изменения ответов.

Практическое применение. Подобная программа имеет потенциал к практическому применению. Случайная генерация переменных и их значений сводит возможность списывания к минимальному значению. Подобная программа может использоваться для проверки понимания работы алгоритмов и языка программирования. При расширении количества языков для генерируемой программы генератор можно будет использовать в школах. Также эта программа может пригодиться для подготовки учеников к ОГЭ и ЕГЭ, в которых

фигурируют задания по теме алгоритмизация и программирование, ведь программа генерирует программы на Python, одном из допустимых на данных экзаменах языков. В дальнейшем планируется дополнение программы генерацией на языке C++, который также входит в перечень языков на ЕГЭ и ОГЭ

#### Заключение

Как итог, можно сказать, что цель выполнена. Создана первая версия программы-тренажера, генерирующей тесты для учеников по теме формальное исполнение алгоритмов. Генерируются программы со случайными командами, а также программы из перечня определенных алгоритмов. В качестве планов на будущее можно выделить:

1. расширение количества языков программирования, на которых генерируются программы (как минимум, добавить C++);
2. сделать более безопасным метод редактирования файла для исключения любой возможности подмена результатов;
3. реализация практического применение программы в учебных заведениях.

#### СПИСОК ЛИТЕРАТУРЫ

1. <https://ru.wikipedia.org/wiki/Метапрограммирование>
2. [https://en.wikipedia.org/wiki/Template\\_metaprogramming](https://en.wikipedia.org/wiki/Template_metaprogramming)
3. <https://ru.wikipedia.org/wiki/Кодогенерация>
4. Степанов Андрей Николаевич «ТЕСТИРУЮЩАЯ ПРОГРАММА ПО Разделу «АЛГОРИТМИЗАЦИЯ И Программирование» (Материалы X Российского соревнования юных исследователей «Шаг в будущее, ЮНИОР», М. 2012).

#### ЭЛЕКТРОННАЯ ОЧЕРЕДЬ НА АЗС НА МЕСТОРОЖДЕНИИ

*Анисимов Владимир*

*Муниципальное бюджетное общеобразовательное учреждение «Лицей №2»,  
10 класс, г.Братск*

Руководители: Малков Фёдор Сергеевич, лаборант-исследователь, Ткачёва Юлия Владиславовна, преподаватель курса проектной деятельности, Архипова Татьяна Юрьевна, учитель информатики, Миниханов Константин Дмитриевич, наставник по программе Сириус.Лето

Данный проект является проектом-заказом в рамках программы «Сириус. Лето». Заказчиком является Иркутская нефтяная компания, у которой есть месторождения. Поскольку боты являются инструментами, помогающими быстро решать поставленные задачи, заказчик предложил создать макет бота для регулировки потока автомобилей на АЗС. Этот проект является актуальным для заказчика, потому что на месторождении возникают проблемы с очередями на заправочные станции, поток больших грузовых автомобилей никак не регулируется, из-за чего возникает проблема с тем, что водители тратят много времени на заправку, образуются большие пробки.

Для решения этой проблемы подходит телеграмм-бот, который будет помогать водителям организовывать процесс заправки. Проблема, которую решает проект, заключается в отсутствии приложения, которое бы смогло снизить поток машин на АЗС на месторождении. Новизна данного проекта заключается в том, что приложение будет реализовано в виде телеграмм-бота, что позволит работникам удобно использовать его во время работы. Целью данного проекта является непосредственно создание макета телеграмм-бота, решающего проблему, описанную выше. При достижении цели необходимо будет выполнить следующие задачи:

1. Составить алгоритм для работы бота
2. Написать алгоритм на языке программирования Python
3. Провести испытание бота

В качестве источников информации будут использоваться статьи журналов о том, как использовать библиотеку Python`а «telebot» и «Pandas» (для работы с базами данных). В реализации данного проекта заинтересован заказчик, а также сами водители грузовых машин, потому что они будут использовать бота. Соответственно, целевой аудиторией проекта является заказчик.

#### Боты в мессенджерах. Теория

Бот по своей сути представляет программой, которая принимает сообщения пользователя, обрабатывает его каким-либо образом, и реагирует на это каким-либо образом. Цель ботов заключается в том, чтобы выполнять определённые функции за человека, бот выполняет функции намного быстрее, поскольку является программой. Боты используются в разных мессенджерах: Facebook, Вконтакте, Whatsapp. Исследовав статьи на тему ботов, мы выделили следующую классификацию ботов по тем задачам, которые они могут выполнять:

- Чат-боты (общение с пользователем на определённую тему)
- Боты-информаторы (информирование пользователя о каких-либо событиях) Игровые боты (дают пользователю возможность играть в определённую игру с ботом)
- Боты-ассистенты (создаются компаниями / сервисами для того, чтобы выполнять определенные функции, например, для упрощения взаимодействия с клиентами)

Опишем работу бота на примере бота в мессенджере Telegram. Пользователь отправляет сообщение, сообщение переходит на сервер телеграмма. Существует две технологии взаимодействия бота и сервера: Webhook и Long polling. Технология взаимодействия Webhook предполагает, что бот будет работать постоянно и Telegram будет сам отправлять ему обновления, на которые бот должен будет ответить, а Long Polling заключается в том, что бот во время работы спрашивает у Telegram`а об обновлениях и, если обновление действительно есть, отправляет соответствующий ответ. Для постоянной работы бота в условиях месторождения необходимо использовать технологию Webhook. Для организации работы бота по технологии Webhook`а необходим

http-сервер, доменное имя, “белый” ip-адрес, что будет обеспечивать постоянную работу бота и его ответы на сообщения пользователей. Для работы с ботом в языках программирования используются специализированные для этого модули. Модуль представляет собой какой-либо код для выполнения определённых задач, но для простоты использования другими программистами эти программы создаются как отдельный модуль, который можно вызвать, прописав его название в начале программы. В Python есть множество типов данных, с которыми можно производить различные операции. Есть типы данных, которые позволяют хранить в них множество значений. Такие структуры данных мы будем использовать в работе. Динамическими структурами данных являются те, которые могут быть изменены в ходе работы программы. Задача проекта заключается в создании программы, которая поможет регулировать поток автомобилей на АЗС, сделать процесс заправки удобнее и быстрее. Наш макет будет работать с помощью технологии Long Polling.

Исследуя информацию об использовании ботов в электронных очередях, было выяснено нами, что в Москве используется телеграм-бот для записи к врачу онлайн. Чтобы записаться к врачу с помощью этого бота, необходимо отправить номер полиса, дату рождения. Также в будущем планируется обучение данного бота общению с пользователем, рекомендации врача и назначению удобной даты.

#### Алгоритм работы телеграмм-бота

При разработке бота нам понадобится среда программирования PyCharm. В ней будет написан код на языке Python для телеграмм-бота. Для регистрации бота в телеграмме был использован бот BotFather.

Задачу регулировки потока машин на АЗС заказчик предложил решить с помощью бота в мессенджере Telegram. Заказчик предложил примерный алгоритм работы бота: бот сообщает информацию об очереди и примерном времени ожидания на АЗС, а водитель должен сам принять решение о заправке. Но алгоритм стоит дополнить дополнительными шагами. В итоге процесс выглядит так: бот прежде всего проверяет, находится ли пишущий ему пользователь в списке тех, кому разрешено ему писать. Для реализации системы идентификации необходимо было создать базу данных с ФИО сотрудников и их табельным номером на месторождении. Такая база данных будет условной и предназначена для испытания бота в рабочих условиях. Каждый сотрудник на месторождении знает наизусть свой табельный номер, поэтому система идентификации становится удобной, если сначала запрашивать у пользователя его ФИО, проверять, есть ли этот ФИО в списке сотрудников, затем запрашивать табельный номер. После успешного прохождения системы идентификации перед пользователем появляется выбор действий:

- Просмотр актуальной информации об АЗС (очередь, примерное время ожидания)
- Записаться в очередь / уже запрашиваюсь (пользователь с помощью кнопки сообщает о своём наличии в очереди на определённый АЗС, обновляя информацию)

- Сообщить об окончании заправки

При нажатии на кнопку записи пользователь выбирает АЗС, на которой он в очереди, а также колонку (стойку) на этой АЗС. При нажатии на кнопку об окончании заправки информация о колонке, на которой заправлялся пользователь, обновляется. При этом предусмотрено ошибочное нажатие на кнопку о записи и на кнопку об окончании заправки. Алгоритм предполагает работу бота с множеством пользователей. Необходимо хранить информацию о том, на каких АЗС и колонках находятся пользователи и находятся ли они на них вообще. Необходимо также регулировать идентификацию для каждого пользователя, что проще всего можно сделать, отсчитывая этапы идентификации для каждого пользователя. Эту задачу может решить такая структура данных в языке Python, как словарь (dict). Словарь представляет собой динамический тип данных, то есть его можно изменять. Словарь представляет собой тип данных, в котором есть ключи и значения, соответствующие ключам. Так же нам понадобится такая структура данных, как список. Список является динамической структурой данных, в котором значения располагаются по числовым индексам. Нам понадобятся три словаря: мониторинг информации об АЗС (ключи – номера АЗС, значения – список с очередями на колонках), мониторинг информации по пользователям (ключи – id пользователя в Telegram, значения – номер АЗС и номер колонки, на которой стоит пользователь или пустая строка, если пользователь не заправляется), мониторинг идентификации по пользователям (ключи – id пользователя в телеграмме, значения – этап идентификации, который прошёл пользователь). Для работы с базой данных в Python нам понадобится модуль Pandas. Этот модуль позволит нам импортировать данные формата \*csv, \*xlsx. При импортировании этих данных в программе будет создаваться особый тип данных – DataFrame, который представляет собой таблицу. В нём мы можем обращаться не только к колонкам, но и к конкретным ячейкам, мы можем фильтровать данные, строить графики по данным. При помощи Pandas мы считываем базу данных по сотрудникам. У нас есть две колонки: Name и ID. В колонке Name содержится ФИО сотрудника, все значения уникальны, а в колонке ID содержится табельный номер сотрудника, где тоже все значения являются уникальными. При запуске бота (т.е при нажатии на кнопку «Начать»), бот просит ввести ФИО. После отправки ФИО пользователем его id вносится в словарь, регулирующий идентификацию, пользователю присваивается значение 1, т.е пользователь прошёл первый этап. После этого проверяется наличие ФИО в списке сотрудников, для этого можно посмотреть наличие введённого ФИО в колонке Name базы данных. При наличии такого ФИО пользователь переходит на следующий этап – проверка табельного номера, введённое им ФИО сохраняется в отдельную переменную, и в словаре по ключу его id в Telegram присваивается значения 2, при отсутствии данного ФИО в списке пользователь остаётся на шаге проверки ФИО. Для проверки табельного номера нам понадобятся функции модуля Pandas для фильтрации таблицы. После успешного прохождения про-

верки ФИО пользователь вводит свой табельный номер, теперь программа обращается к таблице с запросом отобразить такие строчки, ФИО которых соответствует введённому пользователем ФИО, поскольку все значения в колонке Name уникальны, то строчка будет одна. После сортировки данных программа сравнивает табельный номер, введённый пользователем, с тем, который содержится в строчке после фильтрации. При несоответствии номера в словаре шаг идентификации обновляется на 1, что возвращает пользователя на проверку ФИО, а при соответствии шаг обновляется на 3, что говорит о том, что данный пользователь прошёл полностью процесс идентификации и теперь ему доступна работа с ботом. После идентификации, как уже было упомянуто, пользователь имеет выбор из кнопок. Рассмотрим действие просмотра актуальной информации по АЗС. При нажатии на эту кнопку пользователю выводится выбор кнопок с номерами АЗС, а также условный план местности, чтобы пользователю было удобнее ориентироваться. При нажатии на кнопку с номером АЗС боту приходит соответствующее сообщение, которое бот использует как ключ для обращения к словарю, содержащему информацию про АЗС. Бот отправляет пользователю то значение, которое находится в словаре по соответствующему ключу: очередь и среднее время заправки, умноженное на очередь в минутах, для каждой колонки на АЗС. При нажатии на кнопку о нахождении в очереди на АЗС перед пользователем так же появляется выбор из номеров АЗС, и он выбирает ту, на которой находится. Пользователь нажимает на номер АЗС, а в словаре пользователей значению по ключу id пользователя присваивается номер АЗС. После этого пользователю выводятся кнопки с колонками на АЗС, и пользователь выбирает ту, на которой он находится. В этот момент значение, соответствующее данной колонке, увеличивается, а в словаре пользователей к значению прибавляется указатель колонки. Также предусмотрено, что бот не позволяет записаться на другую колонку, если уже есть запись. Для этого он проверяет значение в словаре пользователей и смотрит, есть ли там указатель колонки на АЗС. Теперь опишем работу кнопки об окончании заправки. При нажатии на эту кнопку проверяется сначала то, выбрана ли у пользователя колонка или АЗС. Если в соответствующем значении в словаре пользователей есть только номер АЗС или нет и его, то это значит, что кнопка была нажата ошибочно, и дальнейших действий производить не нужно. Если же пользователь действительно запрашивается, то тогда значение очереди на соответствующей колонке соответствующей АЗС обновляется, а значение в словаре пользователей обнуляется. Теперь пользователь может снова произвести запись на АЗС. Важно, что макет работает на вымышленных данных, которые заменяют реальные данные, с которыми приходится работать в реальных условиях, потому была составлена также условный план местности, который выводится при просмотре информации об АЗС.

Заключение и выводы.

В заключение стоит ответить, что созданный прототип бота вполне соответствует ожиданиям заказчика, однако для более эффективной работы бота

также планируется добавить возможность телеграмм-боту отслеживать местоположение пользователя с помощью геолокации, чтобы сделать работу с ботом более удобной за счёт сообщения водителю ближайшей для него АЗС и корректировкой данных об очереди. Макет бота работает по технологии Long Polling и когда запущен его код в PyCharm, однако для работы в реальных условиях необходим Webhook и постоянная работа бота.

## СПИСОК ЛИТЕРАТУРЫ

1. Хабр [Электронный ресурс]: новостной веб-сайт и журнал - . – режим доступа к сайту.: <https://habr.com/ru/all/> - . – «Простой бот на Python за 30 минут» - . – режим доступа к статье.: <https://habr.com/ru/post/442800/>
2. mos.ru [Электронный ресурс]: новостной веб-сайт - . – режим доступа к сайту.: <https://www.mos.ru> - . – «Первый городской телеграм-бот поможет записаться на приём к врачу» - . – режим доступа к новости.: <https://www.mos.ru/news/item/28021073/>
3. Хабр [Электронный ресурс]: новостной веб-сайт и журнал - . – режим доступа к сайту.: <https://habr.com/ru/all/> - . – «Создание кнопок для телеграмм бота с использованием библиотеки pyTelegramBotAPI» - . – режим доступа к статье.: <https://habr.com/ru/sandbox/163347/>

### КОД ПРОГРАММЫ И СКРИНШОТЫ РАБОТЫ БОТА

Ссылка на google-диск:  
[https://drive.google.com/drive/folders/148mR2Z2bf0xZ\\_YHFTiSI7pMl2Fi9aIjq?usp=sharing](https://drive.google.com/drive/folders/148mR2Z2bf0xZ_YHFTiSI7pMl2Fi9aIjq?usp=sharing)

## БОТ ДЛЯ ОЦЕНКИ И ПОИСКА ИГР

**Байдали Михаил Сергеевич**

*МАОУ Школа «Перспектива», 8 класс, г. Томск*

Руководитель: Осинцев Артем Викторович, п.д.о. АНО ДО Технопарк "Кванториум".

Аннотация. Работа бота состоит в оценивании игры пользователем, также функционал бота позволяет посмотреть топ популярных игр на данный момент.

Ключевые слова: игра, бот, оценка, комментарий, топ.

Введение. Актуальность проблемы подтверждается тем, что некоторые люди пройдя игру [3], хотят внести свой вклад в оценку игры и в данном боте они могут это осуществить и оставить оценку для понравившейся им игре, и также если кто-то ищет игру чтобы пройти её и не может понять по мнению других людей интересна ли она.

Обзор аналогов. Существуют следующие способы оставить отзыв на игру, например, на том приложении где вы её скачали, но многим людям будет намного интереснее посмотреть отзывы реальных людей, а не накрученных отзывов, которые просто позволяют автору идеи продвинуть свою игру в топ

из игр и заработать на этой игре побольше денег, а здесь мне не нужно накручивать хорошие отзывы на ту или иную игру, ведь мне без разницы будет ли эта игра в топе.

Основная часть. В ходе написания данного проекта был создан бот для оценки игры. Целевой аудиторией бота являются люди, которые играют в компьютерные игры. Новизной данного проекта является тот факт, что ранее в ботах не было реализовано функционала по предоставлению статистики и возможности оценки лучших игр, входящих в список лучших. Разработан чат бот на языке python для оценивания и формирования лучших игр по оценкам пользователей. Для разработки бота использовался модуль на python telebot [1].

Ход работы бота:

1. Пользователь обращается к боту.
2. Как только пользователь заходит в бота, бот проверяет наличие новых игр и если они есть то добавляет их в свою базу (изначально в базе все топ игры на ПК начиная с 2010 по н. в.).
3. Чат бот выдаёт вопрос: "На какую игру вы бы хотели оставить отзыв?:
4. Далее пользователь вводит название игры и её жанр.
5. Бот в связи с жанром выдаёт пользователю список оценивания. В жанре «игра-искусство» геймплей должен занимать 10%, графическая составляющая 30%, сценарий 25%, режиссура 20%, звук и озвучивание 15%.;
6. Игры-Аттракционы Геймплей занимает доминирующее положение в 50%, аддиктивность в 15%, стилизованность в 20% и реиграбельность в 15%.
7. Для примера: Игровой процесс (Предложенные механики и удобство пользования): 9 из 10 ( $9 \cdot 50 / 10$ ) = 45
8. Аддиктивность: 7 из 10 ( $7 \cdot 15 / 10$ ) = 10,5
9. Стилизованность: 10 из 10 ( $10 \cdot 20 / 10$ ) = 20
10. Реиграбельность: 6 из 10 ( $6 \cdot 15 / 10$ ) = 9
11. Таким образом игра набирает 84,5 балла из 100 после всех манипуляций бот выдаёт итоги его опроса средняя оценка всех элементов (по совместительству это является и итоговой оценкой игры). В конце бот выдаёт общий итог всех опрошенных ранее теперь с учётом оценок данного пользователя.

В конце бот выдаёт общий итог всех опрошенных ранее теперь с учётом оценок данного пользователя. Также есть функция где, пользователь вводит слово топ и вводит период времени и бот выдаёт ему например за последний месяц в топе такие то такие то игры.

Выводы и дальнейшие перспективы проекта. В дальнейшем проект планируется развивать и добавить такие функции как поиск игр по жанрам, по аналогам, оставлять ссылки, на игры, которые понравились и так далее, и это приведет к масштабному боту по оцениванию, поиску и скачиванию понравившихся пользователю игр. Также планируется опубликовать этого бота в Телеграмме.

## СПИСОК ЛИТЕРАТУРЫ

1. TeleBot [Электронный ресурс]. 2023. Дата обновления: 09.02.2023. URL: <https://github.com/xditya/TeleBot> (дата обращения: 09.02.2023).
2. BotFather [Электронный ресурс]. 2023. Дата обновления: 05.01.2023. URL: <https://telegram.me/BotFather> (дата обращения: 05.01.2023).
3. Steam [Электронный ресурс]. 2023. Дата обновления: 23.02.2023. URL: <https://store.steampowered.com/?l=russian> (дата обращения: 23.02.2023).

## ВЛИЯНИЕ ДИЗАЙНА УПАКОВКИ НА НАШЕ ВОСПРИЯТИЕ

*Байдуров Матвей*

*МАОУ Гимназия № 26*

*г. Томска.*

Руководитель: Реутова Надежда Александровна

Аннотация. Данная работа заключается в том, чтобы понять на какие критерии смотрит покупатель при выборе продукта. Результаты этой работы помогут людям в дальнейшем, при создании упаковок продуктов питания.

Актуальность. Эта работа будет всегда актуальна, т.к. люди почти каждый день будут покупать продукты питания, из-за того, что это жизненная необходимость.

Для будущего создания упаковки, был проведен социологический опрос в виде дистанционного анкетирования, для того чтобы определить, какие критерии влияют на наше восприятие дизайна продуктов. Основными критериями для опроса был вкус, шрифт, цвет, материал, композиция на упаковке. По результатам опроса был сделан вывод.

Результаты:

Красный цвет: Подойдет, если содержимое упаковки острое (80%) или сладкое (15%). Не очень хорошо подойдет к натуральной продукции (22,5%). У большинства красный цвет ассоциируется с мясной продукцией (92,5%) и овощной продукцией (40%), так же с определенными напитками (например, такими как кока-кола, 25%). Так же ассоциируется с итальянской, русской, китайской, корейской кухнями.

Оранжевый цвет: Подойдет, если содержимое упаковки острое (37,5%), кислое (27,5%) или сладкое (25%). Не очень хорошо подойдет к натуральной продукции (22,5%). У большинства оранжевый ассоциируется с фруктовой и овощной (30%), готовой продукцией (35%), с выпечкой (55%), крупами и макаронами (37,5%). Частично подойдет для напитков (25%). Ассоциируется с мексиканской и корейской кухней.

Желтый цвет: Подойдет, если содержимое продукции кислое (62,5%). Подойдет для фруктовой (32,5%), кондитерской продукции (27,5%), круп и макарон (75%), выпечки (70%), готовой продукции (47,5%), полуфабрикат (15%). Так же подойдет для определенных напитков (25%). Подойдет для немецкой, казахской и индийской кухни.

**Зеленый цвет:** Подойдет, если содержимое упаковки кислое (60%) или соленое (25%). Отлично подойдет для ЭКО-продукции (77,5%), фитнес-, овощной и фруктовой (65%) продукции. Частично для напитков (тархун, 25%) и готовой продукции (17,5%). Ассоциируется с итальянской, шведской кухней.

**Голубой/синий цвет:** Подойдет, если содержимое продукции соленое (50%) или сладкое (25%). Не подойдет если продукция натуральная (50%). Подойдет к молочной продукции (30%), морепродуктам (75%), полуфабрикатам (15%) и напиткам (60%). Подойдет для мужской продукции. С кухней мира не ассоциируется.

**Фиолетовый цвет:** Подойдет, если содержимое упаковки сладкое (66,7%). Не подойдет к натуральной продукции (60%). Подойдет напиткам (частично, 17,5%). С кухней мира не ассоциируется.

**Розовый цвет:** Подойдет, если содержимое упаковки сладкое (84,6%). Не подойдет для натуральной продукции (25%). Ассоциируется с кондитерской продукцией (65%) и морепродуктами (75%). Отлично подойдет для женской продукции. С кухней мира не ассоциируется.

**Черный цвет:** Подойдет, если содержимое продукции горьковатое (55,5%). Подойдет к полуфабрикатам (25%) и мужской продукции. Ассоциируется с корейской кухней. Лучше его не использовать на упаковках, так как содержимое кажется горьким.

**Шрифт упаковки:** Шрифт упаковки не оказывает сильное влияние на восприятие упаковки, так как не ассоциируется со вкусами и продуктами, которые находятся внутри (60%).

**Круглая форма упаковки:**

У большинства, круг и круглая упаковка ассоциируется с пиццей (25%) и пончиками (25%). Некоторые выделяют, что круглая форма ассоциируется соленая и острая пища (7%).

**Квадратная форма упаковки:** У некоторых, квадрат и квадратная упаковки ассоциируется с чем-то сладким (5%). У большинства квадратная форма ассоциируется с соком (10%) и тортом (27,5%).

**Треугольная форма упаковки:** У большинства, треугольник и треугольная форма упаковки ассоциируется с сыром (7,5%), онигири (20%) и кисломолочной продукцией (27,5%).

**Материал упаковки: Бумага:** подойдет для эко-продукции (75%), выпечки (22,5%), кондитерской продукции (17,5%)

**Картон:** подойдет для эко-продукции (77,5%), молочной (10%) и кондитерской продукции (55,5%)

**Дерево:** подойдет для эко-продукции (40%), для фруктовой/овощной продукции (20%),

**Стекло:** подойдет для молочной продукции (62,5%), для напитков (77,5%).

Полиэтиленовая упаковка: подойдет для мясной продукции (37,5%), для овощной и фруктовой продукции (52,5%), для макарон и круп (60%), выпечки (45%). Ассоциируется с вредной продукцией (82,5%).

Смешанная упаковка: подойдет для мясной (40%), кондитерской продукции (42,5%), для фруктов и овощей (52,5%), морепродуктов (27,5%), полуфабрикатов (50%) и готовой продукции (50%).

Крафтовая бумага: подойдет для эко-продукции (40%), для фруктов и овощей (12,5%), для круп и макарон (20%), для выпечки (45%).

Предназначение:

Фитнес продукция: яркая и красочная (чаще в зеленых тонах) на картинках присутствует тело, изготовлена из эко-материалов, присутствует рукописный текст (41,5%).

Детская продукция: веселая упаковка, присутствуют яркие цвета, чаще изображены картинки и персонажи. Упаковка меньше и присутствует какая-то игрушка (40%).

Мужская продукция: упаковка строгая, в темных тонах, текст печатный (40%).

Женская продукция: элегантная и изящная упаковка, в розовых тонах, текст рукописный (45%).

Упаковка для пожилых: обычная, удобная и без излишеств. Присутствует доля уюта. Чаще всего продукт полезный и недорогой. В фиолетовых тонах (47,5%).

Композиция: Динамика: вызывает беспокойство, ненадежность и недоверие. Изображает активность (20%).

Статика: вызывает чувство надежности и доверия. Ассоциируется со строгостью, стабильностью и солидностью. Немного скучная (25%).

Отличие ответов мужчин от ответов женщин:

Красный у мужчин ассоциируется только с острым, а у женщин так же с соленым и сладким.

Зеленый у мужчин ассоциируется с кислым сладким и соленым, у женщин только кислый. В других областях различия не наблюдаются.

Отличие ответов представителей разных возрастов. В основном отличий нет, но замечено, что люди после 40, ассоциировали треугольную упаковку с кисломолочной продукцией.

## **РАЗРАБОТКА ОСЦИЛЛОГРАФА НА БАЗЕ МИКРОКОНТРОЛЛЕРА ARDUINO UNO**

***Бердникова Маргарита, Козлова Карина***

*Муниципальное бюджетное общеобразовательное учреждение лицей при*

*ТПУ, 10 класс*

*г. Томск*

Руководитель: Чистякова Надежда Владимировна, канд. физ-мат. наук

Для наблюдения и записи частотных и амплитудных характеристик электрического сигнала используют осциллограф – прибор, принимающий на вход и графически отображающий данные характеристики[1]. Область применения осциллографов велика: их используют в лабораториях или в учебных заведениях для проведения различных научных исследований, на предприятиях электронной промышленности, в энергетике и электротехнике[2]. Настоящий лабораторный осциллограф – сложное и дорогостоящее оборудование, однако в настоящее время существует возможность сделать достаточно функциональный и недорогой прибор с помощью микросхемы Arduino. Arduino произвела настоящую революцию в электронике, позволив людям, ранее не имевшим опыта работы с микроконтроллерами, создавать сложные устройства.

Цель проекта. Создание осциллографа на базе микроконтроллера Arduino UNO.

Задачи проекта:

- Подобрать компоненты, необходимые для создания осциллографа.
- Найти нужные для подключения экрана и работы с ним библиотеки и написать программу.
- Собрать устройство и провести тестовые измерения.

По данным интернет-ресурсов, аналогом в ценовой категории нашего проекта является цифровой осциллограф DSO138, который можно приобрести за 2200 рублей. Себестоимость проекта мы оцениваем в 1800 рублей.

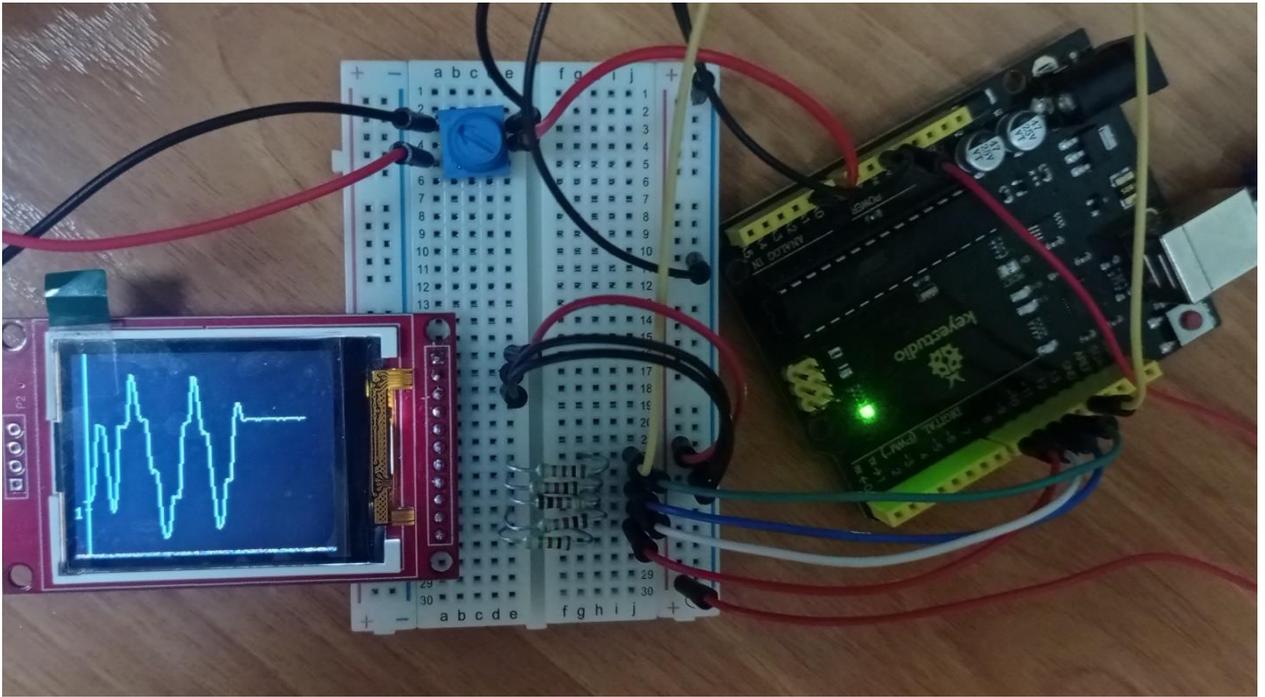
Этапы выполнения проекта

Для сборки осциллографа понадобились: Arduino UNO, TFT -Дисплей SPI ST7735, проводники, макетная плата, резисторы.

При подключении дисплея стоит учесть, что он может выдержать напряжение до 3.3 Вольт, следовательно, необходимо подключать его через резисторы, в нашем случае напряжение на них было 1кОм.

Для того, чтобы Arduino могла работать с экраном, необходимо подобрать библиотеки. Были выбраны библиотеки Adafruit\_GFX, отвечающая за графические процессы в дисплее, Adafruit\_ST7735, отвечающая за работу непосредственно с контроллером дисплея, и SPI, отвечающая за обмен данными по SPI-интерфейсу, т.е. за подключение дисплея к Arduino. Для проверки функционирования дисплея на экран был выведен текст.

Следующим шагом было создание программы для контроллера, обрабатывающей входной сигнал и выводящей информацию о нем на экран. Так как пины Arduino не могут принимать напряжение, превышающее 5 Вольт, первоначально в качестве источника напряжения была взята батарейка напряжением 4.5 Вольт, подключенная к пину A1 через потенциометр. Таким образом, можно было вручную изменять напряжение и проверять работу программы (Рис.1).



*Рисунок. 1 Проверка работы программы*

Часть кода, отвечающая за отрисовку графика, представлена на рисунке 2. В тот момент она была рассчитана на то, что напряжение на батарее не превышает 4.5 Вольт. Так как аналоговые входы на Arduino оснащены 10-разрядным аналого-цифровым преобразователем[3], то максимальное значение, которое может выдать вход A1, будет равно 1023. Чтобы перевести полученное значение в вольты, необходимо это значение умножить на коэффициент, считающийся по формуле  $U/1023$ , где  $U$ - максимальное напряжение на источнике, в нашем случае 4.5. Следовательно, коэффициент будет равен 0.0044. Для того, чтобы вывести его на экран, нужно учесть, что высота экрана, используемая для вывода графика, составляет 124 пикселя, то есть придется рассчитать соотношение показания пина A1(полученное значение мы не будем переводить в вольты) и координаты на экране. Для этого можно воспользоваться пропорцией:  $1023/a = 124/y$ , где  $a$ - показания пина A1,  $y$ - координата по  $y$  данному показанию на графике в пикселях. Следовательно,  $y$  можно вычислить по формуле  $a*0.121$ , где  $0.121=124/1023$ .

```
int x1=round(analogRead(A1)*0.121);
int x2=round(analogRead(A1)*0.121);
for (int16_t y1=6; y1 < tft.height()-5; y1+=1) {
    tft.drawLine(x1, y1, x2, y1+1,ST77XX_CYAN);
    x1=x2;
    x2=round(analogRead(A1)*0.121);
    delay(100);
}
```

*Рисунок.2. Часть кода, рисующая график*

Проект ориентирован на образовательные учреждения. Планируется, что данное устройство будет использоваться при проведении лабораторных работ, так как оно позволяет визуализировать сигналы в электрических цепях.

Выводы и перспективы.

Перспективой развития данного проекта является дальнейшее расширение функциональных возможностей устройства, увеличение удобства использования и точности измерений.

#### СПИСОК ЛИТЕРАТУРЫ

1. Виды и типы осциллографов их назначение и применение. – URL: <https://inkotel.com.ua/vidy-i-tipy-ostsillografov-ih-naznachenie-i-primenenie> (дата обращения: 07.03.2023)
2. Осциллограф. Виды и устройство. Работа и применение. Особенности. – URL: <https://tehpribory.ru/glavnaia/pribory/ostsillograf.html> (дата обращения: 07.03.2023)
3. Аналогово-цифровые преобразования – АЦП — RobotClass. –URL: [https://robotclass.ru/tutorials/arduino\\_adc/](https://robotclass.ru/tutorials/arduino_adc/) (дата обращения: 09.03.2023)

#### АНИМАЛИСТИЧЕСКИЙ ЖАНР В СОВРЕМЕННОМ ИСКУССТВЕ

*Благова Д. С.*

В теоретической части проекта раскрывается суть и актуальность работы графического художника. Особое внимание уделяется анималистическому жанру в современном искусстве. Мы выяснили, что анималистика — это жанр изобразительного искусства, главным мотивом и основным объектом которого являются животные, главным образом в живописи, фотографии, скульптуре, графике и реже в декоративном искусстве. Изображения животных очень часто используются дизайнерами при создании логотипов брендов. Вторая глава посвящена принципам стилизации (превращение объемной формы в плоскостную и упрощение конструкции, обобщение формы с изменением очертания, обобщение формы в ее границах, обобщение и усложнение формы, добавление деталей, отсутствующих в натуре) в анималистике. Подробно описаны этапы и приёмы стилизации .

Практическая часть проекта включает в себя этапы работы в анималистическом жанре «Стилизация животного с применением компьютерных технологий», выполненная в компьютерной программе Krita - бесплатная программа для 2D-рисования. Продуктом проекта стал готовый рисунок, выполненный в вышеуказанной программе. Практическая значимость проекта обусловлена возможностью использования пошагового выполнения подобного или иного рисунка в технике графический дизайн.

Методы исследования, используемые в проекте: эмпирический (наблюдение, сравнение), теоретический (изучение литературы) и экспериментальный (создание рисунка).

Выводы:

1. Раскрыв понятия анималистика и стилизация, приведя примеры стилизации, которыми пользуются другие художники, мне удалось более подробно познакомиться с данной темой и создать собственный алгоритм работы в данном направлении.
2. Подробно описан процесс создания объекта анималистического жанра в графическом редакторе. У меня получилась прекрасное изображение в формате jpg, которое можно использовать как самостоятельный рисунок, для печати принта в текстильной промышленности, полиграфии, логотипе и других направлениях дизайна.

#### СПИСОК ЛИТЕРАТУРЫ

1. Рисование. Сом. <https://risovaniye.com/krita.html>
2. Анималистический жанр в изобразительном искусстве <https://www.izocenter.ru/blog/animalisticheskij-zhanr-v-izobrazitelnom-iskusstve/>
3. Анималистический жанр. От древних времен до XXI века <https://www.arts-dnevnik.ru/animalisticheskij-zhanr/>
4. Галеон <http://www.ba-get1.ru/applied-arts/stylization-applied-arts.php>

#### АНАЛИЗ КЛАВИАТУРНОЙ ДИНАМИКИ

*Бочкарев Егор*

*Муниципальное бюджетное общеобразовательное учреждение лицей при  
ТГУ, 10 класс  
г. Томска*

Руководитель: Кочегурова Елена Алексеевна, канд. техн. наук

Последние годы сетевые и цифровые технологии активно развиваются, что способствует расширению различных возможностей к доступу и хранению конфиденциальной информации на различных цифровых устройствах. Именно поэтому защита информации, предоставляемая пользователями от несанкционированного доступа, становится все более и более востребованной.

Кроме уже существующих мер защиты информации (технических и организационных) в настоящее время разрабатываются и широко используются средства для авторизации и идентификации (аутентификации) личности пользователя.

Проведенными исследованиями было установлено, что особенности пользования клавиатурой, а именно динамика нажатия клавиш – клавиатурный почерк (КП) может использоваться как средство аутентификации личности [1]. Как поведенческая характеристика, КП по своей природе являются динамической. Обычно он формируется через 6 месяцев работы с компьютером. Поведенческие характеристики КП в отличие от физиологических сложнее распознать с высокой точностью, но вместе с тем, сложнее и подделать [2].

В настоящее время разработано несколько классификаций для анализа клавиатурного почерка пользователей. Одной из наиболее часто используемой

является классификация по типу создаваемого текста. Согласно ей идентификация подразделяется:

1. *Статическая аутентификация* [1]. Проверка пользователя осуществляется во время первичной аутентификации на основе структурированного/предопределенного текста. Предопределенный текст обеспечивает более надежную проверку пользователя, чем только ID/пароль при первичной аутентификации. Также предопределенный текст может дополнять проверку ID/пароль в случае возникающих подозрений.
2. *Динамическая или непрерывная аутентификация* [1]. Непрерывная проверка осуществляется на протяжении всего сеанса работы пользователя на основе произвольного текста, вводимого им в любом программном приложении. Такой скрытый мониторинг обеспечивает дополнительную меру безопасности после авторизации пользователя в системе. Свободный текст в большей степени, чем предопределенный, напоминает реальную среду работы пользователя и позволяет выявить поведенческие характеристики.

Целью работы было проведение анализа клавиатурной динамики для идентификации личности.

Задачи:

1. Изучить КП пользователей по ВУК.
2. Изучить КП по скорости набора пользователей.
3. Оценить качество распознавания легитимности пользователей.

Полученные результаты и обсуждения: Тестирование проводилось на основе локального набора данных, собранных в домене университета [1] и общедоступного набора клавиатурного мониторинга (КМ) [3] с использованием Евклидовой, Манхэттенской метрики и Махаланобиса. Мы использовали набор данных, собранный на основе свободных русских текстов. Набор содержит шаблоны порядка 6 пользователей. Однако в локальном наборе имеются данные по нескольким сессиям (7-21), а в международном КМ – только одна сессия.

В связи с этим было принято решение использовать моделирование сессий на основе преобразования Бокса-Мюллера. В результате было получено заданное количество сессий с шаблонами пользователей, как выборки из одной генеральной совокупности. И на рисунке 1 визуализированы шаблоны шести пользователей датасет КМ, подписанные справа сверху на данном рисунке. По оси ординат - ВУК (Время удержания клавиатуры) для букв русского алфавита.

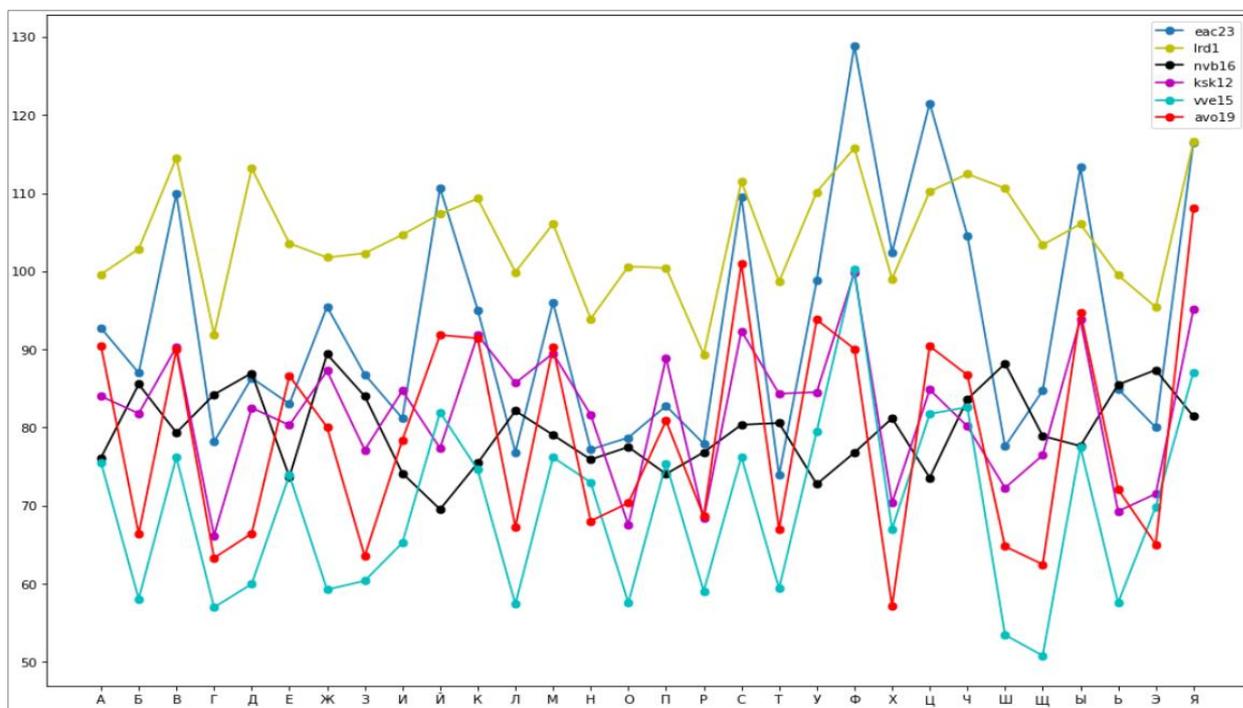


Рисунок 1. Графическое отображение ВУК всех пользователей

По данному рисунку видно, что каждый клавиатурный шаблон уникален по скорости, ритму печати на клавиатуре и некоторыми методами работы на клавиатуре, а именно использование служебных клавиш: Enter, Esc (Escape), Caps Lock. Все эти факторы и определяют и сам КП, и возможность различать пользователя на его основе. К примеру, пользователь lrd1 тратит больше времени на удержание клавиш чем пользователь vve15, что говорит нам о том, что пользователь vve15 более опытен, чем пользователь lrd1.

Статистические характеристики показателей КП разных пользователей также различны. Рисунок 2 иллюстрирует плотности распределения ВУК этих же шести пользователей.

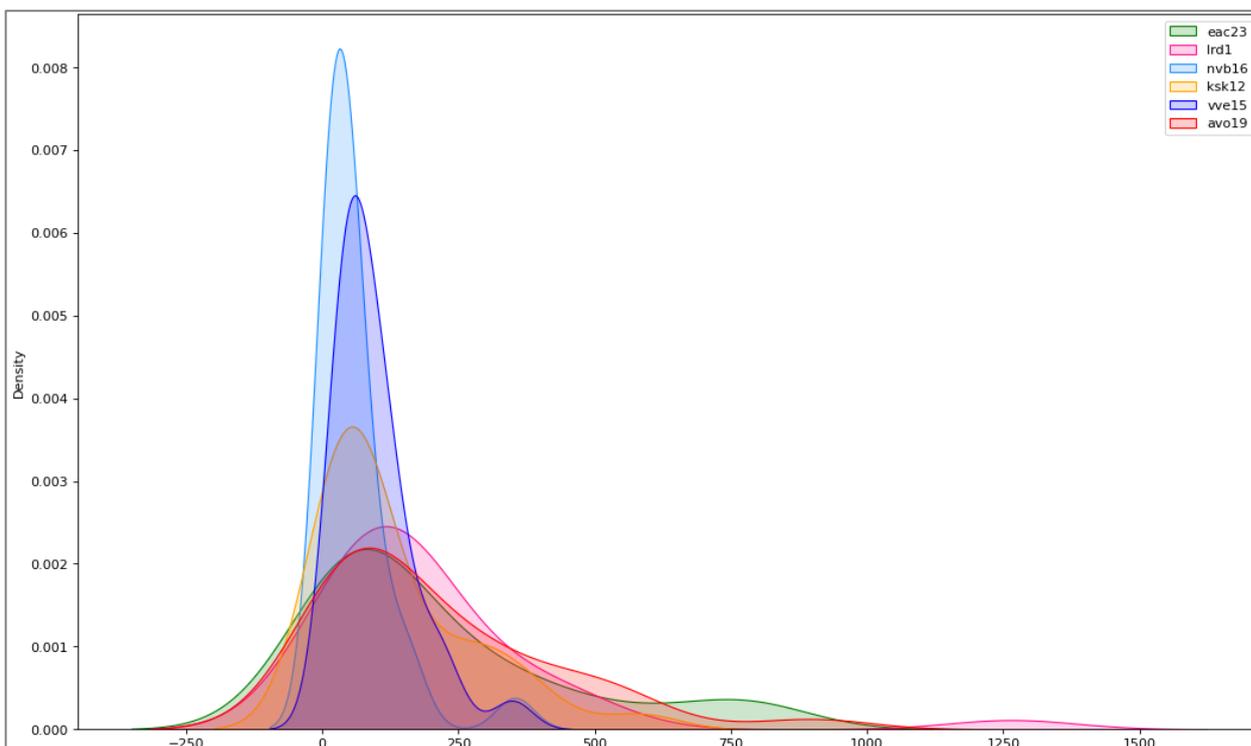


Рисунок 2. Плотность распределения ВУК

Анализ данной гистограммы подтверждает высокую скорость набора пользователей nvb16 и vve15 (голубой и фиолетовой цвет). Но у пользователя nvb16 наиболее развиты навыки работы с клавиатурой, точно так же и разброс для большинства букв. Самый медленный тем набора у пользователя eac23 (зеленый цвет). Эта гистограмма имеет к тому же протяженный характер, что соответствует большим скачкам параметра ВУК для разных букв. И это характеризует слабые навыки набора текстов.

На этапе распознавания проводится сравнение текущего шаблона пользователя за компьютером с его шаблоном из базы данных. В результате легитимность пользователя подтверждается, либо отвергается. И в дистанционной образовательной системе это соответствует прохождению теста, либо блокировке пользователя.

Для оценки качества распознавания легитимности пользователей были использованы ошибки I и II рода. В клавиатурной динамике эти показатели носят название FRR (частота ложного отказа законному пользователю) и FAR (частота ложного допуска незаконных пользователей).

Показатели FRR и FAR заметно зависят от порогового значения, допускающего пользователя в систему. И графически эти показатели представлены на рисунке 3 для Манхэттенской метрики в диапазоне порога от 0,01 до 700 мс.

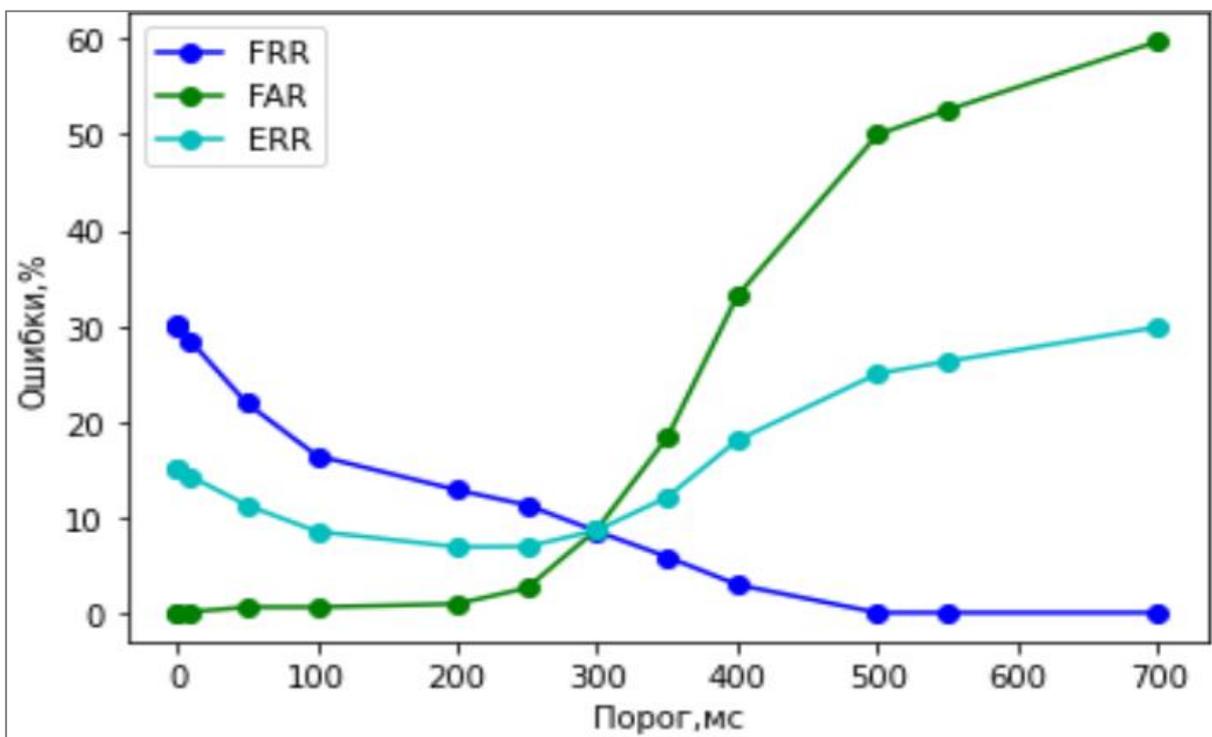


Рисунок 3. Показатели эффективности распознавания

Показатели эффективности позволяют оценить возможности системы распознавания. Наиболее наглядно эти возможности демонстрирует ошибка  $ERR = (FRR + FAR) / 2$ , называемая равной частотой ошибок. И для данного способа распознавания пользователя  $ERR = 10\%$  при среднем пороге 300 мс.

Выводы проекта:

1. При изучении КП пользователей на основе ВУК было выявлено, что КП уникально по скорости и ритму нажатия клавиш.
2. У пользователей разные навыки пользования клавиатурой, разное количество используемых клавиш.
3. Легитимность качества распознавания пользователей составляет 90%

#### СПИСОК ЛИТЕРАТУРЫ

1. Кочегурова Е.А., Мартынова Ю.А. Особенности непрерывной идентификации пользователей на основе свободных текстов в режиме скрытого мониторинга// Программирование. – 2020. -№ 1. -С. 15–28.
2. Иванов А.И. Биометрическая идентификация личности по динамике подсознательных движений. - Пенза: ПГУ. 2000. -187 с.
3. González N, Calot EP, Ierache JS, Hasperué W. On the shape of timings distributions in free-text keystroke dynamics profiles. Heliyon 2021; 7(11). DOI: 10.1016/j.heliyon. 2021.e08413.

#### СОЗДАНИЕ КОНТРОЛИРУЮЩЕЙ ПРОГРАММЫ ПО МАТЕМАТИКЕ ДЛЯ НАЧАЛЬНОЙ ШКОЛЫ

***Васильев Евгений, Абдрахманова Адель***

*Коммунальное государственное учреждение «Средняя многопрофильная школа №37» отдела образования по городу Усть-Каменогорску управления образования ВКО, Казахстан*

Руководитель: Клименко Елена Сергеевна, учитель информатики, категория «педагог-исследователь»

Программирование богато и многообразно. Ведь, кажется, нет такой сферы человеческой деятельности, где нельзя было бы с пользой применить вычислительную машину для оценки, информационно-справочного обслуживания, планирования, моделирования и т. д. И это многообразие задач переходит в многообразие программ, которые должны разрабатывать программисты. Компьютеры интенсивно вторгаются в нашу жизнь. Каждый современный образованный человек должен иметь хотя бы элементарные знания, умения и навыки работы на ПК. Использование компьютера на уроках в начальных классах — это один из путей как можно сделать урок более интересным для школьников, помочь учителю. Компьютер — оперативное средство наглядности в обучении, помощник в организации проведения опроса и контроля школьников.

Цель нашего проекта: создать контролирующую программу по математике для учащихся начальной школы.

Задачи исследования: рассмотреть возможности визуальной среды программирования; подобрать математические задания для реализации программы; создать контролирующую программу по математике; провести апробацию программы.

Объект исследования – возможности визуальной среды программирования. Предмет исследования – использование компьютерных программ на уроках для проверки знаний учащихся.

Гипотеза - если использовать компьютерные программы на уроках в школе, то повышается интерес к предмету у учеников, облегчается работа учителя.

Оригинальные контролирующие программы можно разработать по математике, так как возможно запрограммировать автоматическую проверку результатов. Созданная нами программа проверяет знания арифметических действий учащихся начальных классов сложение и вычитание.

Наш выбор остановился на визуальной системе программирования Lazarus. Почему Lazarus? Он стал первой системой доступной совершенно бесплатно. Более того, является проектом с открытым исходным кодом. Многие программисты по всему миру принимают участие в его развитии. Lazarus имеет поддержку множества языков, в том числе и русского, что выгодно отличает его от других.

Как любая подобная система, Lazarus предназначена для разработки программ и имеет две характерные особенности: создаваемые программы могут работать не только под управлением Windows, а сама она относиться к

классу инструментальных средств ускоренной разработке программ. В данной визуальной среде можно создавать проекты любого типа, в том числе и контролирующие программы, так как Lazarus содержит набор компонентов, функций, процедур, позволяющих запрограммировать задания, отследить правильность их выполнения, вывести результат на экран.

Созданная нами программа состоит из двух форм титульная (рис 1) и рабочая (рис 2).



Рисунок 1. Титульная форма

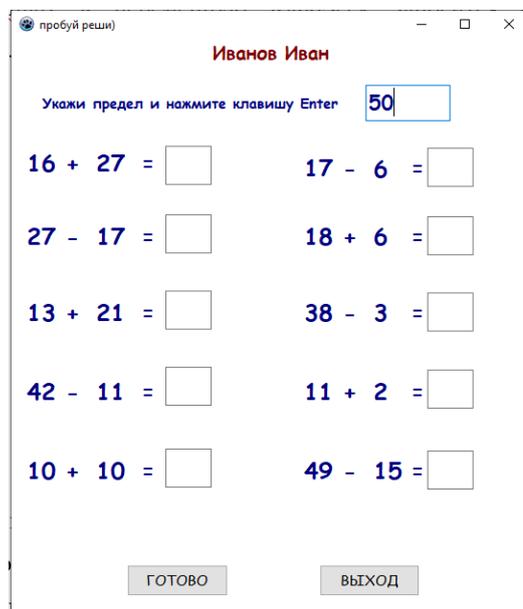


Рисунок 2. Рабочая форма

На титульной форме представляется название программы, расположено в базовом компоненте класса TLabel. Компоненты этого класса предназначены для размещения на форме различных рода текстовых надписей. Для этого служит центральное свойство компонента – Caption. С помощью свойства Font можно разнообразить вид надписи. Также представляется ввести свои данные: фамилию и имя.

Для перехода в рабочую форму используют кнопку «Начать» - класс TButton. Компоненты широко используются для управления программами. Алгоритм управления реализуется в обработке события OnClick.

При переходе на рабочую форму титульная форма скрывается, и появляются 10 примеров: 5 примеров на сложение, 5 – на вычитание. При каждом запуске программы вносится число, в пределах которого будут производиться вычисления. Числа для примеров задаются случайным образом от 0 до заданного числа, что облегчает учителю задачу разработки нескольких вариантов карточек. Для этого используется функция Randomize. Генератор псевдослучайных чисел представляет собой функцию, которая получает некоторое целое число, называемое базовым, изменяет его разряды по определенному алгоритму и выдает новое число как результат. Одновременно с этим новое число становится базовым при следующем обращении к функции и т.д. (Так как алгоритм процедуры не меняется в ходе ее работы, числа называются псев-

дослучайными). В системном модуле System, который автоматически доступен любой программе, базовое число хранится в переменной с именем RandSeek и всегда имеет начальное значение 0. Это означает, что при последовательном обращении к Random в различных программах (или при нескольких прогонах одной программы) будет всегда выдана одна и та же последовательность псевдослучайных чисел.

Для ввода учащимся ответов используется команда класса TEdit, компоненты которого представляют собой однострочное редактируемое текстовое поле. С его помощью можно вводить достаточно длинные текстовые строки.

Для получения результата программа сравнивает ответ ученика с правильными ответами и считывает количество совпадений. Выводится результат в виде количества правильных ответов.

Фрагмент кода программы второй формы:

```
...
begin
repeat
  Randomize;
  x1:=Random(n); y1:=Random(n);
  z1:=x1+y1;
  until z1<=n;
  Form2.Label1.Caption:=IntToStr(x1); Form2.Label3.Caption:=IntToStr(y1);
end;
...
Label48.Caption:='Количество правильных ответов '+IntToStr(k)+' из 10';
end;
procedure TForm2.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
  IF key=#13 then
    Edit2.SetFocus;
end;
...
```

С помощью объекта Image организована вставка значков «Зеленая галочка» и «Красная галочка» около правильных и неправильных ответов. С целью визуальной проверки результатов.

Фрагмент кода программы проверки правильности решения:

```
if z1=x1+y1 then image1.Picture.LoadFromFile('1.jpg') // зеленая галочка
else image1.Picture.LoadFromFile('2.jpg'); // красная галочка...
```

Добавлен компонент Таймер, который срабатывает на второй форме при активации предела вычислений и прекращает работу при нажатии на кнопку Готово. Выводится сообщение о количестве затраченных минут и секунд на решения.

Программа апробирована во 2 классе на уроке математики (рис 3).



Рисунок 3. Вывод результатов

Ученики были заинтересованы новым видом устного опроса. Учителю облегчилась задача при проведении этого опроса. Сократилось время учителя при проверке результатов, так как компьютер выдает результат автоматически. При использовании данной программы у учителя нет необходимости тратить время на составления карточек нескольких вариантов. Данную программу несложно подкорректировать для проверки знаний учеников в области других арифметических действий в любых числовых пределах.

Считаем, что гипотеза подтвердилась, использование компьютерных программ на уроках в школе повышает интерес к предмету у учеников, облегчает работу учителя.

Учителя нашей школы достаточно успешно используют программу на уроках математики. Программа может быть использована для проведения формативного оценивания по 10-бальной шкале.

## РАЗРАБОТКА ИГРОВОГО ПРИЛОЖЕНИЯ ДЛЯ ПОРТАТИВНОЙ КОНСОЛИ НА ОСНОВЕ RASPBERRY PI

*Галимзянов Камиль*

*Муниципальное бюджетное общеобразовательное учреждение  
лицей при ТПУ, 10 класс  
г. Томск*

Руководитель: доцент ОИТ ИШИТР ТПУ Мыцко Е.А.

Цель работы: Разработка собственного простого игрового приложения для портативной консоли, собранной на основе микрокомпьютера Raspberry pi 4 B.

Для достижения поставленной цели необходимо выполнить следующие задачи:

1. Провести обзор существующих операционных систем, эмуляторов, платформ для запуска игровых приложений на Raspberry pi;
2. Установить и настроить операционную систему на Raspberry pi;
3. Осуществить сборку портативного контроллера «Game NAT» и его настройку в операционной системе.
4. Определить тематику и платформу игрового приложения;
5. Провести обзор инструментов, средств, технологий и языков программирования для разработки игрового приложения;
6. Реализовать игровой дизайн, меню и элементы игры;
7. Разработать алгоритмы игрового процесса;
8. Реализовать алгоритмы игрового процесса с применением выбранных средств и технологий;
9. Осуществить интеграцию игрового приложения в соответствующую платформу.

### 1. Обзор существующих операционных систем, платформ для Raspberry PI

- Технические характеристики Raspberry Pi 4 B:
- Broadcom BCM2711 SoC;
- 64-битный четырехъядерный ARMv8 Cortex-A72 процессор с тактовой частотой 1.5 ГГц;
- Графический сопроцессор VideoCore VI®;
- Память на 1/2/4/8ГБ LPDDR4 SDRAM;
- Gigabit Ethernet;
- USB3.0;
- 2 x micro-HDMI;
- 4 ГГц и 5 ГГц IEEE 802.11.b/g/n/ac WI-FI + Bluetooth 5.0 Low Energy (BLE).

Raspberry Pi - это мини-компьютер, который может использоваться в качестве домашней медиа-студии, домашнего сервера, и многих других приложений, включая игры. Он работает на базе ARM-процессоров и обычно имеет небольшой объем оперативной памяти, поэтому выбор операционной системы и платформы для запуска игр важен.

Можно выбрать множество операционных систем в зависимости от задачи, которую необходимо решить. Они могут быть разделены на несколько категорий:

- Первая категория - операционные системы общего назначения, такие как Raspbian OS, Ubuntu, Apertis и другие. Они предназначены для выполнения различных задач и функций на компьютере.
- Вторая категория - операционные системы, специально разработанные для работы с медиа- и видеоресурсами, такие как LibreELEC, OSMC,

Volumio и другие. Они предназначены для управления и проигрывания мультимедийных контентов.

- Третья категория - операционные системы, предназначенные для работы с эмуляторами и играми, такие как Recalbox, RetroPie, Lakka и другие. Они созданы для того, чтобы пользователи могли наслаждаться играми на своих компьютерах.
- И, наконец, четвертая категория - это специальные операционные системы, созданные для выполнения конкретных задач, таких как 3D-печать, Anthias и другие. Они разработаны с учетом специфических потребностей и задач, и предоставляют соответствующие инструменты для их решения.

Мы выбрали Raspbian OS как идеальную операционную систему для Raspberry Pi, ведь она является ее официальным ПО и содержит все необходимые компоненты для выполнения основных задач. Кроме того, Raspbian OS обладает простым и приятным интерфейсом, что делает ее максимально удобной в использовании.

## 2. Установка и настройка операционной системы на Raspberry pi

Для установки операционной системы на Raspberry Pi нам необходимо сначала загрузить и установить Raspberry Pi Imager на компьютер, который имеет устройство чтения SD-карт.



Рисунок 1.

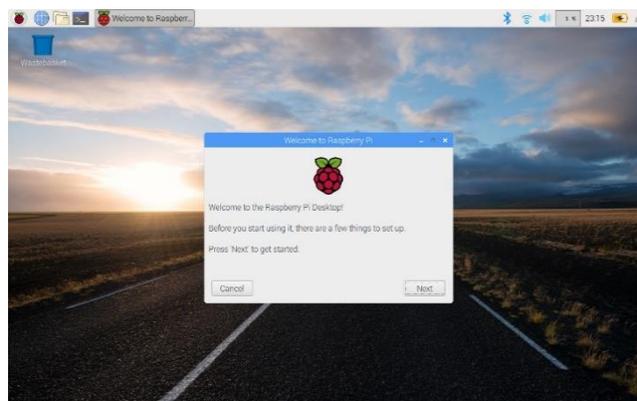


Рис.унок2.

После этого, вставляем SD-карту, которую мы будем использовать с Raspberry Pi, в устройство чтения и запускаем Raspberry Pi Imager, следуя инструкциям на экране. Как только файл изображения будет записан на SD-карту, вы готовы к работе. Вставляем карту в свой Raspberry Pi и загружаем систему, чтобы начать использовать его в соответствии с выбранной нами операционной системой.

## 3. Установка и настройка Контроллера GAME HAT

Game Hat - плата расширения, превращающая Raspberry Pi в портативную игровую консоль.



Рисунок 3.



Рисунок 4.

В разъем GPIO на плате подключается микрокомпьютер и переходник между экраном и контроллером, также в батарейный отсек помещается аккумулятор и сверху вся эта конструкция закрывается акриловым корпусом и крепится шурупами. (рис. 4)

Экран в этой конструкции подключается по HDMI кабелю, а кнопки и джойстик GPIO выводом.

Разработка игрового приложения и интеграция его в систему.

В результате работы было разработано несколько игр, одной из таких является “Flappy bird”.

Для создания данной игры был использован язык программирования Python и библиотека pygame, которая позволяет разрабатывать мультимедийные приложения и игры. Pygame основана на мультимедийной библиотеке SDL.

Первый шаг в создании игры - импорт модуля pygame и создание экрана для отображения игры. Далее создается основной цикл игры для открытия и закрытия приложения. Затем создается "хитбокс" для птицы и задаются правила ее полета.

Для отображения труб используются прямоугольники одинакового размера, и определяются последствия столкновения с трубами и краями экрана. Также загружаются заранее подготовленные текстуры, такие как задний план, модель птицы и трубы.

Для создания хаотического порядка появления труб в игре используется библиотека random. В игре также присутствуют очки и жизни птицы, которые зависят от набранных очков. Ускорение птицы также меняется в зависимости от количества очков.

Для создания атмосферы игры добавляется фоновая музыка и звуки столкновения с препятствиями.

#### Заключение

В заключение проекта по разработке игрового приложения для портативной консоли на основе Raspberry Pi, можно сказать, что работа была выполнена успешно. В ходе проекта были достигнуты все поставленные цели, а именно, разработано игровое приложение с интерфейсом управления на портативной консоли, основанной на Raspberry Pi.

## СПИСОК ЛИТЕРАТУРЫ

1. Raspberry Pi 4 Model B [Электронный ресурс] - Режим доступа: <https://micro-pi.ru/raspberry-pi-4-model-b-rpi-4-b-bcm2711/> (дата обращения: 11.1.2022)
2. Raspberry Pi OS [Электронный ресурс] - Режим доступа: <https://www.raspberrypi.com/software/> (дата обращения: 12.1.2022)
3. Г. Халфакри. Raspberry Pi. Официальное руководство для начинающих / пер. с англ. С. В. Черникова – М.: ДМК Пресс, 2021. – 262 с.: ил.
4. Создание игр на Python 3 и Pygame [Электронный ресурс] - Режим доступа: <https://habr.com/ru/post/347138/> (дата обращения: 12.1.2022)
5. Основы PyGame [Электронный ресурс] - Режим доступа: <https://python-course.readthedocs.io/projects/elementary/en/latest/lessons/18-pygame.html#id1> (дата обращения: 12.1.2022)

## 3D МОДЕЛЬ ДИЗАЙНА ЖИЛОЙ КОМНАТЫ

*Глебов Андрей*

*Муниципальное автономное общеобразовательное учреждение «Гимназия № 12», 8 класс г. Новосибирска.*

Руководитель: Липачук Наталья Владиславовна

Дизайн – это художественное конструирование предметов быта, внутренних помещений, промышленных, типографских и т. п. изделий.

Может сложиться вопрос: «Почему он так важен и так много используется?». Во-первых, он придаёт объекту различные эстетические свойства, которые, все вместе, создают определенные ощущения у человека, лицезревшего его. Профессия целью которой является создание дизайна называется дизайнером.

Дизайнеры используют программы 3D моделирования. Это позволяет потенциальному заказчику, ещё на начальном этапе, увидеть, как будет выглядеть дизайн. Например, заказчик может определить нравится ли ему дизайн или нет, а также добавить свои предпочтения.

Причиной выбора мной этого проекта в том, что моделируя дизайн я буду развивать свои навыки и смогу использовать их в других сферах применения 3D моделирования такие как: анимация, gamedev, 3d печать и т.п.

Цель: создать 3D модель функциональной и комфортной жилой комнаты.

Задачи:

1. Определить функциональные зоны, и какие объекты будут находиться в комнате и их расположение.
2. Смоделировать 3D-объекты и расположить их в комнате, используя для этого Blender.
3. Сделать визуализацию проекта.

#### 4. Подготовить доклад и презентацию проекта.

Есть множество инструментов 3D моделирования. Наиболее популярные из них: Maya, Autodesk 3ds Max, Blender. Autodesk Maya занимает первое место в рейтинге лучших программ для 3D-моделирования, составленном большинством художников.

3ds Max от Autodesk используется в основном при производстве телевизионных и художественных фильмов или для визуализации архитектуры и продукта. Как и его родственное программное обеспечение Maya, 3ds Max может похвастаться надежным набором инструментов для 3D-моделирования, а также моделирования жидкостей, волос и меха, настройки персонажей и анимации.

Перечисленные выше программы платные поэтому я не стал их использовать и выбрал Blender, за его многофункциональность, универсальность и доступность.

Я считаю, что функциональная комната должна содержать в себе 3 зоны: зону хранения в которой будут находиться шкафы, зону отдыха в которой будет находиться диван-кровать и ТВ зона, и рабочая зона с письменным столом и удобным креслом. Зона отдыха занимает центральную часть комнаты, рабочая зона располагается вблизи источника естественного света (окна), зона хранения максимально вместительная, но при этом визуально не перегружает комнату.

Для начала я создал основу для комнаты из плоских объектов (plane) в одной из них я сделал отверстие для двери и саму дверь, а на противоположной стене сделал окно с подоконником в этом мне сильно помог инструмент insert faces. К примеру, с помощью него я сделал раму для окна и стеклопакет. Также мне помог инструмент extrude который позволяет «выдавливает» полигоны. После основания я начал работать над диваном. Его я сделал, создав его низкополигональную модель и увеличил количество полигонов с помощью subdivision. Чтобы полигоны были примерно одинакового размера я использовал инструмент cut loop разбивая большие полигоны на более мелкие. После этого я ещё раз использовал функцию subdivision и добавил дивану физическое свойство ткани. Чтобы диван мог держаться за что то, и не падать бесконечно, я создал vertex группу и добавил в неё дно дивана. Теперь нижняя часть дивана оставалась на месте, а вся остальная падала и свисала. Чтобы этого не было я включил в настройках физики ткани (к дивану) такое свойство как pressure (давление) и установил мощность на 10. Стало понятно, что одного закрепленного дна не хватает, поэтому в каркас также стоит добавить все ребра дивана. Также я добавил ещё граней в каркас для того, чтобы было похоже, что у дивана есть подушки.

Затем я приступил к созданию шкафа, который состоит из блоков-полок и в центре находится место для телевизора. Для создания блоков-полок я создавал куб и на двух противоположных сторонах я использовал инструмент insert faces, а затем соединял внутренние стенки полки с помощью клавиши "F" которая создает грань между выделенными вершинами. Заднюю стену я

сделал из куба изменив его масштаб так, чтобы он действительно был похож на стенку шкафа.

Потом я начал делать столики. Один должен был находиться между диваном и шкафом, находясь на уровне столика, другой перед окном и значительно выше, потому что является рабочим. Столики я делал таким же способом, как и заднюю стенку шкафа, т.е. создавал и масштабировал кубы. Стул рядом с рабочим столиком, я делал почти таким же методом как я делал диван, отличие было только в том, что мне не пришлось использовать инструмент *sculpting*, т.к. стул сохранил свою форму.

Для заполнения пространства я решил создать картину, которая бы располагалась над диваном. В её создании не было ничего сложного поэтому про неё мне особо нечего рассказать. А вот про что мне действительно есть что рассказать, так это про лампу, которую я решил поставить на рабочий стол. Для её создания я применил целых три незнакомых мне ранее модификатора: *bevel*, *curve*, *array*. Сначала я сделал подставку для лампы и уже на ней применил модификатор *bevel*, который сглаживает края сетки. Потом я начал делать основание гнущегося как проволока провода. Сделав основания, я поставил в центре основания провода точку, которая может быть поставлена при активации официального аддона *extra objects*. И выдавил её так, чтобы выстроилась линия, на которой провод будет располагаться. После того как линия выстроена её мне надо было конвертировать в кривую (*curve*). Потом я сделал сегмент провода, позже он будет копироваться. Когда сегмент стал готовым я применил на нём модификатор *array* и выставил направление копирования (*z*). Сразу после этого я добавил на сегмент модификатор *curve*, который заставил копии сегмента копироваться по той самой кривой, создавая подобие провода. Осталось только настроить кол-во копий. Когда провод готов можно делать саму лампу, которую я сделал одной единой частью.

Закончив с лампой, я приступил к шкафу, который будет располагаться вокруг двери. Чтобы сделать места хранения для вещей я опять использовал кубы, которые я масштабировал чтобы они подходили по пропорциям. Потом, я сделал *plane* который я поставил так, чтобы он выглядел как передняя стенка шкафа и на этом моделирование шкафа закончилось.

Чтобы заполнить полки я собирался поставить туда книги и самостоятельно смоделировать каждую, но понял, что это будет крайне невыгодно и решил использовать аддон *archimesh*. Он позволяет генерировать различные объекты, среди которых есть собственно и книги, поэтому я воспользовался им. Также у него есть возможность создавать жалюзи, которые я также благополучно использовал.

Дальше я начал всему этому добавлять текстуры. Дивану и стулу я просто добавил коричневую текстуру. Стене на которой висит картина и полу я добавил текстуру темного дерева. Позже я решил добавить эту текстуру задней стенке шкафа, блокам-полкам шкафа и полкам внутри шкафа вокруг двери. Рабочему столу я сделал коричневые ножки и стеклянную столешницу. Такие же текстуры я дал столику между шкафом и диваном. Шкафу вокруг

двери я решил дать белый цвет. Стенам я также дал белый цвет (не считая стену на которой висит картина).

С лампой всё вышло посложнее. Её основная текстура – это черный цвет, однако светящаяся часть имеет белый цвет. Всё дело в том, что я выделил грань, которая должна светиться и пока она выделена создал новый материал, нажал assign и добавил ему свойства свечения.

Во время работы я столкнулся с проблемами развертки. Из-за кривых полигонов в некоторых местах я не мог нормально текстурировать. К счастью я избавился от этих проблем с помощью выделения всех острых углов и использования mark seem и unwrap.

В процессе работы я получил навыки работы в программе 3D моделирования Blender которые я могу использовать в различных сферах, перечисленных ранее. Также у меня есть готовая 3D модель дизайна комнаты.

#### СПИСОК ЛИТЕРАТУРЫ

1. <https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%B7%D0%B0%D0%B9%D0%BD>,
2. <https://univerpl.com.ua/ru/blog/3d-grafika-aktualnost-napravleniya-i-mnenie-eksperta/>,
3. <https://vc.ru/services/473250-top-10-programm-dlya-3d-modelirovaniya-v-2022-godu>,
4. [https://leroyermerlin.ru/product/plenka-samokleyashchayasya-temnoe-derevo-045x2-m-cvet-temno-korichnevyy-81947685/?utm\\_referrer=https%3A%2F%2Fwww.google.com%2F](https://leroyermerlin.ru/product/plenka-samokleyashchayasya-temnoe-derevo-045x2-m-cvet-temno-korichnevyy-81947685/?utm_referrer=https%3A%2F%2Fwww.google.com%2F),
5. <https://profildoors-doors.ru/profil-doors-serija-u/seriya-u-alyaska/mezhkomnatnaya-dver-1u-alyaska>,
6. [https://pikabu.ru/story/morskoy\\_zakat\\_7777114](https://pikabu.ru/story/morskoy_zakat_7777114).

Приложение:



**РЕЛЬСЫ КРЕПЛЕНИЯ ПАССАЖИРСКИХ КРЕСЕЛ САМОЛЁТОВ  
АН-24 И АН-26-100**

*Глушков Алексей Игнатович*

*МБУДО г.Иркутска ЦДТТ, МБОУ г.Иркутска СОШ №23, 9 класс  
г.Иркутск*

Руководитель: Рейнгольд Михаил Григорьевич

Современные самолеты – сложный механизм/устройство, требующее крайнего внимания ко всем своим составляющим без исключения. Все детали

и все части должны исправно выполнять свои функции. Только так самолет будет тем, чем он является – точным, надежным и безопасным средством передвижения, способным выполнять не только транспортные функции. В настоящее время самолеты способны выполнять самые разные задачи. Большинство из них универсальные и готовы быть использованы в целях, которые могут кардинально различаться. Например, самолет АН-148 – среднемагистральный узкофюзеляжный пассажирский самолёт, производимый на воронежском заводе «ВАСО» (Воронежское акционерное самолётостроительное общество). Он может использоваться в качестве пассажирского самолета, в варианте летающего госпиталя. На данный момент этот самолет входит в состав авиационного парка Министерства чрезвычайных ситуаций и Министерства обороны России. Также в этот авиапарк входят самолеты: Ил-76ТД, АН-74П, Бе-200ЧС, ЯК-42Д, SSJ100-95LR.

Из общего количества самолетов РФ, которое составляет 1287 единиц - из них всего 1101 относятся к классу пассажирских, включая 801 борт основной авиации. Остальные 84 – грузовые самолеты, ещё 42 – класса бизнес джетов, а также около 60 самолетов госавиации. Они обозначены как «фактически не участвующие в коммерческих перевозках», отмечает АТОР.

По данным Минтранса РФ, доля самолетов зарубежного производства в коммерческом парке пассажирских самолетов составляет 67,1%. При этом не стоит забывать, что в марте 2022 года из 801 самолета «основной пассажирской авиации» 78 бортов иностранного производства были арестованы за границей из-за санкций. В Россию они вряд ли вернутся. Это означает, что сейчас в большой авиации не 801, а только 723 самолета. «Самолетов российского производства — 157 из 723 или 21,7%. Но этих 157 86% или 136 — это «Сухой Суперджет» (SSJ-100). А это, увы, означает, что и «российская» часть авиапарка очень сильно зависит от поставок иностранных комплектующих». Указывается также, что российскими авиакомпаниями более 700 самолетов (включая грузовые) получено в лизинг. Стоит отметить, что из-за санкций зарубежные лизингодатели не только отзывают свои воздушные суда из России, но и отменяют поставки новой техники и отказываются обслуживать и поставлять комплектующие для бортов, зарубежного производства.

Как уже было сказано ранее для того, чтобы самолет исправно и эффективно выполнял свои функции, все детали самолета должны быть исправны. К сожалению, в ходе эксплуатации, детали самолета со временем приходят в негодность и требуют замены. Учитывая санкции, делать это становится труднее и труднее, так что нужно сосредоточиться на внутренних производителях. В этой работе речь пойдет об одной из множества деталей самолета Ан-24. Это многофункциональный турбовинтовой пассажирский самолет 3-го класса, предназначенный для полетов малой и средней дальности. На данный момент самолет эксплуатируется тремя российскими авиакомпаниями: ИрАэро, UTair и Ангара. С периода 1962 по 1979 год на Киевском, Иркутском и Улан-Удэнском авиационных заводах было выпущено 1367 таких самолётов различных

модификаций и назначений. В своё время Ан-24 обеспечивал до трети воздушного пассажирооборота СССР. Распространённость самолёта объяснялась его неприхотливостью к качеству покрытия ВПП. Изначально самолёту был назначен ресурс в 15000 летных часов. Несмотря на это большинство летных машин до сих пор активно используется, наработка некоторых самолетов превышает 60 000 летных часов и 45 000 посадок. На текущий момент капитальный ремонт воздушных судов парка: Ан-24, Ан-26-100 не осуществляется, что значительно затрудняет их дальнейшую эксплуатацию. Ремонт самолетов осуществляли 403 АРЗ г. Иркутск (ликвидирован в 2008), 412 АРЗ г. Ростов-на-Дону (закрит в 2021). На сегодняшний день единственным предприятием, выполняющим капитальный ремонт, является 308 АРЗ г. Иваново, но основное направление его деятельности-это выполнение ремонта для воздушных судов Министерства Обороны Российской Федерации, гражданские воздушные суда ремонтируются по остаточному принципу, и стоимость данного ремонта превышает 100 000 000 рублей, при этом самолёту назначается межремонтный ресурс-5000 летных часов-5 лет, после чего эксплуатация данного воздушного судна запрещена, что крайне мало, учитывая ежегодную наработку самолёта. Стоит учесть, что абсолютное большинство самолетов эксплуатируется частными авиакомпаниями-самолёт должен не только перевозить пассажиров и грузы, но и приносить коммерческую прибыль. Продолжение эксплуатации без капитального ремонта возможно на основании оценки технического состояния каждого самолёта индивидуально. Эта оценка проводится специалистами ГосНИИ ГА, и позволяет продлить эксплуатацию на один год и три месяца, либо 1000 летных часов и 700 посадок. Основной проблемой является старение планера самолёта и образование коррозии на элементах конструкции

Мой отец работает ведущим инженером по планеру и двигателям в авиакомпании «ИрАэро» и для заказа на авиаремонтном заводе № 308 г. Иваново и дальнейшего проведения ремонта самолёта Ан-24 и Ан-26-100 авиакомпании был необходим цифровой двойник рельс, на которые устанавливаются пассажирские кресла. В будущем необходимо заказывать их изготовление на предприятиях, аккредитованных для производства деталей и комплектующих для авиационной промышленности, что позволит авиакомпании «ИрАэро» продолжить выполнение рейсов по перевозке пассажиров в отдалённые районы: Иркутской области, республики Саха (Якутия), Магаданской области, Хабаровского и Приморского краёв.

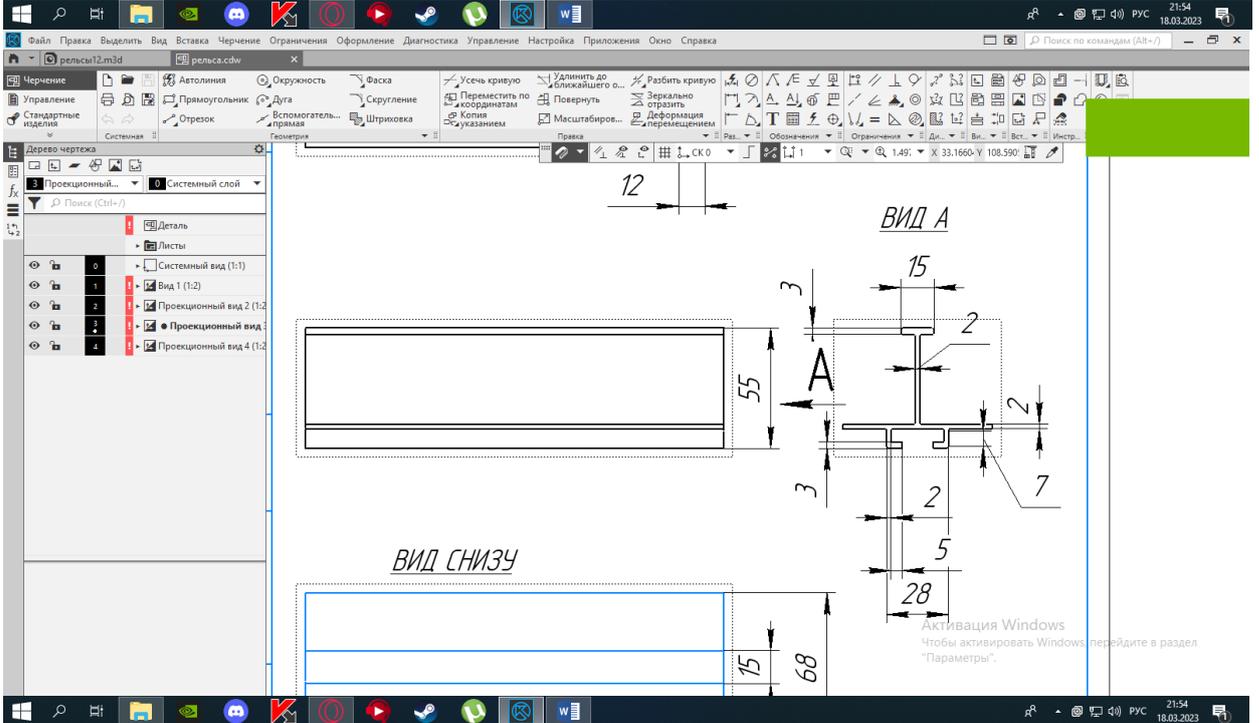
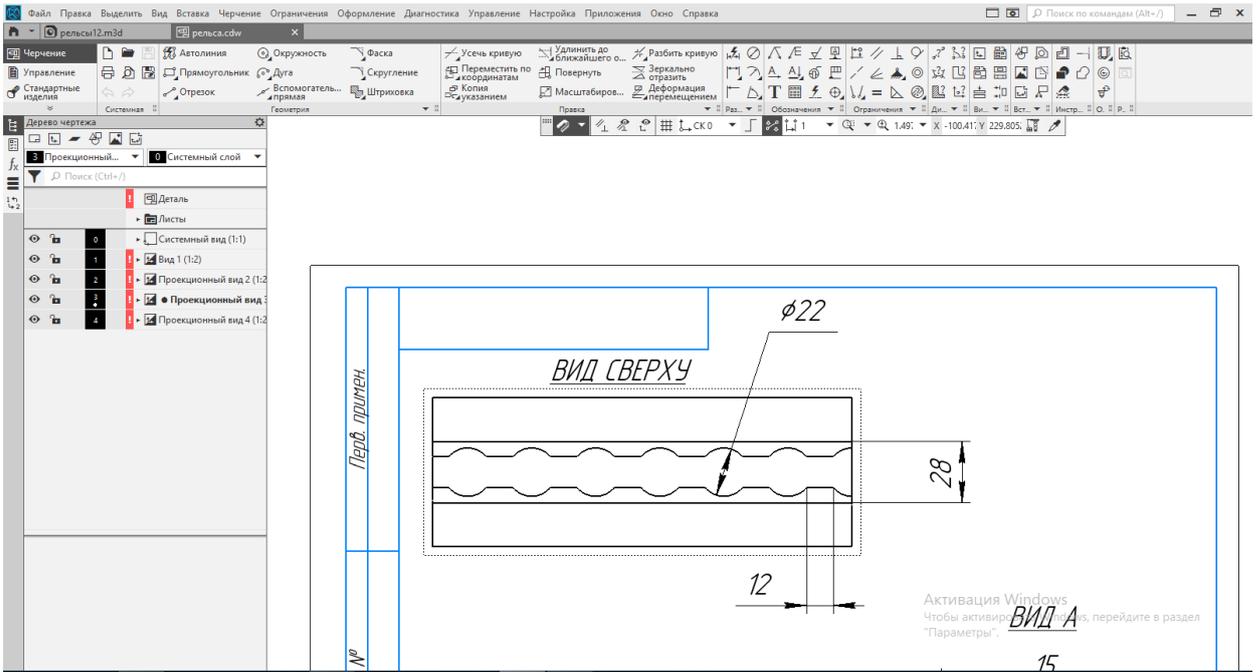
Цель: сделать цифровой двойник рельс крепления пассажирских кресел самолётов Ан-24/Ан-26-100

Задачи:

1. Получить образец детали
2. Снять точные размеры
3. Сделать чертёж
4. Провести 3д-моделирование в САПР
5. Выгрузить из САПР чертежи и модель для передачи на изготовление (на завод).

Мне был предоставлен полноценный образец самого рельса, измерив который я мог создать 3D модель и сам чертеж. Как оказалось, рельс оригинальной конструкции осталось крайне мало. Сам процесс создания чертежа и 3D-модели занял у меня несколько часов. В качестве программы, которую я буду использовать для этих целей, я выбрал САПР «Компас3D». Для начала я измерил общую высоту и ширину рельса с помощью штангенциркуля, и создал скетч в виде прямоугольника, с соответствующими размерами. После чего с помощью команды «Элемент выдавливания» я создал прямоугольник, на котором продолжил создавать более точные очертания самих рельса. Нетрудно заметить, что эта деталь симметрична, это помогло мне при создании этой модели. Убрав лишние части рельса с помощью команды «Вырезать выдавливанием», я смог выделить тот нижний участок, который мне был нужен. Я решил, что сначала можно создать вырез в центре этого выступа. После этого я решил сделать еще один вырез снизу вдоль всего нижнего, для более простого продолжения работы. Далее было необходимо сделать эскизы окружностей, которые были вдоль всей детали. Одним из последних шагов, стало выдавливание этих эскизов. После этого для придания более схожего вида я применил сглаживание там, где это было нужно. Это стало финальным шагом в создании этой модели.

Выводы: создание чертежа успешно завершено, выгруженная в формат STL модель соответствует габаритам реального прототипа. Созданные материалы позволяют отправить на авиаремонтный завод задание на изготовление необходимой детали. Все полученные чертежи, модель и рендеры доступны в приложении к работе, размещённом в облаке: <https://disk.yandex.ru/d/7LbUdGt156-IGw>. Некоторые иллюстрации приложены к этой работе, на Рисунках 1-4.



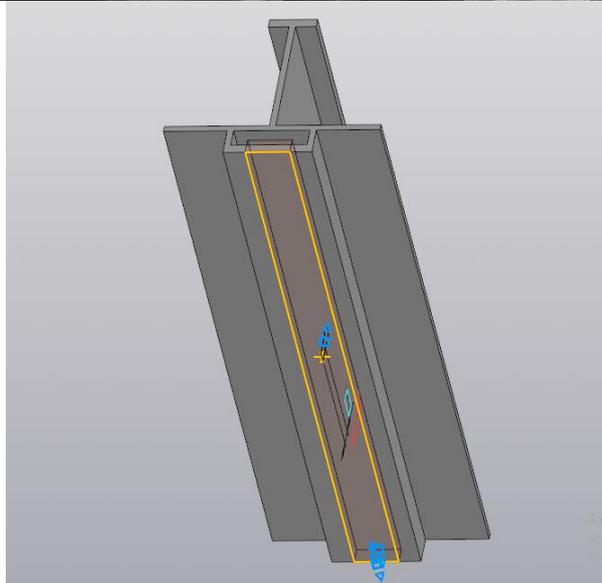
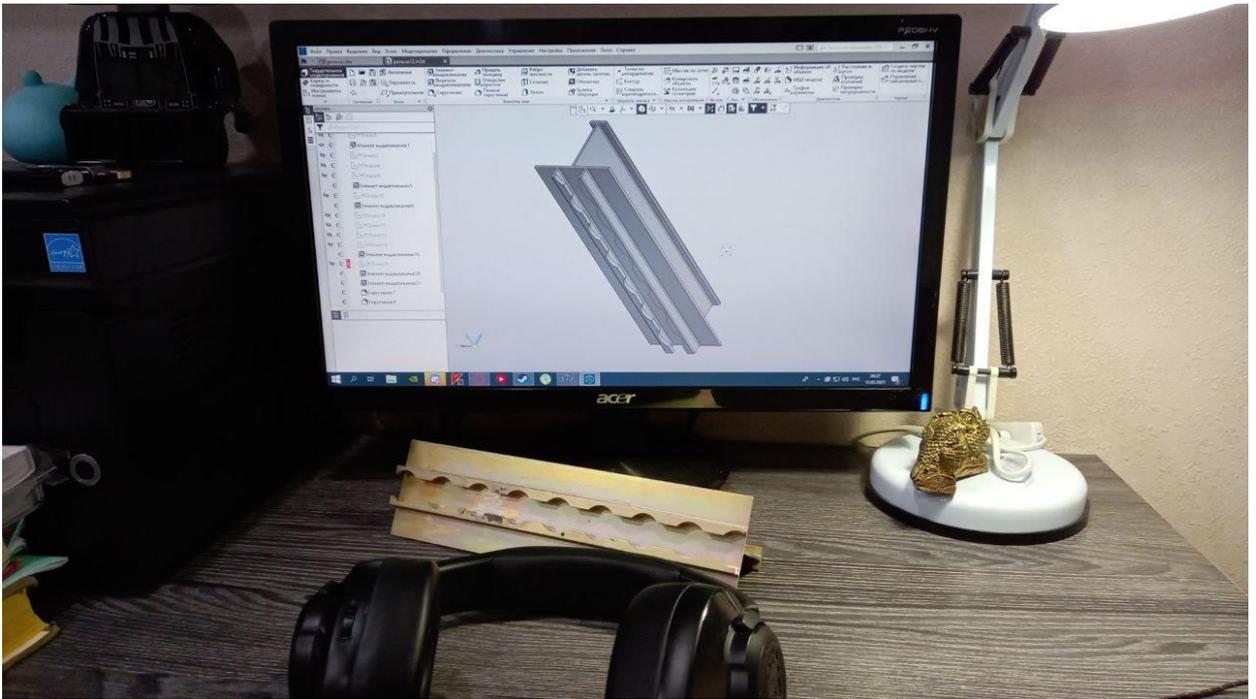


Рисунок 1-4.

## УПРАВЛЕНИЕ БАЗАМИ ДАННЫХ

*Горшунов Олег*

*Муниципальное бюджетное общеобразовательное учреждение лицей при  
ТПУ г. Томска, 10 класс*

Руководитель: Кривошеев Николай Анатольевич, Ассистент ОИТ ТПУ

В информационную эру, когда начали переносить информацию с бумажных на цифровые носители, у людей появилась необходимость в удобном виде записи информации. Важной задачей для разработчиков было создание и оптимизация новых методов хранения информации.

На данный момент хранение информации в базе данных (БД)- один из самых удобных методов. Также информацию можно хранить в электронных таблицах или текстовых файлах, но отличительной особенностью баз данных

является система адресации, при которой адресуется только поле текущей записи, в таблице БД однопользовательская обработка идет по записям, вместо каждой отдельной ячейки, что позволяет оптимизировать работу с данными.

Электронные таблицы и базы данных являются способами управления данными, для неопытных пользователей таблицы будут хорошим вариантом из-за простоты в использовании и удобного интерфейса. Однако использование БД как хранилища предпочтительнее, так как базы данных имеют механизмы защиты (парольная защита, шифрование), возможность для параллелизма. Таким образом электронная таблица - это файл, состоящий из строк и столбцов, который помогает исследовать некоторые данные, файл может иметь числовые и текстовые значения, формулы и функции, а база данных - это файл, который организован для удобной и оптимизированной работы с файлами, в ней осуществляется сортировка, изменение и обслуживание информации, то есть набор информации.

Целью работы было создание программы для управления базами данных при помощи языка программирования Python.

Задачи:

- Запустить портативный локальный сервер Open Server
- Реализовать подключение из приложения Python к базе данных MySQL
- Изучить операции создания, чтения, модификации и удаления элементов баз данных.
- Реализовать SQL-запросы из кода Python к MySQL-серверу

Разбор аналогов. На данный момент в сети Интернет не было найдено полноценных аналогов нашей работы, чтобы мы могли использовать его в качестве референса для выполнения работы.

Целевая аудитория: Вероятнее всего результатом нашей работы смогут воспользоваться люди, которые начинают свой путь в автоматизации бизнес процессов, так как программа довольно простая и не имеет большого количества лишних функций.

Этапы выполнения проекта:

- Выбор среды программирования и веб-интерфейса для администрирования СУБД MySQL

Среди многих сред программирования Visual Studio Code - одна из самых удобных и гибких, позволяет работать со многими языками программирования, а также позволяет легко устанавливать дополнительные библиотеки, в нашей работе используется библиотека pymysql. Помимо среды разработки, для удобства необходим веб-интерфейс для администрирования БД, в нашей работе был выбран интерфейс PhpMyAdmin, это один из самых популярных интерфейсов, которым пользуются многие разработчики и который позволяет запускать команды SQL и просматривать содержимое таблиц непосредственно в браузере.

- Запуск локального сервера

Для работы MySQL и PhpMyAdmin необходим сервер и было принято решение использовать Open Server, что является локальным сервером, имеющим программу для управления с большим функционалом и некоторым количеством подключаемых компонентов. Для запуска сервера необходимо запустить саму программу управления, и через нее запустить сам сервер. Также из программы управления запускается и PhpMyAdmin, что крайне удобно.

•Подключение к базе данных MySQL

Подключения к базе данных MySQL осуществляется посредством языка программирования Python и подключенной библиотеки pymysql при помощи метода connect () в который в качестве аргументов передаются данные сервера, а именно имя хоста при подключении, имя пользователя, пароль, название базы данных, которая будет использоваться. [2]

•Изучение операций создания, чтения, модификации и удаления элементов БД В таблице представлены базовые функции для работы с БД и их описание, также указано как они использованы в нашей программе. [1]

№	Команда	Описание
1	CREATE	Создание объектов в БД. В нашей программе используется для создания таблиц.
2	UPDATE	Модифицирование записей. В нашей программе используется для изменения значений в записи.
3	DELETE	Удаление записей. В нашей программе используется для удаления записи.
4	SELECT	Извлечение записей. В нашей программе используется для получения данных для вывода таблицы.
5	DROP	Удаление объектов в БД. В нашей программе используется для удаления таблиц.
6	INSERT	Создание записи. В нашей программе используется для добавления записей
7	SHOW TABLES	Просмотр доступных БД и таблиц. В нашей программе используется для получения названия таблиц и их вывода

*Таблица 1: Базовые функции, используемые при работе с БД*

•SQL-запросы к MySQL-серверу

После выбора среды программирования и веб-интерфейса, запуска локального сервера, подключения к базе данных и изучения базовых операций стоит заняться реализацией запросов к базе данных посредством языка программирования Python. Для того, чтобы передать запрос к базе данных, для начала необходимо создать сам запрос в строковом типе данных, для этого нужно получить от пользователя информацию о желаемых поправках в БД и после этого основными методами для строк имеющимися в Python, записать

запрос в отдельную переменную, затем с помощью метода.execute() библиотеки MySQL отправить запрос на сервер, где и произойдет изменение.

Предоставление результата. Результатом нашей работы является программа на языке программирования Python, которой можно управлять базами данных, и основная часть которой состоит из двух файлов типа .py в каждом из которых около ста строк кода и поэтому крайне затруднительно представить результат в виде текстового файла.

Заключение.

Мы не можем назвать наш проект ни экономически выгодным, ни экономически не выгодным, так как затраты отсутствуют, как и прибыль.

Выводы проекта

- Запустили портативный локальный сервер Open Server
- Реализовали подключение из приложения Python к базе данных MySQL
- Изучили операции создания, чтения, модификации и удаления элементов баз данных.
- Реализовали SQL-запросы из кода Python к MySQL-серверу

#### СПИСОК ЛИТЕРАТУРЫ

1. Дейт К. Дж. Введение в системы баз данных. 2005. 39-133 с.
2. Документация к библиотеке MySQL. [Электронный ресурс]. - URL: <https://dev.mysql.com/doc/> (дата обращения: 17.03.2023)
3. Документация по языку программирования Python. [Электронный ресурс]. - URL: <https://www.python.org/doc/> (дата обращения: 17.03.2023)

#### **ТЕЛЕГРАМ-БОТ ОНЛАЙН МАГАЗИН КРОССОВОК «Street\_70»**

*Гуренков Родион Константинович*

*МАОУ «Сибирский лицей», 9 класс, г. Томск*

Руководитель: Осинцев Артем Викторович, АНО ДО Технопарк «Кванториум».

Аннотация: Разработка чат бот, работающего как онлайн магазин по продаже кроссовок. В нем пользователь может ознакомиться со всем каталогом кроссовок по выбранному размеру, или подобрать персональную пару кроссовок по нескольким категориям.

Ключевые слова: чат, мессенджер, бот, обувь, продажи.

Актуальность проекта: чат бот – это удобно, быстро, практично и современно. Многие подобные магазины по продаже кроссовок имеют только сайт или сообщество ВКонтакте [1], что последнее время мало пользуется активностью, либо работают только в очном формате. Последнее время многие используют Telegram [2], как основную соц. сеть, следовательно, человеку проще всего обратиться за покупкой обуви к боту.

Введение. По данным интернет-ресурсов [3–4] большинство магазинов пользуются ВК сообществами. Такая деятельность постепенно становится не-

легальной в этой соц. сети. В Telegram [2] пока нет подобных запретов и ограничений. В Томске нет ни одного магазина, у которого есть телеграмм-бот для заказа кроссовок онлайн. Также раньше большинство раньше пользовалось Instagram [5], но теперь это невозможно. При создании бота основное внимание именно на подборе персональной пары и возможность мгновенно оформить заказ.

Сценарий работы чат бота состоит в следующем:

1. открывается бот
2. на экране небольшое приветствие и информация о магазине
3. предложение: посмотреть весь каталог или подобрать персональную пару
4. если нажимают весь каталог, то: выбор размера→выводится каталог
5. для подбора персональной пары: выбор фирмы→выбор размера→выбор модели(по желанию)
6. на экране появляются фотографии кроссовок с доступными размерами, названием, ценой, надпись "заказать"
7. для заказа: - нажать кнопку "заказать" - выбор: "оформить заказ онлайн" или "оформить заказ по телефону" - если заказать онлайн, то высветится окно для указания размера и "оплатить", ссылка для оплаты.
8. Интерфейс прототипа чат бота на Рисунке 1.



*Рисунок 1.*

Выводы и дальнейшие перспективы. Таким образом, представлена разработка прототипа чат бота для мессенджера телеграмм посредством модуля python telebot. В будущем планируется изменить каталог товаров на настоящий.(для проекта был взят для примера каталог из интернета[6]); проработать интерфейс чат бота; улучшить взаимодействие с клиентом.

#### СПИСОК ЛИТЕРАТУРЫ

1. <https://vk.com/?/>
2. <https://web.telegram.org/>
3. <https://gogov.ru/news/888029>
4. <https://www.ekam.ru/blogs/pos/kak-otkryt-internet-magazin>
5. <https://t.me/s/instagram>

#### **РАЗРАБОТКА ИГРЫ В SCRATCH ДЛЯ ЗАКРЕПЛЕНИЯ МАТЕРИАЛА ПО АЛГЕБРЕ И ГЕОМЕТРИИ 7 КЛАССА**

*Дерюшев Михаил*

*Муниципальное автономное общеобразовательное учреждение гимназия № 24 имени М.В. Октябрьской г.Томска*

Руководители: Попов Ю.П., МАОУ 24 гимназия, Красовский А.В.,  
Кванториум, Дерюшева В.Н., к.т.н., ТПУ

Сейчас набирает обороты геймификация. Это адаптивное учебное материала в игру. Преимущества геймификации в том, что, играя, человек отработывает полученный навык или узнает что-то новое, без лишних усилий. Например, решать примеры столбиком занимает больше усилий, чем если же эти же примеры помещены в игровую среду и будут игровым элементом.

Я пользовался подобными играми для отработки скорости сложения, вычитания, умножения и деления и т.д., и собираюсь сделать что-то подобное, используя программу Scratch.

Актуальность темы заключается в том, что у многих учеников математика вызывает затруднение, поэтому создание игры, которая увлечет, поможет отработать навыки освоения материала за 7 класс по алгебре и геометрии.

Цель работы: создать игру в среде Scratch на базе материала по алгебре и геометрии 7 класса.

Задачи

1. Обзор игр для обучения;
2. Разработка структуры уровней игры на базе программы по алгебре и геометрии за 7 класс. Подготовка заданий;
3. Подготовка фона и персонажей, переходы;
4. Программирование логики игры;
5. Тестирование игры;
6. Корректировка по итогам теста.

Объект исследования: создание игр в среде Scratch

Предмет исследования: принципы создание геймификации учебного материала.

Гипотеза: с помощью игры легче осваивать материал по алгебре и геометрии.

Новизна: предложена игра по математике 7 класса.

Методы исследований: поисковый, сравнение, сопоставление, анализ.

Ожидаемый результат: повышенный интерес учащихся 7 класса к алгебре и геометрии.

Практическая значимость: применение игры при закреплении материала по алгебре и геометрии 7 класса.

Существует много различных программ и игр для обучения, например, Duolingo, Puzzle-english, XtraMath, Король математики, Учи.ru, и внутри Учи.ru есть Магическая Математика, О! Математика, Космическая печать и другие. Все они используют элементы игры для обучения. Рассмотрим некоторые из них:

Для изучения языка можно использовать приложение Duolingo (см. рис.1), оно позволяет изучать основные фразы на выбранном языке[1]. Ориентировано на любой возраст. Периодически появляются фразы, которые мотивируют не останавливаться и продолжать. Собираешь монетки, кристаллы за успехи.

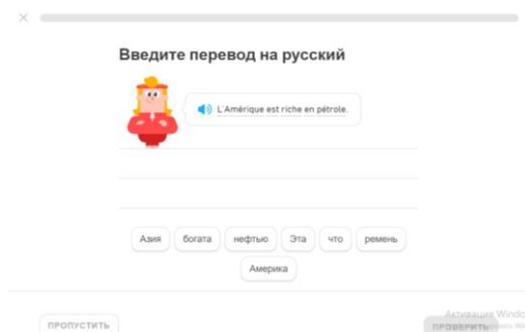


Рисунок 1. Приложение Duolingo

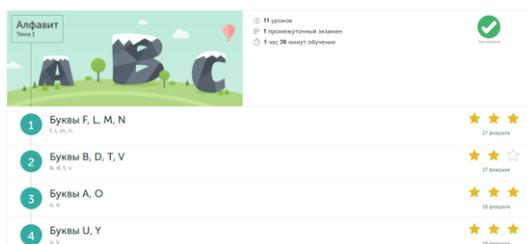


Рисунок 2. Puzzle-English метод Тичера

Puzzle-English – это целый комплекс различных заданий, упражнений и игр по изучению английского языка [2]. Подробнее хочу остановиться на «Метод Тичера» (см. рис.2). Метод Тичера разбит на несколько курсов, каждый курс разбит на темы, а сами темы на уроки. Проходя урок, ты получаешь 1-3 звезды, можно наблюдать свой прогресс.

Ответы в заданиях можно выбирать или собираются фразы как пазлы. Небольшая длительность уроков помогает не уставать, а наличие звездочек увлекает и помогает стремиться делать меньше ошибок. Есть бесплатный и платный контент.

Рассмотрим математические игры.

XtraMath (см. рис.3) – сервис, позволяющий отрабатывать на скорость примеры на умножение, сложение, вычитание и деление [3]. Абсолютно бесплатен. Требуется регистрация. Нацелен на учеников начальных классов. Элементы геймификации: фиксированное время на решение примеры, поощрение в виде смайликов за каждый пример, отслеживание прогресса.

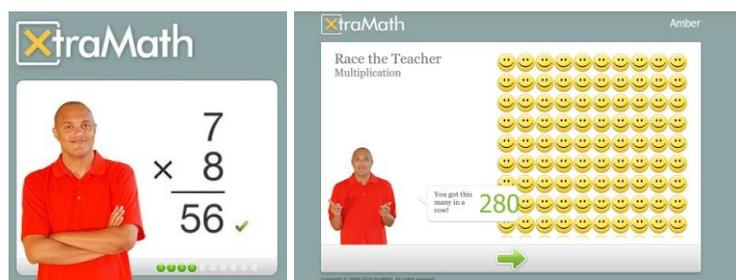


Рисунок 3. Сервис XtraMath

Король математики (см. рис.4) – игра, где ты становишься фермером и должен подниматься по иерархической лестнице, решая задачи [4]. Это приложение для телефонов, его надо скачать, в отличие от предыдущего сервиса. И бесплатно можно получить только небольшую часть. Есть как для начальных классов, так и для среднего звена, до 9 класса.

Уровни отображаются в виде 12 книг. Все начинается с простого сложения и вычитания, затем впоследствии предстоит вспомнить о том, что такое: умножение, деление, геометрия, дроби, степени, статистика, уравнение и еще пара смешанных книг, которые включают в себя вопросы из всех вышеперечисленных областей. На каждую задачу предложены 4 варианта ответов. Полная версия стоит 119 р.

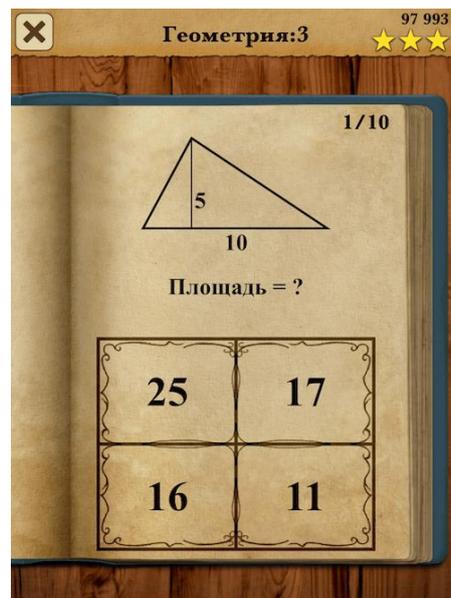


Рисунок 4. Игра король математики

Магическая Математика (см. рис.5) – очень продуманная игра, где ты

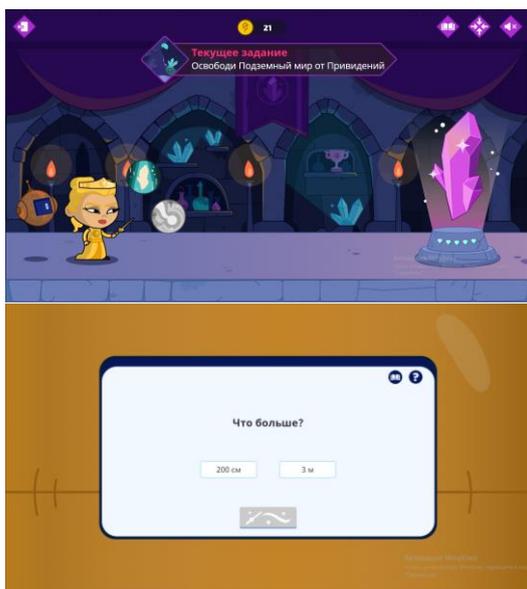


Рисунок 5. Магическая математика

сражаешься с монстрами, боссами и другими игроками [5]. Внутри игры включены материал с 1-го по 6-ой классы. Есть бесплатная демо-версия, но в ней есть только первая глава, часть второй, и нет возможности играть с другими игроками. За игру нужно отдать около 1000 рублей.

Так же недостатком магической математики является то, что она сделана до 6 класса.

И таким образом, на данный момент найдена одна игра с математикой 7-го класса, Король математики. Недостатком является то, что нет возможности отделить один класс от другого, следовательно, нет возможности переключиться на нужный класс для отработки. Выбор

ответа позволяет случайным образом выбрать правильный, не решая задачу.

Поэтому необходимо разработать игру, ориентированную на математику 7-го класса, которая позволяет вписывать ответ.

Разработка игры.

Начала выписали перечень тем по алгебре и геометрии за 7 класс:

Алгебра:	Геометрия:
Уравнения	Смежные углы

<p>Текстовые задачи          Функции          Степени          Формулы сокращенного умножения          Система уравнений.</p>	<p>Вертикальные углы          Признаки равенства треугольников          Периметр равнобедренного треугольника          Свойства равнобедренного треугольника          Признаки параллельности          Свойства углов при параллельных прямых</p>
---	---



Рисунок 6. Локация лес уравнений.

Исходя из списка было решено, что каждая локация игры — это тема по алгебре, внутри локации 7 уровней сложности. А вот темы по геометрии вошли в одну локацию. Например, первая локация Лес (см. рис.6) и там будут уравнения.

В игре предложено 7 уровней сложности локации лес уравнений. На каждом уровне игрок (волшебник) атакует враг (слайм) нажатием левой кнопки мыши, после чего появляется уравнение (рис.7.).

Для того, чтобы победить слайм нужно решить 4 уравнения, которые выбираются из базы случайным образом. Перечень уравнений по уровням приведены в приложении 1. У игрока есть возможность ошибиться только 10 раз, после чего игрок начнем игру с начала.

Победив слайм, происходит переход на следующий уровень. Игрок приобретает новую особенность внешности, но перед ним появляется новый вид слайма, который больше. Все виды волшебников и слаймов приведены в таблице 1.

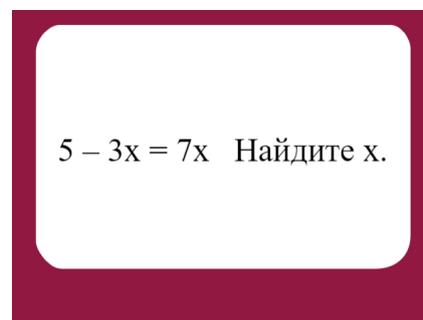


Рисунок 7. Пример уравнения



Рисунок 8. Переход между первым и вторым уровнем.

После каждого уровня есть переход, который отображается салютом волшебных палочек (см. рис.8) и вспышкой перед игроком.

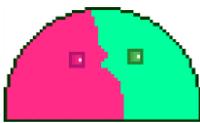
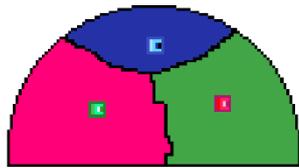
Враги из уровня в уровень отличаются цветом, наличием одного, двух или даже трех глаз.

База для волшебника была взята в интернете, отредактирована в программе для рисования, увеличив разрешение, и

нарисовано несколько вариаций его. Фоны для каждой локации игры взяты в библиотеке Scratch.

База с заданиями (см. приложение 1) по алгебре пополнялась из сборника задач 57 школы [6], а заданиями по геометрии из задачника Балаяна [7].

Таблица 1. Волшебники и слаймы по уровням сложности локации лес.

1 уровень	2 уровень	3 уровень	4 уровень
			
			
5 уровень	6 уровень		7 уровень
			
			

#### СПИСОК ЛИТЕРАТУРЫ

1. Duolingo. <https://www.duolingo.com/learn> (дата обращения - 16.03.2023)
2. Puzzle-english. Английский язык для начинающих. Вы научитесь поддерживать простую беседу, изучите грамматику и произношение без преподавателя <https://puzzle-english.com/teacher> (дата обращения - 16.03.2023)
3. XtraMath. Экстра математика. <https://ru.xtramath.org/#/home/index> (дата обращения - 16.03.2023)
4. Король математики. <https://apps.apple.com/ru/app/korol-matematiki/id473904402> (дата обращения - 16.03.2023)
5. Магическая математика. [https://uchi.ru/magic\\_rpg\\_math/game](https://uchi.ru/magic_rpg_math/game) (дата обращения - 16.03.2023)
6. Сборник задач по математике 7 класс. Пятьдесят седьмая школа. 2014/2015 уч.год.
7. Балаян Э.Н. Геометрия: задачи на готовых чертежах для подготовки к ГИА и ЕГЭ: 7-9 классы / Э.Н.Балаян. – Изд. 5-е, исправл. И дополн. – Ростов н/Д: Феникс, 2013. – 223с. – (Большая перемена).

## **WEB-ПРИЛОЖЕНИЕ «ФИНАНСОВЫЙ КАЛЬКУЛЯТОР»**

*Дикилиева Элина, Кирюшкина Полина, Осипов Вячеслав, Егоров Юрий*

*Муниципальное бюджетное общеобразовательное учреждение лицей при*

*ТПУ, 10 класс*

*г. Томск*

Руководитель: Савельев Алексей Олегович, к.т.н., доцент ОИТ ТПУ

Проблема и актуальность: наш проект ориентирован на создание инструмента для помощи в расчётах на Всероссийском чемпионате по финансовой грамотности и предпринимательству Юниор-лиге U-15 (школьники 6-9 класса, до 15 лет). Мы хотим создать сайт, упрощающий расчёт некоторых дополнительных условий. Школьники до 15 лет зачастую не обладают знаниями, с помощью которых они быстро смогут рассчитать данные по кредиту, вклад через определённый промежуток времени и возможный вариант сокращения расходов. Нам нужно создать инструмент для участников ВЧФГ, помогающий использовать некоторые дополнительные условия при решении кейсов.

Цель: получение сайта – финансового калькулятора.

Задачи:

1. определить целевую аудиторию;
2. определить функции и создать структуру сайта;
3. определить наполнение сайта;
4. разработать программную часть;
5. запустить сайт;
6. устранить ошибки.

Поскольку в основном у школьников проблема с расчётом дополнительных условий при решении финансовых задач, то мы решили создать инструмент для Юниор-Лиги U15 ВЧФГ (школьники 6-9 класса, до 15 лет).

Мы провели поиск похожих сайтов. Банки.ру — самый большой финансовый маркетплейс в России. (рис. 9) Данный сайт может считать вклады, займы, кредиты, инвестиции. Проведя анализ этого сайта, мы выявили его недостатки, чтобы, учитывая их можно было определить функции нашего сайта, который должен иметь над Банки.ру преимущества. Недостатки:

- отсутствие возможности ввода финансовых данных;
- отсутствие плана сокращения расходов.

Функции сайта:

- планирование вклада и вывод суммы на вкладе через указанный срок (с возможностью выбора: с капитализацией процентов или без неё);
- определение возможности выплаты кредита пользователем на основе финансовых данных;
- отображение ежемесячного остатка на основе введённых данных;
- применение возможного варианта сокращения расходов.

Кредит — это финансовые отношения двух сторон, где одна предлагает деньги, а вторая обещает их вернуть в указанные сроки. Заемщик возвращает сумму постепенно, обычно равными платежами, уменьшая при этом свою задолженность перед кредитором. [6]

Существуют два вида внесения ежемесячного платежа по кредиту: дифференцированный и аннуитетный. Дифференцированный способ ежемесячного внесения платежей применяется редко, так как предполагает внесение наибольших выплат в первое время кредита. В ситуации с аннуитетным способом выплаты ежемесячные платежи остаются постоянными. [3] Поскольку наиболее часто используется аннуитетный способ платежа, то для расчета ежемесячной ставки мы использовали данную формулу:

$$x = S * (P + P / ((1 + P)^N - 1))$$

где  $x$  - месячный платеж;  $P$  - 1/12 процентной ставки;  $S$  - первоначальная сумма кредита;  $N$  - количество месяцев. [2]

Депозит – банковский продукт, в результате которого клиент передает деньги банку на хранение на определенное время. В соблюдении требований контракта, банк предписывается предоставить средства обратно и добавить к ним определенный процент. [4] Вклад — сумма средств, принимаемой банком от клиента на определенный или неопределенный срок. Вклад — это вид депозита. Их основное отличие состоит в том, что на депозите могут храниться любые ценности, а на вкладе — только деньги.

Капитализация процентов – это причисление начисленных процентов к сумме вклада. То есть, это общая доходность по банковскому продукту. Чем чаще периодичность капитализации, тем выше доход по вкладу. [5]

Формулы для расчёта вкладов:

- без капитализации процентов:  $x = a * (1 + p * n / 100)$ , где  $a$  – сумма вклада,  $p$  – процентная ставка,  $n$  – срок размещения вклада;
- с капитализацией процентов:  $x = a * (1 + p / 100)^n$ , где  $a$  – сумма вклада,  $p$  – процентная ставка,  $n$  – срок размещения вклада. [7]

Для создания понятной интерактивной среды мы составили структуру сайта – схема описывающая связь страниц сайта (схема 1). Далее нужно было определиться с наполнением сайта. На главной странице мы представили информацию о функциях сайта и кнопку, переносящую пользователя на подстраницы входа и регистрации. После завершения входа или регистрации пользователя переносят в профиль. Из профиля можно перейти на другие подстраницы с расчётом финансовых данных — расчётом вклада, кредита и плана по сокращению расходов.

Для сокращения расходов мы разработали таблицу с диапазонами и соответствующими им коэффициентами сокращения (табл. 1). Диапазоны вычисляли на основе данных различных статистик. [8]

Одной из главных частей нашего проекта являлось программирование.

Frontend — видимая пользователям часть сайта. Иными словами, это весь внешний вид сайта, то есть текст, изображения, кнопки, динамические

слайды, анимация и другое. [1] Мы подобрали стандартный набор инструментов для разработки frontend: HTML, CSS, JavaScript.

Backend — внутренняя часть сервиса, которая работает не на компьютере, а на сервере. [1]

Для написания backend мы решили использовать язык программирования Python и веб фреймворк flask. Веб фреймворк — это набор инструкций и инструментов для создания и поддержки сайтов. Также мы использовали СУБД SQLite и текстовый шаблонизатор Jinja2.

Итоговый сайт состоит из главной страницы (рис.1), страницы входа (рис.2), страницы регистрации (рис.3), страницы профиля (рис. 4), страницы завершения регистрации (рис.5), страницы расчёта кредитов (рис. 6), страницы расчёта вкладов (рис. 7).

#### СПИСОК ЛИТЕРАТУРЫ

1. Простыми словами о «фронтенде» и «бэкенде»: что это такое и как они взаимодействуют. [Электронный ресурс] - Режим Доступа: <https://tproger.ru/translations/frontend-backend-interaction/> (дата обращения 12.02.2023) [1]
2. Формула расчёта процентов по кредиту: [Электронный ресурс] - Режим Доступа: <https://kredit-on.ru/kak-rasscitat-procenty-po-kreditu-formula/> (дата обращения: 12.02.2023) [2]
3. Аннуитетный и дифференцированный платеж: в чем отличия? [Электронный ресурс] - Режим Доступа: <https://спроси.дом.пф/instructions/annuitetnyu-i-differentsirovannyy-platezh-cto-eto-i-v-chem-otlichiya/> (дата обращения: 12.02.2023) [3]
4. Что такое депозит [Электронный ресурс] - Режим Доступа: <https://www.poidem.ru/blog/depozit-v-banke-cto-eto> (дата обращения: 12.02.2023) [4]
5. Как работает капитализация процентов [Электронный ресурс] - Режим Доступа: <https://www.banki.ru/news/daytheme/?id=10978528> (дата обращения: 12.02.2023) [5]
6. Что такое кредит простыми словами [Электронный ресурс] - Режим Доступа: <https://www.vtb.ru/articles/cto-takoye-kredit/> (дата обращения: 12.02.2023) [6]
7. Формулы для расчёта вкладов: [Электронный ресурс] – Режим Доступа: <https://com-business.ru/sloznyj-procent-po-vkladam-formula-sloznyh-procentov/> (дата обращения: 13.03.2023) [7]
8. Статистики о расходах россиян. [Электронные ресурсы] – Режим Доступа:
9. <https://journal-tinkoff-ru.turbopages.org/journal.tinkoff.ru/s/plate-statistics/>
10. <https://rb-ru.turbopages.org/rb.ru/s/news/purchase-change/>
11. <https://rb-ru.turbopages.org/rb.ru/s/news/oplata-podpisok-russia/>
12. <https://turbo.ria.ru/20210915/dosug-1750099970.html>
13. <https://1prime-ru.turbopages.org/1prime.ru/s/transport/20220812/837769690.html>

14. <https://1prime.ru.turbopages.org/1prime.ru/s/transport/20220812/837769690.html> (дата обращения: 15.03.2023) [8]

Таблица 1

Категория расходов	Диапазон введенной суммы, руб.	Сокращение (рекомендуемый %)	Коэффициент сокращения
Еда	От 0 до 10000	0%	1
	От 10001 до 20000	30%	0,7
	От 20001	70%	0,3
Одежда и обувь	От 0 до 20000	0%	1
	От 20001 до 35000	40%	0,6
	От 35001	65%	0,35
Коммунальные услуги		0%	1
Услуги сотовой связи и Интернет		0%	1
Подписки на различные сервисы	От 0 до 1500	0%	1
	От 1501 до 2000	25%	0,75
	От 2001	75%	0,25
Развлечения	От 0 до 3000	0%	1
	От 3001 до 6500	50%	0,5
	От 65001	80%	0,2
Личный автомобиль		0%	1
Автобус и такси	От 0 до 2500	0%	1
	От 2501 до 4000	30%	0,7
	От 4001	65%	0,35
Домашние животные	От 0 до 4000	0%	1
	От 4001 до 6000	30%	0,7
	От 6001	50%	0,5

Приложение:  
Схема 1

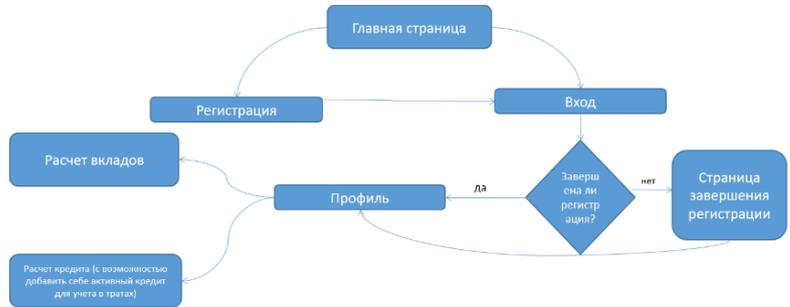


Рисунок 1



Рисунок 2

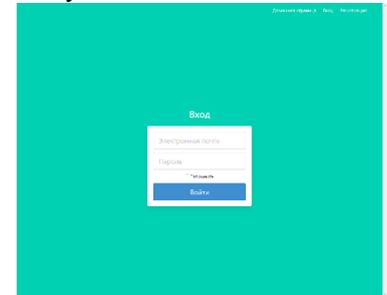


Рисунок 3

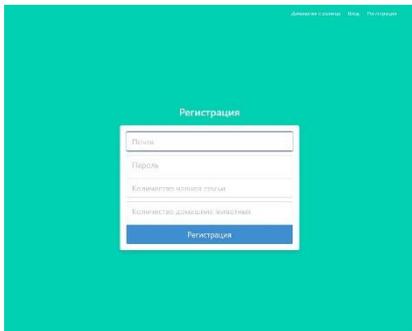


Рисунок 6

Рисунок 4

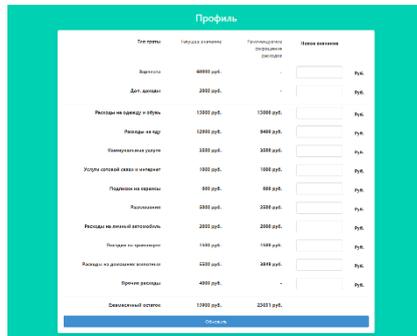
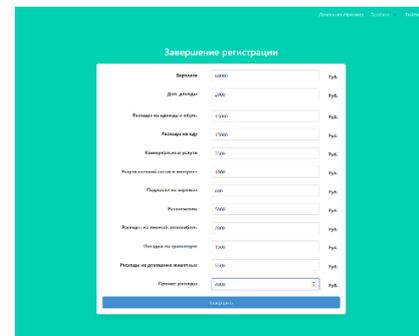
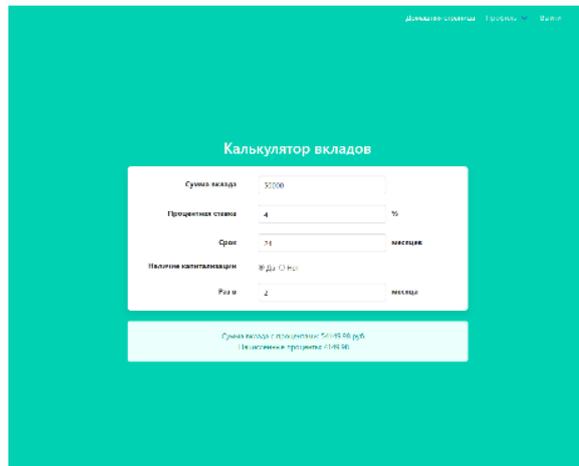
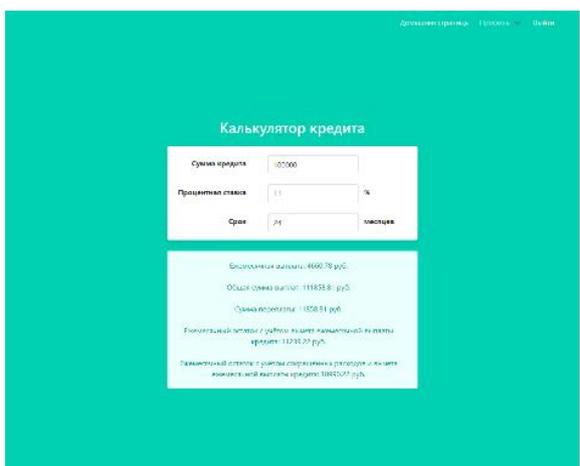


Рисунок 7

Рисунок 5





## ПЛАТФОРМА ДЛЯ ИЗУЧЕНИЯ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ УЧЕБНЫХ ОРГАНИЗАЦИЙ

***Думин Андрей Павлович***

*Автономная некоммерческая организация дополнительного образования  
«Детский технопарк «Кванториум», Частная школа «Резонанс».*

***Сиренко Егор Александрович***

*Образование: Томский Механо-Технический техникум, 1 курс*

*Руководитель проекта: Осинцев Артем Викторович*

*АНО ДО «Детский технопарк "Кванториум"»*

Аннотация проекта: Данный проект направлен на разработку платформы, предназначенной для учеников 5-11 классов, которые хотят самостоятельно изучать высокоуровневые языки программирования. Основной целью платформы является предоставление ученикам возможности изучения языков программирования в удобном и доступном формате, а также обеспечение их интерактивного взаимодействия с обучающимися материалами.

На платформе будут представлены курсы по языкам программирования, составленные опытными специалистами, которые будут включать в себя как теоретические, так и практические материалы. Ученики смогут проходить курсы в любом удобном для них темпе. Так же платформа имеет алгоритм, который выявляет проблемные места в изучение, и подбирает аналогичные задания, для отработки и закрепления теории на практике.

Кроме того, платформа будет иметь различные функциональные возможности, такие как анализ уровня знаний, дополнительные задания и учебные материалы для расширения знаний и навыков.

Таким образом, разработка данной платформы поможет ученикам в самостоятельном изучении высокоуровневых языков программирования, а также предоставит им уникальный опыт интерактивного обучения, позволяющий эффективно использовать свое время и достигать успеха в обучении.

Ключевые слова: Платформа для самостоятельного изучения программирования, Изучения языков программирования, Курсы по программированию, Интерактивное обучение программированию, Программирование, Самостоятельное изучение.

Введение: В 2023 году отрасль Информационных технологий крайне востребована, многие школьники заинтересованы в продвижение себя, в данной отрасли. В ходе нашего проекта были проведены статистические-анализы в ходе которых мы выявили проблемы с которыми сталкиваются обучающие 5-11 классов при самостоятельном обучении языкам программирования. Во время опроса было выявлено что 71.4% участников опроса имеет интерес к изучению языков программирования, из них 23.4% уже изучают. Участники опроса указали, что им интересно изучать Python, C++, Java. Так же 80% участников опроса указали, что хотят в дальнейшем связать себя со сферой информационных технологий. В ходе опроса были выявлены проблемы, с которыми столкнулись участники опроса:

- Отработка информации и ее систематизация.
- Поиск информации и ресурсов для обучения.
- Недостаток практики в ходе обучения.
- Отсутствие обратной связи при обучении.

Так же участники опроса, сказали что хотели бы получать проекты, или принимать участие в них, для продвижения себя.

Обзор аналогов:

CodeLearn[1] - платформа на английском языке для изучения языков программирования. Плюсы: Огромное изобилие материала, и наличие хороших курсов. Минусы: Сайт полностью на иностранных языках, много курсов не все проверенные, так же мало практики, и отсутствует обратная связь или анализ освоения.

FreeCodeCam[2]- Платформа для изучения языков программирования на английском языке. Плюсы: Огромное изобилие материала, и наличие хороших курсов. Минусы: Сайт полностью на иностранных языках, курсы заточены на профессионалов и людей которые имеют опыт, так же мало практики, и отсутствует обратная связь или анализ.

Яккласс[3]- Российская платформа для изучения языков программирования. Плюсы: доступность и хорошо подобранный материал. Минусы: Устаревший язык программирования и отсутствие достаточно практики.

Основная часть:

В ходе разработки платформы были проведены различные статистическо-аналитические работы, в результате которых была получена информация о проблемах, с которыми встречаются школьники при изучении языков программирования. Так же параллельно процессу сбора данных о проблемах велась разработка прототипа платформы, графическая часть и работа над алгоритмом, который и позволит нашей платформе стать отличным инструментом для обучающихся которые заинтересованы в изучении языков программирования.

Сайт сделан с крайне простым читабельным дизайном, для более простого взаимодействия пользователя с ним. Дизайн сайта сделан на основе HTML academy, так как их дизайн довольно хорошо воспринимается визуально. Сайт так же имеет страницу на которой будет реализована проектная деятельность, и буду показаны проекты в которых пользователи могут принять участие. Окно решения задач включает в себя блок с теорией, место с информацией о задаче(Дано, условия, требования), окно вывода(Консоль), и окно компилятора(Окно где пользователь будет писать код). Основной фишкой сайта является алгоритм, который работает по следующему принципу: «Получает данные о решении задачи пользователем, выявляет материал в котором у пользователя возникли трудности в ходе изучения, и подбирает задания на основе данного материала, чтобы пользователь мог отработать на практике материал, и более хорошо его освоить. Таким образом данный алгоритм может стать хорошим наставником и помочь пользователю даже в самых тяжелых ситуациях». Такой метод практики будет крайне полезным для тех кто планирует сдавать информатику как экзамен(ОГЭ,ЕГЭ), так как программирование это в первую очередь практика, то наша платформа направлена именно на данную область изучения. Для более лучшего результата мы решили взять мнение профессиональных разработчиков, которые согласились дать нам обратную связь, мы презентовали им наш проект и рассказали как именно он поможет юным программистам. После чего получили компетентный отзыв от профессионалов.

Наша платформа принципиально отличается от аналогов, потому что у нас различаются цели и целевая аудитория. Допустим автор данного проекта Андрей Думин при изучении языков программирования столкнулся с проблемой не адаптивного материала для своего возраста, что очень осложнило изучения языков программирования. Так же проблемы заключаются в том что множество действительно хороших платформ иностранного производства из-за чего отсутствие русского языка может оттолкнуть юного программиста от данного ресурса. Данные проблемы были отмечены при личной беседе с участниками тестирования демоверсии платформы.

С прототипом программы можно ознакомиться по ссылке [4].

#### СПИСОК ЛИТЕРАТУРЫ

1. <https://codelearn.io/learning/>
2. <https://www.freecodecamp.org/learn/>
3. <https://www.yaklass.ru/p/informatika/8-klass/nachala-programmirovaniia-14963/nachalnye-svedeniia-o-iazyke-programmirovaniia-paskal-14000>
4. Архив с фотографиями проекта <https://disk.yandex.ru/d/poDxg8B-TMMQbw>

#### **СОЗДАНИЕ РОБОТА ВВ-8 (РАЗРАБОТКА СХЕМЫ, НАПИСАНИЕ И ОТЛАДКА ПРОГРАММЫ, СБОРКА УСТРОЙСТВА) НА МОДУЛЯХ ARDUINO.**

## ***Шипачев Никита, Емельянов Кирилл***

*Муниципальное бюджетное общеобразовательное учреждение лицей при  
ТПУ, 10 класс  
г. Томск*

Руководитель: Костина Мария Алексеевна, доцент ОЭИ ИШНКБ ТПУ

Робот ВВ-8 является дроидом-астромехаником кино вселенной “Звёздные войны”. [1] Реализация проекта не требует профессиональных знаний в области робототехники благодаря использованию платформы Arduino. [2] Базирование и упор в работе платформы делается на простое в использовании программное обеспечение и оборудование. Поэтому возросло количество разработчиков и потребителей продуктов, сделанных на основе модулей Arduino. В результате, проекты, не уступая по функционалу аналогам от серийных производителей, могут быть дешевле производственных моделей в несколько раз.

Рассмотрим один из аналогов, сделанный в Китае: BB8 RC Robot V4.0 [3], цена которого составляет 8555,37 руб (по состоянию на 07.03.2023). Диаметр его сферы составляет 11,5 см, высота вместе с головой - 15 см. С помощью отвёртки есть возможность менять две батареи “AAA”. Время работы при одном заряде составляет 30 минут. В комплекте есть шнур для подзарядки. Предусмотрено звуковое сопровождение. Голова поддерживает положение во время движения корпуса, управление которого осуществляется с помощью пульта в радиусе 30 метров. Мы приходим к выводу, что у робота довольно обширный функционал. Тем не менее, его цена довольно высока, управление с помощью пульта не позволяет управлять роботом в любое удобное время и с нескольких устройств, что отрицательно сказывается на мобильности данного продукта. В нашем прототипе робота ВВ-8 мы постарались решить эти проблемы за счёт заказа комплектующих по выгодным ценам и осуществления управления роботом с помощью мобильного телефона.

Целью работы являлось создание работающего прототипа робота ВВ-8, управляемого с помощью приложения, на базе платформы Arduino.

Перед нами стояли такие задачи:

1. Собрать контроллер для управления моторами, приводящими в движение конструкцию.
2. Запрограммировать контроллер для работы с приложением для управления роботом.
3. Распечатать на 3D-принтере две полусферы-оболочки из пластмассы одинакового диаметра. Сделать каркас, на который будет закреплен контроллер.
4. Декорировать работу с помощью красок и малярного скотча.

Перейдем к описанию компонентов робота. За подключение к компьютеру, ввод и вывод данных отвечает микроконтроллер Arduino Uno R3. [4] Его работа координируется программным обеспечением на языке программирования C++. За координацию движений моторов ВВ-8 отвечает непосредственно подключающийся к Arduino Uno R3 моторный драйвер Arduino L293D. [5] Он

может управлять несколькими видами моторов. В случае робота ВВ-8 - это DC-моторы. Данный вид моторов способен выполнять вращательные движения и изменять направления движения. Для реализации робота потребуется следующий вид DC-моторов: два мотора-редуктора 120 TT Gear Motor и колесо к каждому.[6] Они подключаются одним проводом к моторному двигателю в каналы Arduino L293D, предназначенные для подключения к DC-мотору. На рис.1 возможные каналы подключения обозначены как “four channels for DC Motors”: Питание осуществляется с помощью крон с напряжением 9 Вольт каждая. Для спайки проводов был использован паяльник.

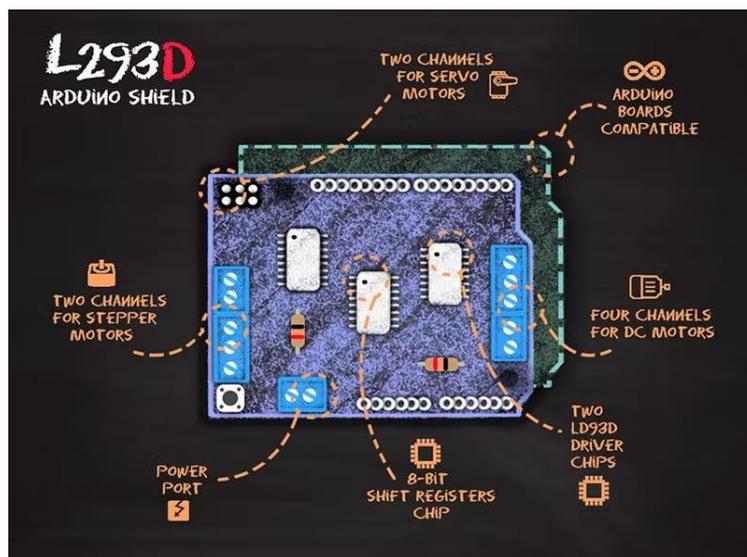


Рисунок 1. Arduino Motor Shield L293D [5].

Управление движениями робота в двухмерной системе координат осуществляется через мобильное приложение “Bluetooth RC Controller”.[7] Сопряжение контроллера с установленным приложением к ВВ-8 осуществляется посредством подключения к Arduino L293D bluetooth модуля HC-06.[8].

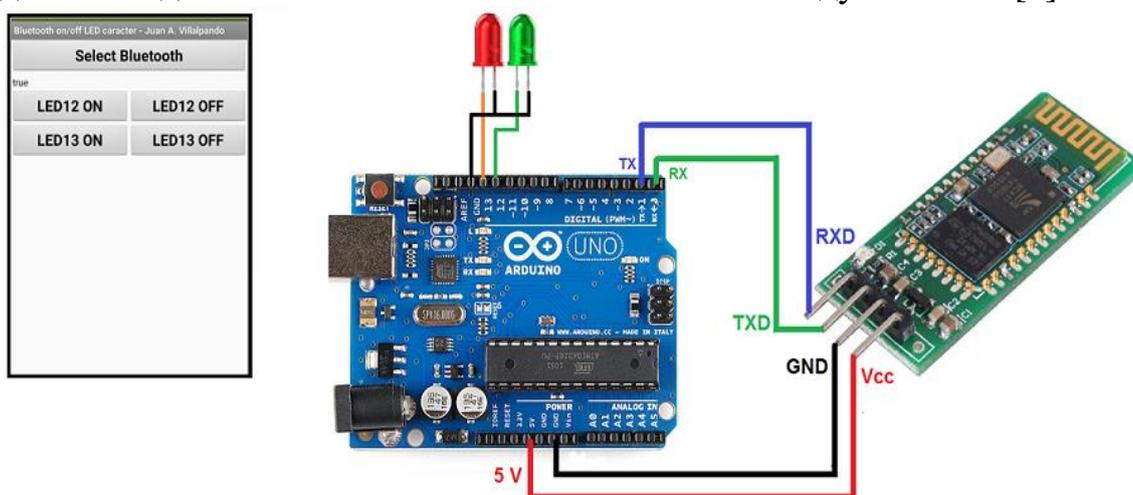


Рисунок 2. HC-06, Bluetooth module [9].

Код для управления роботом разрабатывался и компилировался в системе Arduino IDE.[10]

Следующим этапом было создание и декорирование оболочки. Пластмассовые полусферы диаметром 19,5 см, напечатанные на 3D принтере, оформлены с помощью аэрозольных красок оранжевого, черного и белого цвета. Пенопластовая голова диаметром 15 см представляет собой полусферу, вырезанную из пенопластового шара. Она окрашена маркером в оранжевый цвет и имеет два черных глаза, напечатанных на 3D - принтере.

После сборки и отладки конструкция помещается в оболочку в виде шара. Равновесие поддерживается магнитами, взаимодействующими между головой и движущейся внутренней конструкцией через оболочку робота.

Бюджет проекта обозначен в таблице 1. Цены представлены по состоянию на 07.03.2023 в магазине “Яндекс-Маркет”.[11]

*Таблица 1. Бюджет, необходимый для создания робота:*

Компоненты/материалы, шт.	Цена
Arduino Uno R3, 1 шт	560 Р
120 TT Gear Motor, 2 шт	544 Р
HC-06 bluetooth модуль, 1 шт	659 Р
L293D Motor Shield, 1 шт	319 Р
Пенопластовый шар диаметром 15 см, 1 шт	295 Р
Эмаль Kudo аэрозоль оранжевая RAL 2004 520 мл, 1 шт	259 Р
Аэрозольная эмаль REF белый 520 мл, 1 шт.	238 Р
KUDO аэрозольная автоэмаль KU-1002 черный, гляцевый, 520 мл, 1 шт	299 Р
Неодимовый магнит, 6 шт	60 Р
BRAUBERG Маркер меловой POP-ART 5 мм, оранжевый, 1 шт	166 Р
Набор проводов «Мама — Папа», для Arduino, 10шт	60 Р
Батарейка GP Super Alkaline 9V Крона, 2 шт	390 Р
Малярный скотч, 10 метров	26 Р
ДЕКО Клеевой стержень 11x200 мм прозрачный, 1 шт	26 Р
Пластик PETG для 3D принтера белый Plastic, 1.75мм, 30 метров	100 Р
Итоговая цена:	4001 Р

Подсчитав суммарную цену компонентов, мы увидим, что себестоимость разработанного нами прототипа, равная 4001 руб., приблизительно в 2,1 раза меньше, чем стоимость упомянутого выше BB8 RC Robot V4.0 [3] стоимостью 8555,37 руб.

Нам удалось собрать и запрограммировать прототип робота BB-8, управляемый с помощью приложения. В перспективе, он может использоваться как игрушка или элемент декора. Целевой аудиторией проекта являются семейные люди; энтузиасты, желающие подробнее изучить робототехнику; технологические компании, заинтересованные в массовом производстве продуктов в сфере робототехники.

#### СПИСОК ЛИТЕРАТУРЫ:

1. BB-8 - Wikipedia [электронный ресурс]: <https://en.wikipedia.org/wiki/BB-8> (дата обращения: 16.10.2022).
2. “What is Arduino?” [электронный ресурс]: <https://www.arduino.cc/en/Guide/Introduction> (дата обращения: 22.10.2022)
3. “Радиоуправляемый робот BB 8 2,4G с дистанционным управлением и звуковой экшн-фигуркой, обновленная интеллектуальная модель робота BB-8 Ball Droid” [электронный ресурс]: [https://aliexpress.ru/item/1005004410234546.html?sku\\_id=12000029084954734&feed\\_id=20&\\_randl\\_currency=RUB&\\_randl\\_shipto=RU&sellermenu\\_hide=true&src=googleweb&albch=dspl&aff\\_short\\_key=brxT3bLh&isdl=y&aff\\_platform=jvru&utm\\_medium=cpc&utm\\_source=google&utm\\_campaign=googleshopping](https://aliexpress.ru/item/1005004410234546.html?sku_id=12000029084954734&feed_id=20&_randl_currency=RUB&_randl_shipto=RU&sellermenu_hide=true&src=googleweb&albch=dspl&aff_short_key=brxT3bLh&isdl=y&aff_platform=jvru&utm_medium=cpc&utm_source=google&utm_campaign=googleshopping) (дата обращения: 01.11.2022)
4. Arduino Uno R3 [электронный ресурс]: <https://docs.arduino.cc/hardware/uno-rev3> (дата обращения: 18.11.2022)
5. Arduino L293D Motor Driver Shield Tutorial [электронный ресурс]: <https://electropeak.com/learn/use-arduino-l293d-motor-driver-shield-tutorial/> (дата обращения: 03.12.2022).
6. 120TT Gear Motor with Arduino [электронный ресурс]: <https://electropeak.com/learn/interfacing-single-axis-1-120-tt-gear-motor-with-arduino/> (дата обращения: 19.12.2022).
7. Bluetooth RC Car [электронный ресурс]: <https://play.google.com/store/apps/details?id=braulio.calle.bluetoothRCcontroller&hl=en&gl=US> (дата обращения: 05.01.2023).
8. Interfacing the HC-06 Bluetooth module with Arduino [электронный ресурс]: <https://create.arduino.cc/projecthub/RucksikaaR/interfacing-the-hc-06-bluetooth-module-with-arduino-f9c315> (дата обращения: 17.01.2023).
9. Bluetooth HC-06 [электронный ресурс]: <https://community.appinventor.mit.edu/t/bluetooth-hc-06-arduino-send-receive-send-text-file-multitouch-image/9518> (дата обращения: 13.02.2023).
10. Arduino Development Environment [электронный ресурс]: [https://arduino.ru.translate.goog/Arduino\\_environment?x\\_tr\\_sl=ru&x\\_tr\\_tl=en&x\\_tr\\_hl=en&x\\_tr\\_pto=sc](https://arduino.ru.translate.goog/Arduino_environment?x_tr_sl=ru&x_tr_tl=en&x_tr_hl=en&x_tr_pto=sc) (дата обращения: 01.03.2023).

11. Интернет-магазин Яндекс Маркет [электронный ресурс]:  
<https://market.yandex.ru/> (дата обращения: 07.03.2023).

## **ПЛАТФОРМА ДЛЯ РЕШЕНИЯ ОЛИМПИАДНЫХ ЗАДАЧ ПО ПРОГРАММИРОВАНИЮ**

*Иванов Артём, Ефимов Артём, Александров Илья*

*Муниципальное автономное образовательное учреждение лицей №7, 9 класс,  
г. Томск*

Руководитель: Козлова Елена Геннадьевна, учитель информатики

Актуальность. В последние несколько лет очень сильно видна тенденция роста IT-отрасли в мире, программисты сейчас очень востребованы на рынке. В начале прошлого года мы узнали о существовании платформы codeforces, на которой программисты со всего мира соревнуются в решении задач по спортивному программированию. Попробовав поучаствовать в нескольких состязаниях, мы выяснили, что задачи там для нашего уровня знаний достаточно сложные. Через некоторое время нам пришла в голову идея создать собственный сайт, на котором мы сможем сами составлять свои задания, и предоставить возможность всем интересующимся (в первую очередь школьникам) оттачивать свои умения в решении задач на уровне кода в соревновательном формате. Работа над этим проектом предоставила бы нам много опыта работы в команде.

Проект может помочь тем, кто уже знает какие-либо языки программирования писать более быстрые скрипты. С его помощью можно состязаться в оптимизации кода и решении нестандартных проблем. Эта платформа также была бы полезна для подготовки к олимпиадам по информатике.

Целевая аудитория – школьники с 5 по 11 класс, умеющие программировать, или изучающие язык программирования

Потенциальные заказчики – школы или компании, желающие провести своё соревнование по программированию

Цель: создание платформы с соревнованиями по спортивному программированию для школьников

Задачи:

- Изучить существующие платформы
- Выработать концепцию
- Создать платформу для состязаний в решении задач по программированию
- Испытать платформу на тестовых соревнованиях, выявить ошибки
- Установить план долгосрочного развития проекта

Аналоги платформы.

На сегодняшний день существует несколько аналогов к нашему проекту. Это сайт российского программиста Михаила Мирзаянова codeforces.com [1], экосистема All Cups от VK [2], платформа Сириус [3], на которой проводятся школьный этап всероссийской олимпиады школьников по информатике и

туры для поступления к ним на смены, связанные с информатикой, а также зарубежная платформа Codechef [4].

Codeforces – пожалуй, самый популярный сайт с соревнованиями по спортивному программированию, на нём примерно два раза в неделю проводятся сложные соревнования в которых участвует в среднем 20 тысяч человек, иногда могут проходить престижные турниры. На нём также устраивают муниципальный и региональный этапы ВСОШ. Единственный существенный недостаток платформы – большинство задач тяжелые, поэтому новичку там вряд ли удастся решить больше одной-двух задач (тем более тому, кто ещё не очень хорошо разбирается в программировании).

All Cups – платформа с интересным дизайном, на которой проходят все соревнования VK. Поэтому она закрытая, соревнования проходят редко. То же самое можно сказать и про Сириус.

Codechef – зарубежная платформа, во многом сравнимая с codeforces, но она полностью на английском языке.

Концепция платформы.

Изучив основные аналоги и добавив свои идеи, мы выработали концепцию нашего проекта. Она заключается в том, что на нашем сайте будут регулярно проводиться соревнования, каждое из которых ограничено по времени и состоит из нескольких задач, каждую из которых нужно решить, написав алгоритм на каком-либо языке программирования. Этот алгоритм проверяется на сервере на тестах с помощью компилятора. За полноту и быстроту решения начисляются баллы. Лучшие участники получают очки рейтинга, у худших они снимаются. Цель каждого пользователя – набрать наибольший рейтинг на сайте.

Реализация.

Используемые методы и средства: языки HTML, CSS, JavaScript для frontend-части сайта, Python + фреймворк django для backend-части, программа Figma для создания дизайна, макета сайта, github для обмена файлами проекта.

Распределение ролей в проекте: Александров Илья отвечает за дизайн сайта, создание текстовой части соревнований, помогает с компиляцией файлов на сервере, Иванов Артём верстает страницы сайта, связывает пользователя и сервера (frontend), Ефимов Артём делает backend-часть, тестирующую систему, систему рейтинга, модели и др.

Этапы выполнения проекта:

1. Разработка дизайна, макета сайта, моделей - с августа 2022 по сентябрь 2022. На этом этапе шло продумывание основных страниц, создание к ним дизайна в программе Figma и первая вёрстка. Параллельно изучались возможности django для тестирования и создания моделей.

2. Создание backend-части и frontend-части – с сентября 2022 по ноябрь 2022. Шло создание разных систем сайта: тестирующей, системы входа/регистрации, компиляции и другого, создавались и модели. Продолжались верстаться страницы сайта.

3. Соединение frontend-части сайта с backend-частью - с декабря 2022 по февраль 2023. К тому моменту были сверстаны основные страницы и готовы модели, первый вариант тестирующей системы. Требовалось объединить эти части в один проект.

4. Переписывание, доделывание кода - февраль 2023. Слишком нагромождённые части кода пришлось переписывать. Шло изменение иерархии элементов сайта и их сортировка, доделывание тестирующей системы и функции подведения итогов соревнования, добавление компиляторов под разные языки, починка всего, что ломалось.

5. Первое соревнование - февраль 2023. Были подготовлено первое соревнование, состоящее из пяти задач. Мы позвали своих знакомых, чтобы они поучаствовали. Вердикт - люди сказали, что у сайта хороший дизайн, крутая тестирующая система, интересные фишки, но наблюдались баги, были высказаны пожелания о недостающих функциях.

6. Развитие проекта - с марта 2023 по май 2023. Идёт адаптация сайта под разные форматы, проводятся соревнования, создаётся страница друзей, проверяются компиляторы и др. Происходят выступления на конференциях разного уровня. Ближе к концу этого этапа мы будем решать, как развивать проект дальше и в какую сторону двигаться.

#### Заключение

Подводя итоги, можно сказать, что мы выполнили поставленную цель – у нас получилось создать платформу с соревнованиями по спортивному программированию. Мы получили много опыта, развили навыки в промышленном программировании, работы в команде, кооперации, организации событий.

Перспективы. В дальнейшем мы будем развивать проект, адаптировать его под разные устройства, добавлять необходимые функции. В скором времени мы планируем провести соревнование на уровне школы и регулярно проводить новые соревнования, расширяя круг людей, в них участвующих. Бюджет проекта заключается в покупке домена для сайта и серверов, его обеспечивающих. На разработку самой платформы денег потрачено не было. В будущем мы можем устраивать соревнования для каких-либо IT-компаний в коммерческих целях или добавить монетизацию на сайт.

Ссылка со скриншотами проекта:  
<https://disk.yandex.ru/d/ksrtntZTFptTCA>.

#### СПИСОК ЛИТЕРАТУРЫ

1. <https://codeforces.com/> - платформа codeforces
2. <https://cups.online/ru/> - платформа All Cups
3. <https://edu.sirius.online/> - платформа Сириус
4. <https://www.codechef.com/> - платформа Codechef

**ТЕЛЕГРАММ БОТ ДЛЯ ИЗУЧЕНИЯ АНГЛИЙСКОГО ЯЗЫКА**  
*Колмогоров Степан Дмитриевич*

Руководитель: Осинцев Артем Викторович, педагог доп. образования, АНО  
ДО Технопарк "Кванториум".

Аннотация. В работе представлен прототип чат бота выполняющего роль помощника в изучении английского языка. Для прототипа, выбран Telegram, как один из самых популярных мессенджеров на территории России. Разработан алгоритм и реализован прототип чат бота, который позволит человеку учить слова английского языка в любое время.

Ключевые слова: чат, бот, английский, изучение.

Актуальность. Актуальность проблемы подтверждается тем, что в эпоху глобализации и расширения международных контактов изучение английского языка приобретает особую значимость. Имеется много причин, по которым люди изучают английский язык. Одна из них заключается в необходимости общаться письменно и устно с людьми, говорящими по-английски. Вы можете общаться с ними через Интернет или по почте. Английский совершенно необходим тем, кто работает в какой-либо области науки, т.к. является языком научных книг и журналов. Более общей целью изучения английского языка является стремление расширить свои интеллектуальные и культурные горизонты.

Разработка чат бота позволит современному человеку изучать английский язык в привычной обстановке – переписке.

Обзор аналогов. Существуют следующие способы изучения английского в интернете: 1) Универсальные сайты для изучения английского языка для начинающих, но они чаще всего бывают платными и могут подойти не для всех. 2) Просмотр видеороликов на английском скорее всего не подойдет для человека совсем не знающего языка. 3) Фильмы и сериалы с субтитрами также могут не подойти людям без каких либо знаний.

Основная часть. В ходе реализации проекта был создан чат бот для изучения английского языка. Бот направлен на людей, стремящихся расширить свои интеллектуальные и культурные горизонты. Разработан чат бот на языке python, использовался модуль на python telebot [1].

Ход работы бота:

1. Пользователь обращается к боту.
2. Задание вопросов пользователю о его знаниях в сфере английского языка.
3. Анализ ответов пользователя и определение наиболее подходящего уровня слов.
4. Чат бот начинает отправлять английские слова с переводом на русский, для их запоминания.
5. После отправки определенного количества слов (25) и изучения их пользователем, бот составляет тест на проверку знаний человека.

## СПИСОК ЛИТЕРАТУРЫ

1. TeleBot <https://github.com/xditya/TeleBot>
2. BotFather [Электронный ресурс]. 2023. Дата обновления: 05.01.2023. URL: <https://telegram.me/BotFather> (дата обращения: 05.01.2023).

### **ЧАТ-БОТ КАРЛ**

*Кондратьев Евгений*

*МБОУ «Тогурская СОШ им. С.В. Маслова», Колпашевского района Томской области, 10 класс*

Актуальность данного проекта заключается в том, что данный чат-бот предназначен для частичной замены собеседника и использование некоторых полезных функций в социальной сети Discord. Его главной возможностью является искусственная интерпретация собеседника с расширенными возможностями, командами и функциями.

Введение.

Бот — это программа, которая выполняет автоматические, повторяющиеся и заранее определенные задачи. Боты обычно имитируют человеческое поведение или подменяют его. Поскольку все их действия автоматизированы, они работают намного быстрее, чем обычные пользователи.

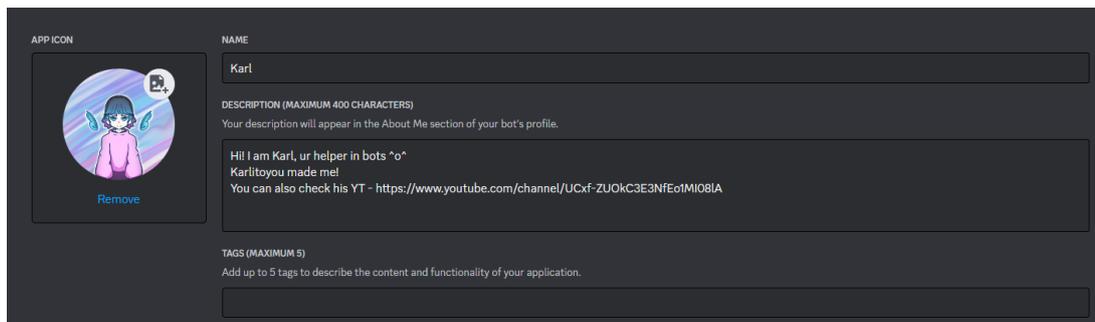
Боты очень часто используются для ответов на типовые вопросы, возникающие в различных сообществах. Они действуют как своего рода представители человека для выполнения различных функций. Основной особенностью такого бота является то, что он устраняет необходимость вручную отвечать каждому участнику и полностью автоматизирует данный процесс.

Цель исследования: создание чат-бота в Discord средствами языка программирования Python.

В соответствии с поставленной целью были сформулированы задачи:

- Сбор информации по документации языка программирования Python;
- Изучение документации и возможностей Discord API;
- Задание определенных функций и событий для чат-бота Discord;
- Добавление готового бота на сервер в Discord.
- Описание процесса работы над проектом.

Первым делом для того чтобы начать работу мне предстояло выбрать в какой программе я буду кодировать моего бота, и я остановился на Visual Studio Code. Довольно удобная программа для редактирования кода любого известного языка программирования. Дальше я приступил к созданию прототипа моего бота, то есть основы с которых я буду брать идею для дальнейших работ с ними. После воссоздания прототипа я приступил к созданию профиля бота в Discord. Это можно сделать на сайте «Discord developer portal»



После того как я создал бота, я пригласил его на свой сервер, перейдя по ссылке приглашения бота. Дальше я начал высматривать варианты, как его можно запустить в Discord. Остановившись на одном из них, я выбрал вариант виртуальной среды. Запустив виртуальную среду через терминал в программе VSC:

```
#python -m venv venv
```

Далее я скачал библиотеку «discord.py» через командную строку.

```
#pip install discord.py
```

Столкнулся я правда с одной проблемой, когда хотел установить эти библиотеки. Оказывается, что исполнитель «pip» в моем случае не хотел работать потому что командная строка не могла его найти, для этого мне пришлось внести кое-какие изменения в моем компьютере, после чего я успешно установил их.

После установленных библиотек я принялся за код, импортировал внутри кода все необходимые мне библиотеки, и принялся за работу.

Первым делом что требовалось написать, это подтверждение токена бота, для того чтобы авторизовать его как бота в Discord. И также мне понадобился префикс бота, для определения команд. Токен и префикс я загрузил через словарь, но уже на другом языке программирования – Java Script.

```
{
  "token": "0TU00TU1MDk40Dg5NTI3Mjk3.GPR_Kw.wPOZYcDnI0s5ASpi8TRbbEF3rEqh6M4xkQ3lC0",
  "prefix": "!"
}
```

После всех этих действий, в главном коде я подтвердил токен и префикс бота через вот этот код:

```
intents = discord.Intents.all()

file = open('config.json', 'r')
config = json.load(file)

bot = commands.Bot(config['prefix'], intents=intents)
```

В заключение я принялся писать простейшие команды для бота, такие как «random» и «say», сделал я их для наглядности.

Для демонстрации приложу файлы с кодом данных команд:

```

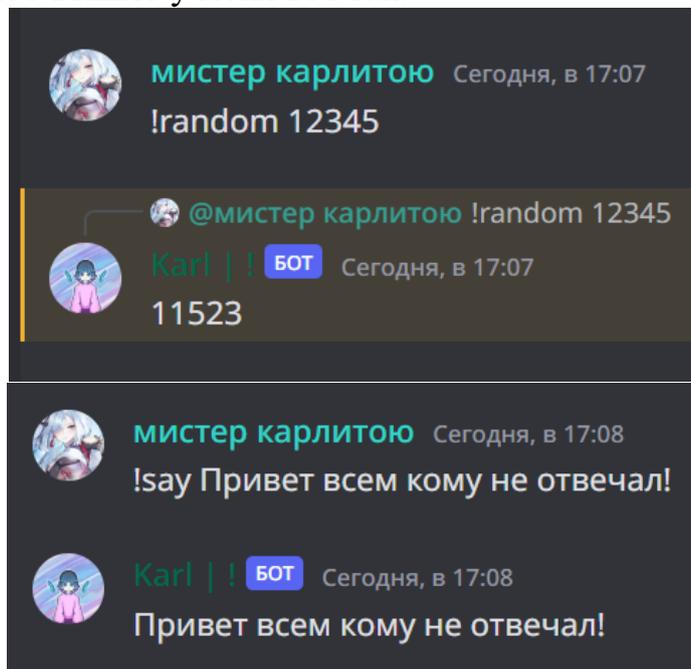
@bot.command(name= 'say', help= 'Эта команда выводит написанный тобой текст.', args=('Ваш текст'))
async def say(ctx, *args):
    say_list = str()
    for s in range (len(args)):
        say_list = say_list + ' ' + args[s]
    await ctx.send(say_list)

@bot.command(name= 'random', help= 'Эта команда выводит число в заданном диапазоне.', arg=('Число диапазона'))
async def random(ctx, numrandom: int, *arg):
    num3 = 0
    for num2 in range (len(arg)):
        num3+=arg[num2]
    if num3 == int():
        num1=str(randint(1,numrandom))
        await ctx.reply(num1)
    else:
        await ctx.reply('Ошибка')

```

Также в этих командах я сталкивался с проблемой, такой как кортежи. Аргументы, вписываемые людьми в команде, переводились в кортежи, что являлось главной моей проблемой, поэтому я решил перебрать аргумент с помощью цикла и у меня всё получилось.

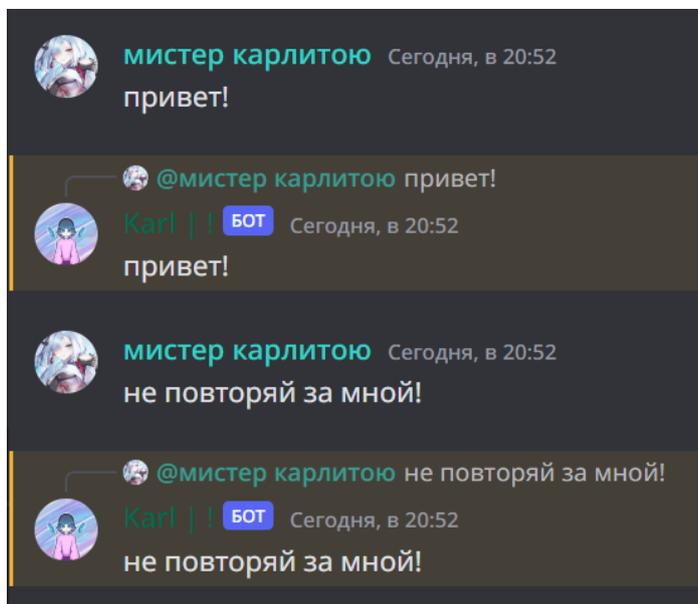
Но в итоге всего вышло у меня вот так:



Для данного проекта был выбран язык программированию Python, JS.

Также он имеет несколько команд, полезных и развлекательных, которые могут помочь тебе. Пока что в нём есть не все функции, которые я планирую ему сделать, но планируется минимум 20 команд, и более 500 различных вариантов ответа на ваши вопросы! Правда в данный момент у чат-бота есть только 5 команд и всего лишь 30 реплик, но с учетом времени, достичь цели мне удастся. В своем коде я использовать Библиотеку «Discord.py» для того чтобы синхронизировать моего бота в Python с Discord. Также мне понадобилась библиотека «json» для того чтобы подтвердить токен бота и его префикс. Ну и как же можно обойтись без вспомогательных библиотек random, time, math и asyncio. Я устанавливал библиотеку «Discord.py» через командную строку.

```
# Применял «pip install discord.py»
Путь решения моей проблемы был таков:
Синхронизировал токен и префикс бота через библиотеки json.
Взялся первым делом за простого «эхо-бот», для того чтобы проверить
работает от или нет.
Задал боту событие, по которому он прочтёт мое сообщение:
@bot.event
Асинхронная функция на сообщение отправителя:
async def on_message(ctx):
    Задаем условие чтобы бот не отвечал боту, иначе они будут бесконечно
повторять друг за другом:
    if ctx.author != bot.user:
        В итоге наш бот должен отправить сообщение отправителя:
        await ctx.reply(ctx.content)
И готово!
```



Но главный вопрос всё же остается не ответенным, для кого этот бот? Для людей кто хочет просто поговорить или же для тех, кто хочет повеселиться? Ответ на этот вопрос можете дать вы сами, или же Карл? Кто знает, пока не попробуете, не узнаете! В любом случае он нужен тем, кто нуждается в нём. На этом хочу подвести итог и в заключение сказать, что Карл, бот не простой, а довольно широко применяемый и многофункциональный, и я ещё не закончил его, ведь мы созданы для того, чтобы доводить всё до идеала!

## **РАЗРАБОТКА ИГРОВОГО ПРИЛОЖЕНИЯ ДЛЯ ПОРТАТИВНОЙ КОНСОЛИ НА ОСНОВЕ RASPBERRY PI**

*Косенчук Максим, Дроздов Максим*

*Муниципальное бюджетное общеобразовательное учреждение лицей при  
ТПУ, 10 класс  
г. Томск*

Руководитель: Мыцко Евгений Алексеевич, доцент ОИТ

Игровая деятельность в современное время играет важную роль в жизни людей, и все больше и больше компаний производит разные игровые устройства. Многие компании, понимая нужды рынка, создают разных размеров устройства, чтобы человек мог погрузиться в виртуальный мир и обучиться программированию в интересном формате.

Цель работы: Разработка собственного простого игрового приложения для портативной консоли, собранной на основе микрокомпьютера Raspberry pi 4 B.

Задачи:

- Провести обзор существующих операционных систем, эмуляторов, платформ для запуска игровых приложений на Raspberry pi;
- Установить и настроить операционную систему на Raspberry pi;
- Осуществить сборку портативного контроллера «Game NAT» и его настройку в операционной системе.
- Определить тематику и платформу игрового приложения;
- Провести обзор инструментов, средств, технологий и языков программирования для разработки игрового приложения;
- Реализовать игровой дизайн, меню и элементы игры;
- Разработать алгоритмы игрового процесса;
- Реализовать алгоритмы игрового процесса с применением выбранных средств и технологий;
- Осуществить интеграцию игрового приложения в соответствующую платформу.

*Обзор существующих операционных систем, эмуляторов, платформ для запуска игровых приложений на raspberry pi.*

Raspberry pi достаточно мощный портативный компьютер, который можно использовать для работы с различными задачами.

Технические характеристики Raspberry Pi 4 B:

- Broadcom BCM2711 SoC;
- 64-битный четырехъядерный ARMv8 Cortex-A72 процессор с тактовой частотой 1.5 ГГц;
- Графический сопроцессор VideoCore VI®;
- Память на 1/2/4/8ГБ LPDDR4 SDRAM;
- Gigabit Ethernet;
- USB3.0;
- 2 x micro-HDMI;
- 4 ГГц и 5 ГГц IEEE 802.11.b/g/n/ac WI-FI + Bluetooth 5.0 Low Energy (BLE).

Существует много операционных систем для их решения. В зависимости от типа задачи можно выбрать соответствующую операционную систему.

Системы, которые можно выбрать в установщике, делятся на категории:

- общего назначения (Raspbian OS, Ubuntu, Apertis, e.t.c)

- для работы с медиа и видео ресурсами (LibreELEC, OSMC, Volumio)
- для работы с эмуляторами и играми (Recalbox, RetroPie, Lakka)
- другие специального назначения (3D printing, Anthias и другие)

Нами была выбрана система Raspbian OS, потому что это официальная система Raspberry Pi, в ней есть все программные обеспечения, которые понадобятся для каждой основной задачи с компьютером. Она проста в освоении и имеет приятный интерфейс.

*Установка и настройка операционной системы на raspberry pi.*

Загрузите и установите Raspberry Pi Imager на компьютер с устройством чтения SD-карт. Вставьте SD-карту, которую вы будете использовать с Raspberry Pi, в устройство чтения и запустите Raspberry Pi Imager. (рис. 1)

Как только файл изображения будет записан на SD-карту, вы готовы к работе. Просто вставьте карту в свой Raspberry Pi и загрузите ее. (рис. 2)



Рисунок 1.

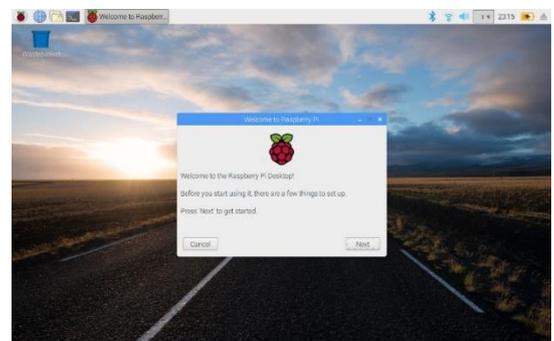


Рисунок 2.

*Установка и настройка контроллера game hat.*

Game Hat — это плата расширения, которая позволит вам превратить Raspberry Pi в портативную игровую консоль.



Рисунок 3.



Рисунок 4.

Сборка осуществляется из платы Game Hat и Raspberry Pi (рис. 3). В разъем GPIO на плате подключается микрокомпьютер и переходник между экраном и контроллером, также в батарейный отсек помещается аккумулятор и сверху вся эта конструкция закрывается акриловым корпусом и крепится шурупами. (рис. 4)

Экран в этой конструкции подключается по HDMI кабелю, а кнопки и джойстик GPIO выводом.

#### *Разработка игрового приложения и интеграция его в систему.*

В процессе работы было создано несколько игровых приложений, одно из них игра Flappy bird, в которой игрок путем нажатия на экран должен следить за полетом птицы, чтобы она летала между рядами зелёных труб, не задевая их. Данная игра создавалась на языке программирования python, с помощью библиотеки pygame. Pygame — набор модулей языка программирования Python, предназначенный для написания компьютерных игр и мультимедиа-приложений. Pygame базируется на мультимедийной библиотеке SDL.

Процесс создания игры начинается с импортирования модуля pygame и отрисовки экрана, в котором после будет отображаться наша игра. Следующим шагом создаем основной цикл игры, для ее открытия и закрытия. После этого, создаем “хитбокс” (область персонажа, которая срабатывает во время столкновения) птички и задаем ей правила полета. Далее отрисовываем трубы в виде прямоугольников одинакового размера и прописываем последствия после столкновения с ними, а также с краями экрана. Создав основной каркас игры мы загружаем, заранее подготовленные текстуры, такие как: задний план, модель птички и трубы. Следующим шагом импортируем библиотеку random и заставляем наши трубы отрисовываться в хаотичном порядке. После этого, создаем очки и жизни птички и заставляем птичку менять ускорения в зависимости от набранных очков, а также добавляем в проект фоновую музыку и звуки столкновения с преградами.

#### *Заключение*

В результате работы над проектом мы достигли цели — разработали свое игровое приложение для портативной консоли, все поставленные нами задачи были достигнуты. В процессе работы над проектом мы научились работать в команде и грамотно распределять между собой обязанности, также улучшили свои навыки программирования, освоили новые библиотеки и ООП (объектно-ориентированное программирование).

Бюджет проекта:

Наименование	Цена
Контроллер Game Hat	3790.00 Р
Raspberry Pi 4 model B	17,999.00 Р
Суммарная стоимость	
Полная стоимость всех комплектующих:	21,789.00 Р

#### СПИСОК ЛИТЕРАТУРЫ

1. Raspberry Pi 4 Model B [Электронный ресурс] - Режим доступа: <https://micro-pi.ru/raspberry-pi-4-model-b-rpi-4-b-bcm2711/> (дата обращения: 11.1.2022)

2. Raspberry Pi OS [Электронный ресурс] - Режим доступа: <https://www.raspberrypi.com/software/> (дата обращения: 12.1.2022)
3. Г. Халфакри. Raspberry Pi. Официальное руководство для начинающих / пер. с англ. С. В. Черникова – М.: ДМК Пресс, 2021. – 262 с.: ил.
4. Создание игр на Python 3 и Pygame [Электронный ресурс] - Режим доступа: <https://habr.com/ru/post/347138/> (дата обращения: 12.1.2022)
5. Основы PyGame [Электронный ресурс] - Режим доступа: <https://python-course.readthedocs.io/projects/elementary/en/latest/lessons/18-pygame.html#id1> (дата обращения: 12.1.2022)

## **ПРИЛОЖЕНИЕ В ПОМОЩЬ ВРАЧАМ НЕЙРОФИЗИОЛОГАМ**

*Лазаренко Ефим*

*Муниципальное общеобразовательное учреждение «Лицей №1», 11 ИТ класс*

*Г. Ачинск*

Руководитель: Попова Лидия Владимировна, учитель информатики

Цель работы: создание приложения на основе высокоуровневого языка программирования Python с использованием нейросети для анализа и тренировки лицевых эмоций людей, перенесших инсульт.

Актуальность: эмоции оказывают большое влияние на жизнь и деятельность человека, на протекание всех психических процессов, проявление свойств. Эмоции — это психические процессы, протекающие в форме непосредственных переживаний каких-либо жизненных событий и отражающие личную значимость и оценку внешних и внутренних ситуаций для жизнедеятельности человека. Эмоции имеют множество функций, но их главная функция состоит в том, что благодаря эмоциям мы лучше понимаем друг друга, можем, не пользуясь речью, судить о состояниях друг друга и лучше настроиться на совместную деятельность и общение. Но к сожалению, не все люди способны выражать эмоции. Это происходит вследствие серьезных болезней. Инсульт - самый страшный враг человеческих эмоций. У некоторых людей, перенесших инсульт, атрофируются мышцы лица, также часто происходит лицевая асимметрия, соответственно у них пропадает способность говорить, выражать эмоции. Как говорил Зигмунд Фрейд “К несчастью, подавленные эмоции не умирают. Их заставили замолчать. И они изнутри продолжают влиять на человека”. Ученые всего мира активно исследуют возможности применения нейросетей в медицине. Так работа Сергеева Ю.А., Стерлёвой Е.А., Ниязиной Д.А., опубликованная в научно-образовательном журнале для студентов и преподавателей “StudNet” в 2021 году, посвящена сравнению методов нейросетевого и группового анализа патологий. Авторы приходят к выводу что применение нейросетей дает возможность эффективно решать задачи прогнозирования динамики патологий. [1] Разработка специального цифрового тренажера для анализа эмоций пациента должна облегчить реабилитацию после инсульта.

Гипотеза: можно разработать кроссплатформенное приложение для врачей, используя высокоуровневый язык разработки Python.

Постановка и формулировка проблемы: восстановление мимики лица после перенесенного инсульта.

Основные задачи:

- Анализ проблемы (беседа с заказчиком проекта с “Фабрики программирования”).
- Анализ возможностей нейросетей с открытым кодом.
- Разработка и апробация модуля распознавания эмоций.
- Создание базы медиа-материала для приложения.
- Публикация приложения, последующие разговоры с заказчиком.
- Методы решения: анализ, беседа, измерение, математический метод, описание, практическое моделирование.
- Заказчики, стейкхолдеры: Институт нейрофизиологии, Медицинские учреждения.

Полученные результаты: разработано приложение, способное считывать эмоции, проводить тренировку с помощью показа медиа-материала, а также записывать результаты сеансов в отдельный файл. Также присутствует User Interface – интерфейс для более простой работы с приложением.

На сегодняшний день перед медициной стоит множество проблем и вопросов, которые необходимо решать. Так как мы живем в веке, когда все действия и процессы цифровизируются, современным технологиям необходимо активно внедряться в систему здравоохранения. [2]

Одним из основных направлений развития цифровой индустрии является создание нейросетей, которые должны облегчать жизнь человеку. Использование нейронных сетей опирается на накопленный опыт в виде данных и подходит для решения задач, с которыми человечество уже знакомо. [3] Нейросеть в медицине могут стать отличным помощником врача (но ни в коем случае его заменой).

Мы решили рассмотреть проблему людей, перенесших инсульт. После такой ужасной болезни, лицевые мышцы могут атрофироваться, и выражение эмоций становится затруднительным.

Наше приложение — это помощник врача. Оно поможет врачу записывать эмоциональный фон пациента, изменения в нем, а также тренировать мышцы лица пациента. Приложение основывается на нейросети, способной в точности распознавать эмоции из заданного списка, выявлять ее коэффициент (от 0 до 1).

#### *Нейросеть, используемая в приложении.*

Существует множество видов нейросетей. Для нашего проекта мы решили использовать сверточную нейронную сеть (convolutional neural networks, CNN). Для работы с нейросетью использовался высокоуровневый язык программирования Python, среда разработки PyCharm и такие библиотеки как: TensorFlow, OpenCV, NumPy, Datetime, PyQt5 и др.; виртуальное окружение – Venv. Сама модель нейросети была взята на GitHub. Обучение нейросети

проходило по подобранному нами датасету. Название датасета: FER–2013. Данные в нем состоят из изображений лиц в градациях серого размером 48x48 пикселей. Лица были автоматически зарегистрированы, так что лицо находится более или менее по центру и занимает примерно одинаковое количество места на каждом изображении. Задача состоит в том, чтобы классифицировать каждое лицо на основе эмоций, отображаемых в выражении лица, в одну из семи категорий (0 = злость, 1 = отвращение, 2 = страх, 3 = радость, 4 = грусть, 5 = удивление, 6 = нейтрально). Обучающий набор состоит из 28709 примеров, а общедоступный тестовый набор состоит из 3 589 примеров. Модель обучалась на 48 эпохах (epochs) при 64 партиях (batches). Для обработки слоев использовалась модель Sequential() – стек слоев, где каждый слой имеет ровно один входной тензор и один выходной тензор.

#### *Тестирование работы программы.*

Для того чтобы сделать выводы, необходимо провести тестирование работы приложения.

Приложение сейчас имеет следующие возможности:

1. Отслеживание эмоций в реальном времени
2. Запись эмоций, коэффициентов, результатов отслеживания в отдельный файл
3. Тренировочный режим: показ медиа–файлов для повторения эмоций
4. Имеет интерфейс

Заключение

Результаты: в заключение, хотелось бы отметить, что диапазон использования нейросетей очень обширен и те возможности нейросетей, которые нам кажутся на первый взгляд развлечением, могут помочь в решении серьезных проблем. Наша гипотеза подтвердилась: можно разработать кроссплатформенное приложение для врачей, используя высокоуровневый язык разработки Python. Результаты по задачам: проанализирована проблема (проведена беседа с заказчиком проекта с “Фабрики программирования”), проанализированы возможности нейросетей с открытым кодом. разработан и апробирован модуль распознавания эмоций, создана база медиа-материала для приложения, приложение опубликовано на ресурсе GitHUB (см. приложение №1) .

Выводы: используя высокоуровневый язык разработки Python, нам удалось разработать кроссплатформенное приложение для врачей нейрофизиологов “Эмпа” на основе нейросети для распознавания эмоций. Данное приложение имеет дружелюбный интерфейс, который позволяет врачу провести анализ интенсивности мимики пациента, запустить режим тренажера, а также сохранить данные о различных сеансах. В перспективе развития последующие посещения «Фабрики программирования» для доработки приложения, связи с реальным заказчиком.

#### *Приложение 1*

Ссылка на проект “Эмпа” в GitHUB: <https://github.com/efim1222/emotions.git>



*Рисунок 1. GitHUB проекта «Эмна»*

### СПИСОК ЛИТЕРАТУРЫ

1. Булгакова С.В. “Новая личность и нейрокоммуникации: нейрогенетика и нейросети, психо–нейроиммуноэндокринология, 5p медицина и 5g технологии”, журнал “Бюллетень науки и практики”, 2021
2. Волчек Ю.А. “Положение модели искусственной нейронной сети в медицинских экспертных системах”, журнал “Juvenis scientia”, 2017
3. Чупакова А.О. “Разработка и обучение модели искусственной нейронной сети для создания систем поддержки принятия решений”, журнал “Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика”, 2020

### УМНАЯ КОПИЛКА

*Бондарев Максим, Матюшин Александр, Сидоров Дмитрий  
государственное бюджетное общеобразовательное учреждение Самарской  
области средняя общеобразовательная школа пос. Кинельский  
муниципального района Кинельский Самарской области, 8 класс  
пос. Кинельский*

Руководитель: Вдовина Ксения Викторовна,  
учитель математики и информатики

Обоснование актуальности и практической значимости выбранной темы.

Как известно, рациональное распределение денежных средств составляет основу финансовой грамотности школьников. Развитию этого навыка способствует осуществляемый контроль-учёт транзакций. Это достигается благодаря включению в деятельность соответствующих заданий [2].

Возникает вопрос – как осуществлять процесс транзакции, сделав его максимально эффективным, и суметь накопить «на мечту»?

Именно поэтому актуальность выбранной темы исследования заключается в разработке и демонстрации мобильного приложения со свободно конвертируемым кодом для различных цифровых устройств, предназначенное для контроля доходов и расходов денежных поступлений.

Цель: Повышение уровня мотивации и заинтересованности в контроле учёте денежных средств с помощью разработанного мобильного приложения, функционал которого направлен на личное финансовое планирование в единстве с формированием финансовой грамотности.

Гипотеза исследования: реализация деятельности по разработке и распространению авторского мобильного приложения с применением математических вычислений способствует эффективному формированию финансовой грамотности.

В соответствии с целью и гипотезой исследования, были сформулированы следующие задачи исследования:

- изучить необходимую литературу в соответствии с обозначенной проблемой, выбрать соответствующий инструментарий для исследования;
- определить экспериментальную группу обучающихся-участников исследования [2];
- провести исследование уровня сформированности навыка проведения транзакции;
- зафиксировать полученные результаты, провести их аналитическое обоснование (посредством наглядных форм представления промежуточных результатов);
- составить эскиз (схему) приложения, направленного на решение выявленных проблем;
- разработать листинг (программный код), позволяющий производить подсчёт баллов и отражающий основную идею эскиза;
- провести апробацию полученного приложения.

Обзор аналогов:

- «Сбербанк-Online» - многофункциональное мобильное приложение, ориентированное на персонифицированный выбор опций пользователем;
- «СберKids» - приложение, в котором ребёнок может самостоятельно управлять своими денежными средствами, повышая финансовую грамотность;
- MirPay – специализированное мобильное приложение для бесконтактного проведения транзакций смартфоном;
- ЮMoney – платёжный Online-сервис электронных платежей.

Целевая аудитория проекта: целевая аудитория – обучающиеся общеобразовательных учреждений, а также иные лица старше 12 лет.

Используемые материалы, методы и оборудование:

- ноутбук с Интернет-доступом;
- виртуальная среда разработки мобильных приложений (MitApp Inventor);

- USB-шнур;
- мобильное устройство на базе OS Android.

Представление плана проектных действий (этапы выполнения проекта).

I этап. Анкетирование респондентов: на данном этапе были выявлены реперные точки по созданию эскиза продукта, основанные на мнении и предпочтении потенциальных пользователей (Вопрос №5 Online-анкеты):

<p><b>Вопрос №5.</b>  <b>Как повысить навык распределения доходов и расходов?</b></p>
<p>Фиксировать доходы и расходы</p>
<p>Запоминать основные моменты транзакции</p>
<p>Планировать покупки заранее</p>
<p>Вести блокнот расходов и доходов</p>

II этап. Разработка эскиза продукта, основанная на функциональных возможностях среды программирования;

III этап. Непосредственная вёрстка дизайна продукта, работа в режиме «Дизайнер»;

IV этап. Программирование основных компонентов каждого экрана продукта, основанное на принципе инкапсуляции свойств:

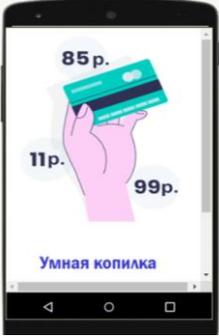
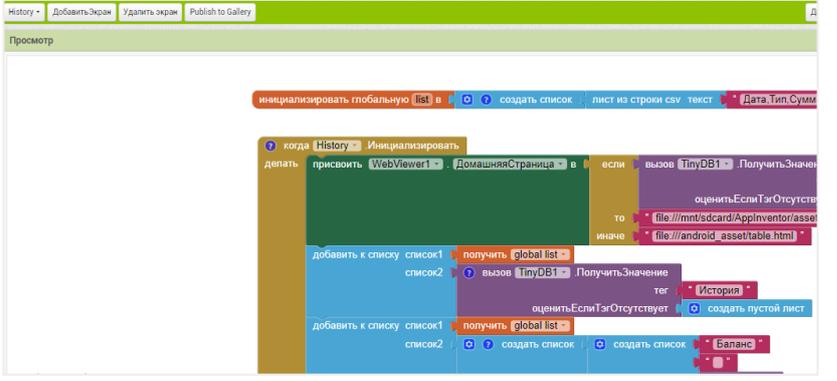
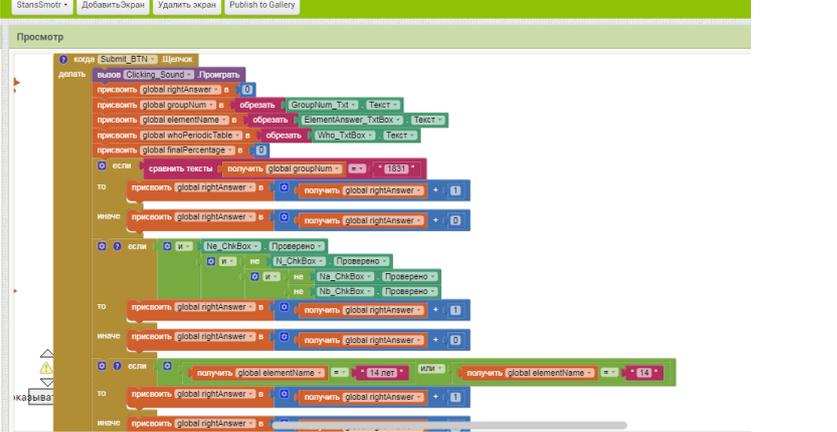
Экран приложения	Программные блоки
	
	



Рисунок 1. Интерфейс мобильного приложения в режиме «Дизайнер» и «Блоки»

V этап. Тестирование и отладка промежуточной версии продукта-авторского мобильного приложения: корректировка программного кода, введение глобальных переменных в структуру листинга;

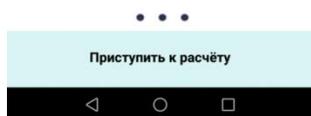
VI этап. Построение .apk-версии приложения, повторное тестирование полученной версии;

VII этап. Распространение среди ученической общественности полученного продукта, сбор отзывов и предложений по результатам апробации мобильного приложения «Умная копилка».

Представление результата – проектного продукта в виде фото, макета, прототипа



### Умная копилка



#### Сведения о транзакции

Дата покупки

Расход

Количество



Рисунок 2. Интерфейс итоговой версии мобильного приложения

### Заключение

Полученный продукт-мобильное приложение «Умная копилка» - было разработано с учётом недостатков, выявленных в процессе детального обзора аналогов и представляет собой авторское решение команды обучающихся 8 класса.

Благодаря доступному интерфейсу и автоматическим отчётам по транзакциям ребёнок может приучить себя к самоконтролю своих средств, что позволит избежать проблем в будущем [2].

Навигация приложения довольно проста: выбирается тип транзакции, которую пользователь собирается осуществить, а затем-категория, по которой планируется провести оплату (приступить к накоплению).

Итогом проведённой транзакции является протокол, отражающий примерную дату расходов (проведения накоплений) с детализацией основных категорий расходных статей («Оплата за коммунальные услуги», «Планируемое приобретение бытовой техники» и т.п.).

Данный проект также имеет перспективу в будущем. Благодаря тому, что его содержательный компонент соответствует программе внеурочной деятельности по общеинтеллектуальному направлению, приложение можно использовать в учебном процессе [2].

Примечательно, что приложение не оказывает негативного воздействия на временные и материальные ресурсы, не требует подключения к сети Интернет.

Важно отметить, что гипотеза исследования действительно была подтверждена: реализация деятельности по разработке и распространению авторского мобильного приложения способствует эффективному формированию финансовой грамотности у обучающихся: это подтверждается приобретением навыка рационального распределения денежных средств.

Безусловно, в будущем планируется изменять характеристики данного приложения, улучшая его параметры и изменяя дизайн.

Надеемся, что приложение принесёт пользу в образовательный процесс, ведь оно экономит материальные ресурсы (печать материалов) и также существенно снижает временные затраты, поскольку проведение контроль-учёта денежных средств на гаджетах намного ближе современным школьникам, чем традиционное оформление «сметы расходов» [1].

Выводы и предложения по внедрению результатов (перспективы использования, потенциальные заказчики и потребители).

Благодаря тому, что разработанный интерфейс мобильного приложения идентичен популярным версиям его прототипов, оно может быть использовано в качестве альтернативной версии для контроль-учёта денежных средств различными предприятиями (например, в режиме более масштабной апробации).

## СПИСОК ЛИТЕРАТУРЫ

1. Нагибин Ф.Ф., Канин Е.С. Математическая шкатулка: Пособие для учащихся. – М., 2016.
2. Хуторской А.В. Методика личностно-ориентированного обучения. Как обучать всех по-разному? Пособие для учителя. – М.: Изд-во ВЛАДОС-ПРЕСС, 2005. – 383 с. - (Педагогическая мастерская).

## КОМФОРТНЫЙ ДОМ

*Осипов Максим*

*Муниципальное бюджетное общеобразовательное учреждение «Лицей №2»*

*муниципального образования г. Братска*

*9 класс*

*г. Братск*

Руководитель: Харина Наталья Павловна, учитель математики

## Введение

В мире количество людей постоянно растет, на данный момент это число составляет уже порядка 8.9 млрд. человек. Каждому человеку нужна крыша над головой и самым хорошим вариантом остаются квартиры, но подбор квартиры очень сложный и долгий процесс. Где-то вам не нравится планировка, где-то цена, а где-то рядом живут очень шумные соседи. Одним из главных составляющих выбора квартиры является ее планировка: расположение комнат, коридоров, ванны и других необходимых частей. Для подбора удобной планировки вам нужно рассматривать различные варианты и искать подходящий, ездить по городу осматривать их. Это все занимает огромное количество времени и сил. Именно поэтому эта проблема заинтересовала меня.

### Актуальность:

Я провел социальный опрос, который поможет определить актуальность темы более подробно. В опросе участвовало 34 взрослых человека, которые сталкивались с проблемой выбора квартиры. Как показал опрос, для большинства планировка важна (диаграмма 5), время поиска подходящей квартиры составляло более 3 месяцев (диаграмма 4) и поиск квартиры проходил под взором риэлтора (диаграмма 3). Следовательно, для большинства поиск квартиры долгий, но важный процесс, что подтвердил социальный опрос (диаграмма 2).

Цель работы: Разработать программу для поиска подходящих планировок квартир по имеющемуся плану, чертежам или обычным рисункам.

### Задачи:

- Изучить работу с компьютерным зрением и сравнение изображений с использованием языка Python.
- Собрать информацию о важности планировки квартиры при ее выборе.
- Изучить работу с большим массивом изображений.
- Создать программу

В своей работе я буду использовать язык программирования Python по нескольким причинам:

- Данный язык программирования я хорошо знаю и изучаю уже 2 год.
- Вокруг создания нейронных сетей на Python собралось огромное количество полезной для моего проекта информации, которую я собираюсь использовать. Также в Python есть огромное количество библиотек для работы с нейронными сетями, которые помогут мне.

• Низкий порог входа. Хотя я довольно долго знаю Python, но с нейронными сетями буду работать в первый раз, поэтому это очень важно для меня. Язык предоставляет возможность создание сложных нейронных сетей, очень просто и быстро.

• Нейронные сети на Python, очень гибкие и я могу их постоянно изменять, создавать вариант, который будет для меня лучше.

• Язык Python по всем параметрам очень хорошо подходит для написания проекта человеком ни разу не работавшим с нейронными сетями. Также в

своей работе я буду использовать среду разработки PyCharm. Данная среда разработки самая удобная и практичная из всех, что я пробовал.

- Мое приложение будет работать как поисковик схожих картинок. Оно будет брать картинку или эскиз пользователя, и искать похожий из базы данных планировок. Программа будет написана на языке программирования Python с использованием различных библиотек, ускоряющих работу приложения.

Вот некоторые из них:

- Keras – это открытая библиотека, написанная на языке Python и обеспечивающая взаимодействие с искусственными нейронными сетями. Данная библиотека будет нужна для работы с нейронными сетями, которые и будут работать с картинками и искать похожие.

- TensorFlow – это открытая программная библиотека для машинного обучения, разработанная компанией Google для решения задач построения и тренировки нейронной сети с целью автоматического нахождения и классификации образов. Нужна будет для обучения нейронной сети.

- NumPy - это открытая библиотека для Python, который предоставляет общие математические и числовые операции в виде готовых, быстрых функций. Нужна будет для ускорения приложения.

- Scikit-image – это открытая библиотека для обработки изображений написанная на языке Python. Она включает в себя алгоритмы сегментации, геометрических преобразований, манипуляций с цветом. Нужна будет для обработки изображений, чтобы нейронная сеть могла их использовать.

- OpenCV - это открытая библиотека компьютерного зрения, которая предназначена для анализа, классификации и обработки изображений. Нужна будет для того чтобы нейронная сеть смогла видеть, что находится на картинке.

- Pandas – это программная библиотека на языке Python для обработки и анализа данных. Работа Pandas с данными строится поверх библиотеки NumPy. Предоставляет специальные структуры данных и операции для манипулирования числовыми таблицами и временными рядами. Нужна будет для более быстрой и удобной работе с приложением.

- Tkinter – это кросс-платформенная событийно-ориентированная графическая библиотека, с помощью которой будет реализован GUI-интерфейс (Графический интерфейс взаимодействия с пользователем) для работы в приложении.

- Также будет использоваться еще некоторое количество других библиотек, не попавших в этот список.

Создание программы проходит в два этапа:

1. Создание поиска схожих картинок.
2. Построение приложения вокруг программы поиска.

Основная функция моего приложения, то есть поиск схожих картинок, будет реализована с помощью создания специального типа нейронной сети, называемой автоэнкодер.

Изображения в этом типе не будут дополняться различными метками, по которым можно было бы их сравнивать. Извлечение нужных для сравнения признаков из картинки будет происходить с помощью визуального содержимого (текстуры, формы и другие). Такой тип поиска изображений называется поиск изображения на основе содержимого (CBIR), в отличие от поиска ключевых слов или изображений на основе текста.

Любую нейронную сеть нужно обучить, чтобы она могла выполнять какие-либо действия, так и в нашем случае ее нужно будет обучить. Автоэнкодер используется для преобразования изображения в векторное представление (то есть в виде точек и линий). После этого, будет вычисляться расстояние между этим векторами и чем меньше оно будет, тем более похожими будут картинки.

Весь алгоритм будет выглядеть вот так:

- Пользователем загружается изображение планировки квартиры, которую он нарисовал.
- Изображение преобразуется в векторную форму.
- Извлекаем важные для сравнения признаки и преобразуем их в вектора.
- Преобразованные вектора сравниваются с набором других векторов и находятся похожие изображение на основе расстояние между ними, чем меньше, тем более похожи планировки между собой.
- Самая похожая планировка будет отображаться пользователю вместе со ссылкой на какой-либо ресурс по продаже недвижимости.

Приложение, построенное вокруг определения, будет очень простым.

Состоять приложение будет из (рис 1.):

- Две картинки. Одна для демонстрации рисунка пользователя, одна для демонстрации похожей планировки.
- Текстовое поле. Для вывода ссылки на сайт с планировкой
- Кнопка для изменения рисунка пользователя

Заключение

Приложение по поиску планировок готово, оно выполняет свою функцию. Видео, в котором показан способ работы приложения размещено по ссылке: <https://cloud.mail.ru/public/uiDG/VJxppJvLr>.

Основной вектор развития дальше - это обучение нейронной сети, для лучшего поиска планировок. Обучить нейронную сеть можно с помощью большего объема информации, то есть большего количества изображений планировок квартир. Большее количество планировок квартир можно получить обращаясь к планировкам других городов. Также можно добавить функцию обращения к застройщикам со своей планировкой, если вы не нашли подходящего варианта. Динамическое изменение базы данных планировок, сейчас программа работает только с внедренными планировками, а это позволит не предлагать уже проданную квартиру.

## СПИСОК ЛИТЕРАТУРЫ

1. Франсуа Шолле Deep Learning with Python 2018г.
2. Hands-On Machine Learning with Scikit-Learn and TensorFlow 2018г.

Электронные ресурсы – дата обращения 20.01.2023

3. Как найти и сравнить похожие изображения автоэнкодером статья: <https://habr.com/ru/post/703796/>
4. Сверточный автоэнкодер статья: <https://pgaleone.eu/neural-networks/2016/11/24/convolutional-autoencoders/>
5. Сверточная нейронная сеть: структура, топология, функции активации и обучающее множество статья: <https://habr.com/ru/post/348000/>
6. Поиск похожих изображений документация: [https://oke-aditya.github.io/image\\_similarity/image\\_simi/](https://oke-aditya.github.io/image_similarity/image_simi/)
7. Практическое руководство: Python сравнивает два изображения статья: <https://pyimagesearch.com/2014/09/15/python-compare-two-images/>
8. Поиск похожих изображений на основе хеширования с учетом местоположения статья: <https://lkaihua.github.io/posts/find-similar-images-based-on-locality-sensitive-hashing/>

Результаты опроса



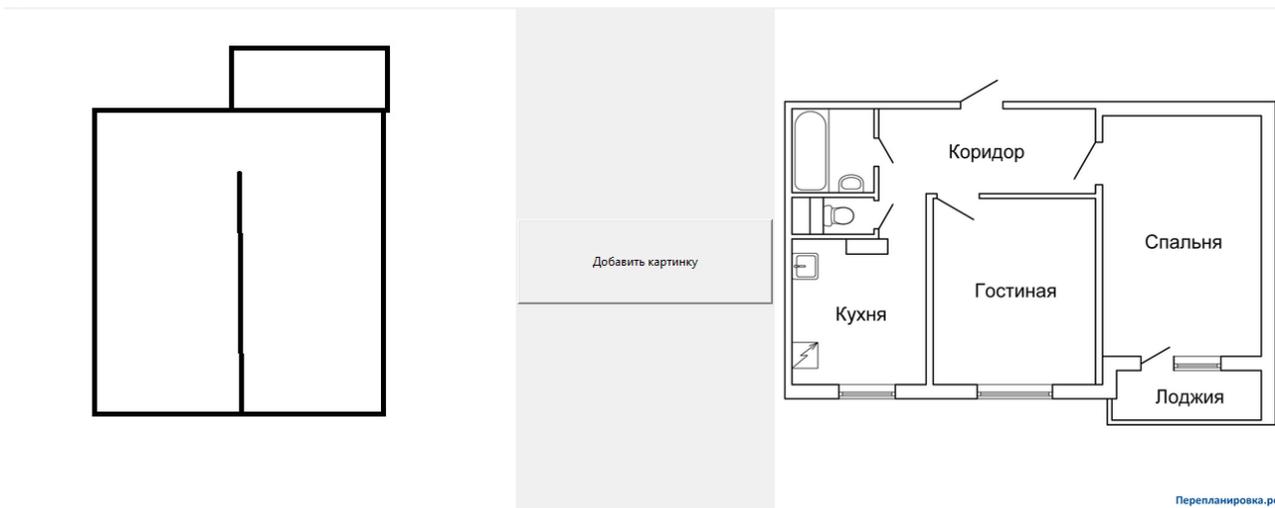


Рисунок 1.

## СОЗДАНИЕ САЙТА-ВИЗИТКИ С ПОМОЩЬЮ КОНСТРУКТОРА TILDA PUBLISHING

*Пакулев Иван*

*Муниципальное автономное общеобразовательное учреждение лицей №7, 9  
класс*

*г. Томск*

Руководитель: Козлова Елена Геннадьевна, учитель информатики

Современные технологии играют одну из ведущих ролей в повседневной жизни, а Интернет, несомненно, занимает лидирующие позиции. Проект по созданию сайта-визитки я выбрал, так как свою будущую профессию я планирую связать с IT-сферой, а навыки создания сайта всегда будут полезными. Актуальность и практическая значимость данного проекта не вызывали сомнений: моя мама работает преподавателем английского языка, и ей нужен был сайт в рекламных целях. Поскольку наши интересы совпали, я занялся созданием сайта-визитки.

Цель: Создание сайта-визитки с функцией приёма заявок.

Задачи:

- Рассмотреть доступные конструкторы сайтов и выбрать подходящий;
- Изучить выбранный конструктор;
- Продумать структуру сайта;
- Разработать дизайн и навигацию сайта;
- Подобрать материал для наполнения сайта (текст, фото, видео);
- Загрузить видео отзывы на канал YouTube;
- Заполнить сайт отобранным материалом.

Этапы и сроки реализации проекта:

Этап	Виды деятельности	Сроки
------	-------------------	-------

1	Рассмотреть доступные конструкторы сайтов и выбрать подходящий	Сентябрь 2022
2	Изучить конструктор Tilda Publishing	Октябрь 2022
3	Продумать структуру сайта	Ноябрь 2022
4	Разработать дизайн и навигацию сайта	Декабрь 2022
5	Подобрать материал для наполнения сайта	Январь 2023
6	Загрузить видео отзывы на канал YouTube	Январь 2023
7	Заполнить сайт подобранным материалом	Февраль 2023

Теоретическая часть.

Сайт-визитка — это небольшой сайт, как правило, состоящий из одной веб-страницы и содержащий основную информацию об организации, частном лице, компании, товарах или услугах, прејскуранты, контактные данные. Часто такие сайты содержат значительные объёмы графики, в том числе анимации.

Одним из важнейших плюсов сайта-визитки является то, что он не требует постоянного обновления контента. Так как цель создания сайта — это предоставление информации, которая может меняться лишь в случаях изменения контактных данных компании или лица. И если всё же потребуется внести корректировки, то это сможет сделать даже сам владелец сайта.

Практическая часть.

Сначала я определил задачи и этапы реализации проекта. Далее, согласно плану, я рассмотрел доступные конструкторы сайтов. Из найденных мною конструкторов сайтов (Wix, SquareSpace, Тинькофф) я выбрал именно Tilda Publishing, потому что это наиболее простой в использовании инструмент с интуитивно понятным интерфейсом.

Tilda Publishing — блочный конструктор сайтов, который не требует навыков программирования. Сайты на этой платформе собираются из готовых блоков, которые автоматически адаптируются под мобильные устройства и выделены в смысловые категории.

Я продумал структуру сайта и согласовал ее с заказчиком, выделил блоки для дипломов, видео отзывов и обратной связи. Перед тем как заполнить соответствующие блоки на сайте, потребовалось отсканировать дипломы и загрузить видео отзывы на канал YouTube.

Заключение.

Цель проекта была достигнута в поставленные сроки, все задачи выполнены. Я изучил Tilda Publishing и создал с его помощью реально работающий

сайт-визитку с функцией приёма заявок. Итоговый продукт эстетичен и полностью соответствует заявленным целям. Заказчик уже использует данный сайт в своей профессиональной деятельности. Ознакомиться с готовым продуктом можно по ссылке: [pakuleva-english.tilda.ws](http://pakuleva-english.tilda.ws)

Бюджет данного проекта составил 0 (ноль) рублей. На данном этапе, затраты на создание сайта-визитки были исключительно временные. В будущем целевой аудиторией могут быть как компании, так и частные профессионалы, которым требуется представить информацию о себе в интернете.

В ходе реализации проекта я научился планировать свою деятельность, ставить задачи и достигать цели. Я не собираюсь останавливаться на достигнутом результате и планирую в перспективе:

- Освоить другие функции конструктора Tilda Publishing;
- Продолжать использовать полученные навыки для создания новых сайтов;
- Использовать полученные навыки для подработки.

#### СПИСОК ЛИТЕРАТУРЫ

1. Tilda Publishing: <https://tilda.cc/ru/>
2. Википедия, статья «Тильда»:  
<https://ru.wikipedia.org/wiki/%D0%A2%D0%B8%D0%BB%D1%8C%D0%B4%D0%B0>
3. Википедия, статья «Сайт-визитка»:  
<https://ru.wikipedia.org/wiki/%D0%A1%D0%B0%D0%B9%D1%82-%D0%B2%D0%B8%D0%B7%D0%B8%D1%82%D0%BA%D0%B0>
4. Обучающий курс: <https://tilda.education/>

#### ТЕХНОЛОГИИ ПОИСКА В СЕТИ ИНТЕРНЕТ

*Пасечников Клим, Самойлов Фёдор*

*Коммунальное государственное учреждение «Средняя многопрофильная школа №37» отдела образования по городу Усть-Каменогорску управления образования ВКО, Казахстан, 11 класс*

Руководитель: Клименко Елена Сергеевна, учитель информатики, категория «педагог-исследователь»

В современном мире мы не представляем жизнь без компьютеров, смартфонов и других гаджетов. Практически ежедневно людям приходится искать какую-нибудь информацию, используя глобальную сеть интернет. Нас заинтересовал данный вопрос: как это работает?

Цель нашего проекта: проанализировать используемые поисковые системы, расширить знания по данной теме.

Задачи: изучить понятие «поисковые системы»; рассмотреть принципы работы поисковых систем; сравнить работу разных поисковиков, провести опрос по использованию систем; подобрать способы запросов для оптимизации поиска информации.

Объект исследования: использование поисковых систем. Предмет исследования: возможности поисковых систем и использование запросов для поиска информации. Гипотеза: если использовать различные способы запросов в поисковых системах, то оптимизируется поиск информации, результат получается более точный.

Обратимся к понятию «поисковая система». Что это такое? Поисковая система (или так называемый «поисковик») – это компьютерная система, созданная специально для поиска необходимой информации в интернете. Чтобы найти что-то конкретное, нужно ввести поисковый запрос. Дальше система ищет документы по ключевым фразам, либо слова, как-то связанные с ключевиками. Во время этого поисковик генерирует страницу с результатами поиска.

Поисковых систем на сегодняшний день достаточно много, все они разнообразны по интерфейсу: Google, Яндекс, Yahoo!, поиск@mail.ru и другие. Естественно, что в каждой из них поиск построен на собственных алгоритмах, но в общем принцип работы поисковых машин довольно похож.

Сбор данных. С помощью ботов ПС собирает необходимую информацию со всех страниц ресурса (тексты, изображения, видео и др.). Эти процессы начинаются после запуска сайта и в дальнейшем происходят по составленному роботом расписанию. Благодаря этому время от времени происходит обновление данных о страницах на предмет изменений.

Индексация – присвоение каждой странице определенного индекса, который позволит производить быстрый поиск по их содержимому.

Различные ресурсы могут быть отображены на страницах выдачи поисковика по отдельным запросам.

Мы проверили как работают поисковые системы. Для этого в строку поиска в разных поисковых системах ввели слово “школа”. После завершения поиска, мы сравнили результаты, которые предоставили нам системы.

Google. Было найдено 400 млн результатов. Первым результатом была ссылка на приложение “карты” и показаны ближайшие школы.

Яндекс. Было найдено 462 тыс. результатов. Первая ссылка предложение на поступление в онлайн школу. Далее значение слова “школа” и ближайшие школы в приложении 2Гис.

Рамблер. Найдено 6 тыс. результатов. Первая ссылка на сайт “okko.tv и фильм “Школа”. Далее Википедия и 2Гис.

Данный пример демонстрирует различный отклик поисковых систем при одинаковом запросе как по количеству найденных страниц, так и по содержанию.

Обратившись к рейтингу использования поисковых систем в Казахстане, отмечается, что наиболее популярной является поисковая система Google, на втором месте Яндекс.

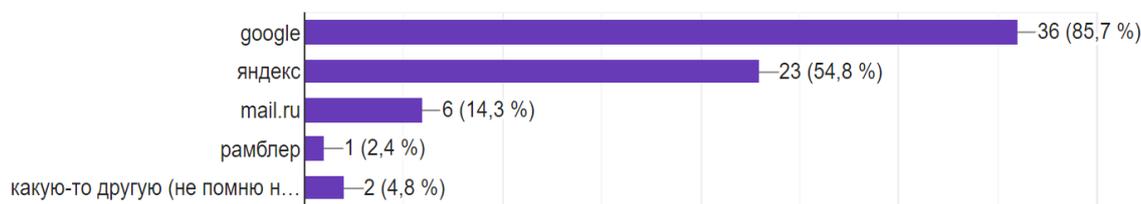
Мы провели анкетирование среди учащихся школы, учителей и родителей по использованию поисковых систем. Ответы учащихся 9-11 классов под-

тверждают популярность использования Google (85%). При этом учащиеся отмечают, что пользуются поисковыми системами несколько раз в день и при необходимости, ищут информацию к урокам и фильмы (рис 1).

Какие поисковые системы вы используете? (можно выбрать несколько вариантов)

 Копировать

42 ответа

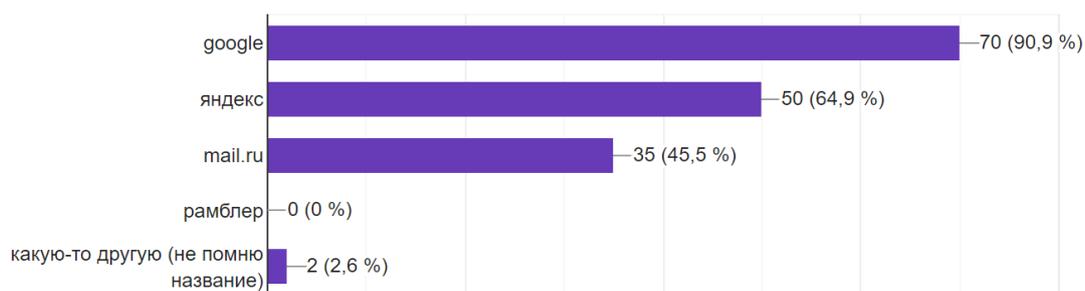


Из 77 опрошенных учителей предпочтения Google отдают 90%, при этом не исключают Яндекс 65%. Несколько раз в день и при необходимости ищут информацию по работе, полезную и для подготовки к урокам (рис 2).

Какие поисковые системы вы пользуетесь? (можно выбрать несколько вариантов)

 Копировать

77 ответов

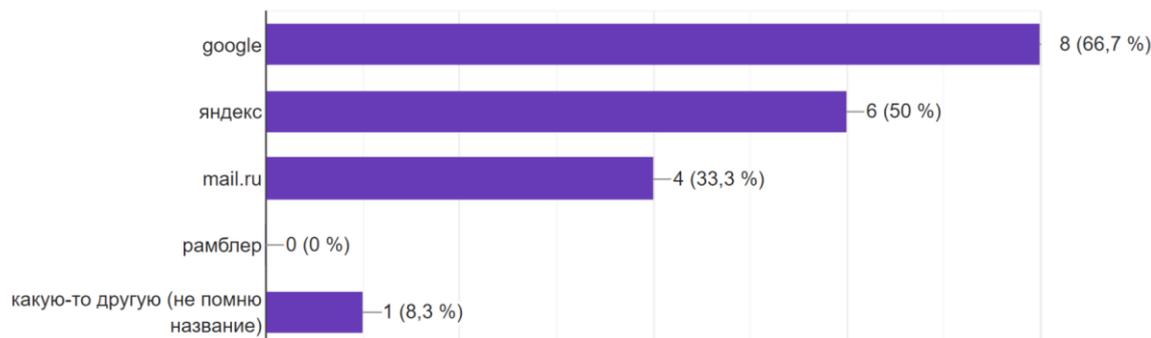


Ответы родителей: 66% пользуются Google и 50% Яндекс. Вводят запросы по поиску информации по работе, а также новостей и фильмов несколько раз в день и при необходимости (рис 3).

Какие поисковые системы вы используете? (можно выбрать несколько вариантов)

 Копировать

12 ответов



Таким образом, можно сделать вывод, что среди опрошенных нами пользователей интернет наибольшей популярностью также пользуется поисковая система Google.

Для того, чтобы можно было быстро и качественно найти нужную информацию в поисковой системе, существуют определенные правила формирования запросов. Мы изучили их и предлагаем вам также ознакомиться.

Для примера введем запрос «физика математика» в строку Google. Результатами поиска стали специальности математика-физика, группы образовательных программ, физико-математический факультет. Попробуем изменить запрос, включив правила.

Правило 1. Исключение из Google поиска. Чтобы исключить из поисковой выдачи какое-либо слово, фразу, символ и т.п., достаточно перед ним поставить знак "-" (минус), и оно не появится в результатах поиска. Это позволяет убрать ненужную информацию.

Правило 2. Обязательное включение слов в Google поиск: перед словом поставить знак "+" (плюс), то данное слово обязательно будет включено в поиск. Я хочу поступать на специальность, связанную с информатикой. Если в поисковике ввести «информатика специальности», то появится описание профессий. Но к тому же мне нравится математика. Изменим запрос «+информатика +математика специальности» и добавим «+Казахстан», то система выдаст конкретную информацию по данным специальностям в нашей стране, что нам и нужно!

Правило 3. Поиск по типу файла. Учитель по литературе дал задание прочитать произведение «А зори здесь тихие». Если ввести название в поисковик, то найдутся фильм, описание и др. Но добавив в данной фразе filetype:pdf, то сразу появится произведение в формате pdf. Поиск упростился. Для нахождения презентации необходимо ввести filetype:ppt.

Правило 4. Поиск по синонимам. Используйте символ "~" для поиска схожих слов к выбранному.

Правило 5. Поиск на выбор из вариантов. Используя оператор "|", можно осуществить Google поиск по нескольким сочетаниям фраз, заменяя несколько слов в различных местах.

Апробировав различные правила поиска на практике, одноклассникам и учителям стало проще искать нужную информацию. Мы создали «Справочник для поиска информации», в котором собрали не только представленные правила, но и другие.

Таким образом, мы проанализировали, что при вводе одинакового запроса, но используя специальные правила, результаты поиска получаем различные. Следовательно, наша гипотеза подтверждается, если использовать различные способы запросов в поисковых системах, то оптимизируется поиск информации, результат получается более точный. Необходимо лишь знать эти правила и правильно их применять. Чаще такими правилами пользуются IT-специалисты, но они будут полезны и для рядового пользователя.

## ПРОГРАММЫ ДЛЯ ВИЗУАЛИЗАЦИИ ГЕНЕАЛОГИЧЕСКИХ ДРЕВ

*Пивень Александр*

*Государственное автономное учреждение дополнительного образования  
«Кузбасский центр детского и юношеского туризма и экскурсий», 10 класс*

*МБНОУ «Городской классический лицей»*

*г. Кемерово*

Руководитель: Селиванова Любовь Петровна

Изучением родословных занимаются не только историки, литераторы, но и биологи, и медики, которые прослеживают носительство признака, гена из поколения в поколение. Родословие очень разносторонне.

2 ноября 2022 года я в онлайн формате посетил финал Межрегиональной научно-практической конференции им. Л.Р. Кызласова в 2022 году. Сама конференция проводилась в городе Абакан. Я наблюдал за выступлениями в секции «История моей семьи». Просмотрев выступление 10 человек, выяснил, что почти никто из них не смог предоставить нормальное графическое изображение генеалогического древа. Под словом «нормальный» я понимаю порядочную систему, не перегруженную лишней информацией и содержащую её необходимый минимум. На мой взгляд, представленные деревья не подходят для отображения родственных связей в семье из-за переполненности или отсутствия информации. Я неоднократно видел, как участники конференций рассказывают про историю своей семьи, сам представлял на конференциях Историю рода Пивень. К сожалению, почти все авторы аналогичных работ не пользуются существующим программным обеспечением и не придерживаются какой-либо системы при составлении родословных деревьев, генеалогических схем, поколенных росписей.

Родословное древо – схематичное представление родственных связей, в виде условно-символического древа, у корней которого указывается родоначальник, а на ветвях, его потомки [1].

Цель данного проекта: проведение анализа существующего программного обеспечения для составления генеалогических деревьев.

Данный проект будет полезен всем, кто хочет заняться изучением своей семьи, ведь он предлагает наиболее удобный и понятный для других способ создания генеалогических схем. Доминирующая в проекте деятельность: прикладной проект, исследовательский проект.

Задачи проекта:

- Составить максимально полный перечень программ для визуализации генеалогических деревьев.
- Составить сравнительную таблицу программ для визуализации генеалогических древ.
- Выбрать лучшие программы для визуализации родословных деревьев.
- Познакомить краеведов, людей, изучающих свой род, с программным обеспечением.

Определен объект проекта: программное обеспечение для компьютера.  
Предмет проекта: программы для составления генеалогических деревьев.

Методы и методики: анализ литературных источников, поиск информации в сети интернет, сравнение характеристик программ.

Продолжительность проекта: 2022-2023 учебный год.

При планировании проекта выявлено более 2 десятков тематических ресурсов для составления родословных. Среди них есть программное обеспечение, сайты, приложения. Список данных тематических ресурсов продолжает пополняться.

Планируемые результаты проекта:

- Будет составлен список программ по визуализации родословных деревьев.
- Будет определена лучшая (или лучшие) программа (программы) для визуализации генеалогических деревьев.
- Публикация результатов проекта в сети интернет.

1. Обзор существующих ресурсов

С 2019 года для создания графической схемы рода Пивень мной выбрано из аналогичных и использовано программное обеспечение GenoPro. В то время эта программа была выбрана случайно, и в её функции до работы над проектом я не вникал. В 2022 году в рамках работы над данным проектом я составил перечень программ для составления родословного дерева:

1. Генеалогическое древо семьи (FamilyTree) 1.1.7  
<https://allsoft.ru/software/hobbies/family/genealogicheskoe-drevo-familytree/>
2. Древо Жизни. – Режим доступа: <https://genery.com>
3. Дерево Поколений. – Режим доступа: <https://generatree.com/ru/index.html>
4. Родственники. – Режим доступа:  
<http://www.genealogysoft.com/ru/help.chm>
5. Семейная Летопись. – Режим доступа: <https://the-family-chronicle.com>
6. Твоё Родословное Древо. – Режим доступа:  
<https://tvoiprogrammy.ru/tvoyo-rodoslovnnoe-derevo#download-section>
7. Ages! V2.0. – Режим доступа: <http://www.daubnet.com/en/ages>
8. Ancestris. – Режим доступа: <https://www.ancestris.org/index.html>
9. FamilyHistorian. – Режим доступа: <https://www.family-historian.co.uk/>
10. FamileTreeBuilder. – Режим доступа:  
<https://www.familytreebuilder.com/family-tree-builder?lang=RU>
11. GenealogyJ. – Режим доступа: <https://sourceforge.net/projects/genj/>
12. Geni. – Режим доступа: <https://www.geni.com/family-tree/html/start>
13. GenoPro. – Режим доступа: <https://genopro.com/ru/>
14. Gramps. – Режим доступа: <https://gramps-project.org/blog/>
15. Legacy Family Tree. – Режим доступа: <https://legacyfamilytree.com>
16. MacFamilyTree: <https://apps.apple.com/us/app/macfamilytree-10/id1567970985?mt=12>

17. Master Genealogist. – Режим доступа: <https://ru.freedownloadmanager.org/Windows-PC/The-Master-Genealogist.html>
18. MyFamily (МояСемья) 2.11.114 – Режим доступа: <https://allsoft.ru/software/independent-vendors/1097936/myfamily-moya-semya/>
19. MyHeritage. – Режим доступа: <https://www.myheritage.com/family-tree?lang=RU>
20. Personal Ancestral File (PAF). – Режим доступа: <https://tvoiprogrammy.ru/personal-ancestral-file#download-section>
21. Roots Magic Essentials. – Режим доступа: <https://www.rootsmagic.com/essentials/>
22. SimTree. Семейная летопись. – Режим доступа: <https://simtree.ru/semeynaya-letopis>
- Данные программы были протестированы.

## 2. Способы изображения рода

Начнём с вопроса о том, что такое «способы представлений информации» о которых пишут авторы многих программ. Все они используют как основу один из следующих способов: нисходящее древо, восходящее древо, круговое древо, каталог и географический.

Каталог или список персон можно просмотреть в любой программе.

Географический способ представления рода в разных программах и в рамках одной программы может выглядеть по-разному. Обычно – это карта событий. К событиям относятся рождение, учеба и многое другое.

Во многих программах восходящее и нисходящее древо объединяются в одно, к примеру, это происходит в программах Древо поколений и Семейная летопись, но есть так же те, в которых есть только одно из этих деревьев, к примеру Gramps имеет только восходящее.

Все программы используют основы, изменяя количество информации. К примеру, он может показывать всех людей или семей. Возьмём для примера программу Gramps. В ней используются только одно нисходящее древо и одна карта, всё остальное – это различные каталоги и круговые деревья.

Так же моё внимание привлекла функция «мастер семьи» программы GenoPro. По сути, эта функция нужна для того, чтобы добавить к уже существующему человеку семью, с минимально необходимым количеством информации (имена, даты рождения, смерти). С помощью этой функции можно быстро добавить партнёра и нескольких детей.

## 3. Формат GEDCOM

Формат GEDCOM – это стандартный формат для генеалогических деревьев. Почти все программы (все, кроме SimTree) поддерживают как минимум импорт в этом формате. Я решил провести эксперимент с переводом одного дерева (файла) через разные программы. Изначально дерево было составлено в программе «Семейная Летопись», после чего я открывал это дерево в разных

программах. Самыми показательными оказались программа «GenoPro» и программа на андроиде «FamilyGem».

Во многих программах самым проблемным оказалось создание внебрачного ребёнка. Однако в программе «GenoPro» не возникло каких-либо трудностей, так как там понятия «муж», «жена» и т. д. не отражены в древе. Можно просто создавать анкеты новых людей.

При переносе файла в формате GEDCOM, созданного в другой программе, в программе «FamilyGem» произошла полная потеря ветви.

Кроме того, все даты сместились на один месяц - декабрь года, предшествовавшего указанному.

Также я открыл этот файл в той же программе, в которой он был сделан (Семейная летопись). В данном случае по созданной мной легенде для тестирования программ, у человека отобразилось два брака одновременно, а все дети стали отнесены к новому браку. Лучшим решением проблемы оказалось указание очерёдности браков (первая и вторая супруги).

Так же надо отметить, что если в файле формата GEDCOM есть данные, которые не используются в программе, к примеру место рождения или смерти, то после редактирования в этой программе эти данные будут утеряны во всех программах. Программа работает с теми данными, которыми она может управлять. При сохранении этих изменений остальные данные не сохраняются, отсеиваются. Так же, фотографии не хранятся в этом формате.

Кроме формата GEDCOM можно работать в форматах SVG, XML, PDF, AbiWord, kword, OpenOffice, Writer, HTML, RTF, Latex, txt, Draw, BPM, PNG, JPG, XPS, PJC, SQLite, rmgc, rmtree, ANSI. Форматы SVG, XML, HTML и PDF предложены для работы как минимум в двух разных программах.

#### 4. Краткая сравнительная характеристика программ

Самая старая из выявленных – программа «Personal Ancestral File». Она написана в 1983 г. международной некоммерческой организацией «Family Search». Самой молодой можно назвать «MyHeritage». Она создавалась с 2003 г.

Самой простой и бесплатной программой является «SimTree». Про неё подробнее рассказано ниже.

Почти все программы позволяют редактировать проекты, сделанные в другой программе, т.к. они поддерживают формат GEDCOM.

Практически все программы работают со следующими данными: ФИО, дата и место рождения, дата и место смерти, так же, часто встречаются профессия или деятельность. Есть и уникальные функции. Например, функция «найди могилу» программы Legacy Family Tree является платной и достаточно уникальной. Эта функция работает с существующей базой данных. На мой взгляд, она бесполезна для большинства людей. Только «Древо поколений» и «Legacy Family Tree» сохраняют данные функции «найди могилу».

FamilyHistorian была признан лучшим программным обеспечением 2022 года сайтом TopTenReviews (сайт о различном программном обеспечении).

Все программы, устанавливаемые на компьютер, не требуют сети.

Большинство программ русифицировано.

Рассмотрим подробнее программы, которые я рекомендую использовать.

Начнём с самой простой программы «SimTree». Её главным плюсом является простота работы с ней. Она абсолютно бесплатна, а так же использует минимальное количество информации. Однако, эта программа не поддерживает какие-либо форматы и хранит всю информацию в папках. При этом из «SimTree» можно переместить информацию в другую программу.

Дальше рассмотрим программу «Семейная Летопись». В ней очень удобно работать с информацией. Кроме того, вы сами настраиваете количество информации, отображающихся на древе, а также вы можете делать как восходящее, так и нисходящее древо для конкретно личности из списка. Кроме того, здесь есть вкладка «Биография» для каждой персоны. В эту вкладку можно вписывать любую информацию в текстовом виде.

Рассмотрим более интересную программу «GenoPro». В отличие от других представленных в этой главе программ, эта не имеет функции автоматического создания древа на основе списка персон. Вместо этого вы должны сами его рисовать. Другая интересная черта этой программы – это огромное количество информации. Часть из этой информации условно показана на древе. К примеру, подозрение на алкоголизм обозначается закрашенной в серый нижней половины значка персоны. Есть и огромное количество неотображаемых непосредственно на древе данных: начиная от врача, принимавшего роды, заканчивая организацией, отвечающей за похороны. Тут появляется одна проблема, если человек действительно серьёзно занят родословной, то у него, вероятно, окажется в базе данных родословного древа больше 40 человек, а, следовательно, не получится бесплатно сохранить эти данные программе. Если использовать «GenoPro» только для создания самого древа, то его можно просто нарисовать в этой программе, сфотографировать его и забыть. Однако, если необходимо сохранить всю эту информацию, то можно сохранять всех персон в разных проектах, так как импорт из гедкома в «GenoPro» добавляет персон в ваш действующий проект.

Рассмотрим программу «Древо поколений». Она содержит примерно столько же информации о персоне, как и «Семейная Летопись», но в отличие от неё не имеет «Биографии», в которой можно было бы писать всё что угодно. С другой стороны, в программе «Древо поколений» может делать круговое древо или даже древо – бабочку. Так же, она полностью бесплатна.

**Выводы.**

В ходе исследовательского проекта проведён анализ существующего программного обеспечения для составления генеалогических деревьев.

Составлен список из 22 программ для визуализации родословных деревьев.

Рассмотрены самые популярные способы визуализации генеалогического древа. Составлена сравнительная таблица программ для визуализации генеалогических древ.

В качестве лучших программ для визуализации генеалогических деревьев рекомендованы простая в использовании «SimTree», универсальные «GenoPro», бесплатные «Семейная Летопись», «Древо поколений».

Составлена серия информационных постов для социальных сетей с информацией о программном обеспечении для краеведов и людей, изучающих свой род. Посты будут опубликованы по графику.

## СПИСОК ЛИТЕРАТУРЫ

1. Туголукова О. В., Кадашова В. М. И моя семья должна принадлежать истории: методическое пособие. Волгоград, 2007. 120 с.

## СОЗДАНИЕ ИГРОВОЙ КОНСОЛИ НА RASPBERRY PI

*Будаев Никита, Матвеева Ангелина, Святный Иван, Пономаренко  
Никита*

*Муниципальное бюджетное общеобразовательное учреждение лицей при  
ТПУ, 10 класс  
г. Томск*

Руководитель: Мыцко Евгений Алексеевич, канд. техн. Наук

Игровая консоль – специализированное электронное устройство, для которого запуск и воспроизведение видеоигр является основной задачей. Игровые консоли всегда пользовались большим спросом (табл. 1), а сами игры занимают одно из лидирующих мест в списке развлечений в свободное время для большинства молодежи [1].

Pos	Platform	North America	Europe	Japan	Rest of World	Global
1	PlayStation 2 (PS2)	53.65	55.28	23.18	26.59	158.70
2	Nintendo DS (DS)	57.92	51.84	32.99	11.28	154.02
3	Game Boy (GB)	43.18	40.05	32.47	2.99	118.69
4	Nintendo Switch (NS)	42.29	29.76	27.13	18.10	117.27
5	PlayStation 4 (PS4)	38.20	45.92	9.58	23.32	117.03
6	PlayStation (PS)	40.78	31.09	21.59	9.04	102.49
7	Nintendo Wii (Wii)	45.51	33.12	12.77	10.23	101.63
8	PlayStation 3 (PS3)	29.92	30.87	10.47	16.14	87.40
9	Xbox 360 (X360)	47.09	25.08	1.66	11.90	85.73
10	Game Boy Advance (GBA)	40.39	21.31	16.96	2.85	81.51

Таблица 1 - Общий объем продаж игровых консолей по всему миру (в млн. штук) [2]

### Обзор игрового рынка

На данный момент по статистике на конец ноября 2022 года общий объем игрового рынка составил 184,4 миллиарда долларов, из которых 50% пришлось на мобильные игры, 28% - на консольные, 21% - на компьютерные и 1% - на браузерные ПК-игры [3]. Выручка от продажи цифровых игр составила 173,8 миллиарда долларов, а от физических изданий - 10,7 миллиарда долларов. [4]

Ниже приведен краткий обзор популярных и недавно вышедших моделей игровых консолей:

Nintendo Switch – портативная игровая консоль, занимающая 4 место среди самых продаваемых консолей в мире за все время (табл. 1). Основным преимуществом данной консоли являются контроллеры Joy-Con, которые могут быть отсоединены от приставки для совместной игры. Основным отличием от стационарных консолей заключается в более низкой производительности из-за малых размеров консоли. [5]

Steam Deck – только набирающая популярность портативная игровая система. Представляет из себя мини-компьютер, способный запускать игры из библиотеки Steam, среди которых большинство предназначены для персональных компьютеров и являются довольно требовательными к техническим характеристикам устройства, что делает консоль довольно серьезным конкурентом стационарных аналогов. [6]

Актуальность. Исходя из объемов продаж игровых консолей и видеоигр можно сделать вывод, что данная сфера разработки является прибыльной и перспективной. Однако российских аналогов на рассмотренном рынке пока нет, поэтому разработка отечественного варианта консоли на данный момент является актуальной задачей.

Практическая значимость темы. Практическая значимость проекта характеризуется тем, что наша консоль может быть использована пользователями для погружения в виртуальный мир и отдыха от повседневных забот в обстановке видеоигры.

Целью работы было: создание модели игровой консоли на Raspberry Pi и разработка игры для консоли.

Задачи:

- Разработать модель консоли.
- Создать концепт игры.
- Выбрать необходимое программное обеспечение и материалы для разработки.
- Реализовать проект игры.
- Собрать игровую консоль.
- Протестировать игру.
- Провести опрос среди первых пользователей.
- Целевая аудитория.

Наша игровая консоль подходит для любителей видеоигр средне-школьного возраста и старше.

Этапы выполнения проекта

Модель консоли

Сборкой модели консоли занимались Ангелина Матвеева и Пономаренко Никита.

Наш проект реализован на Raspberry Pi — это популярный среди начинающих радиолюбителей недорогой мини-компьютер [7], предоставляющий возможность легкого подключения дополнительных модулей к основной

плате. Технические характеристики выбранной нами модели Raspberry Pi 4 model B позволяют запускать программы, требующие немалого ресурса компьютера. [8]

Используемые материалы и оборудование:

- 7-дюймовый IPS-дисплей с сенсорным экраном.
- Пластиковый корпус.
- Игровой геймпад.
- Внешний аккумулятор на 10 тыс. мАч.
- Raspberry Pi 4 model B.

*Составление концепта игры.*

На данном этапе был сформирован концепт-документ, вмещающий в себя полное описание игры, ее игровых механик, стилистики. В результате был разработан следующий концепт: игра представляет собой 2D Top-Down Action roguelike. Главный герой – робот, пытающийся выбраться из ловушки, наполненной враждебно настроенными противниками-роботами. Игроку необходимо пройти 4 уровня, состоящие из случайно расположенных комнат, в которых находятся враги. Для перехода на следующий уровень игроку необходимо устранить всех врагов в комнатах, используя имеющееся оружие, разнообразие которого увеличивается с каждым уровнем. С каждым уровнем также увеличивается и сложность противников, появляются новые классы роботов, использующие различные тактики ведения боя.

*Выбор программного обеспечения*

В целях экономии времени и ускорения создания главной составляющей любой консоли, а именно видеоигры, было принято решение использовать специальную среду разработки. Согласно [9], выделяются два лидирующих игровых движка, подходящих для разработки 2D игр: GameMaker: Studio, Unity.

Движок GameMaker: Studio использует язык программирования GameMakerLanguage совместно с drag-and-drop интерфейсом для упрощения процесса программирования. Однако данный язык не предназначен для написания сложной логики, а бесплатная версия движка предоставляет недостаточный инструментарий для портирования игры на Raspberry Pi. [10]

Unity имеет довольно обширный инструментарий для удобного взаимодействия с проектом и реализации основных игровых механик. Данный игровой движок является наиболее популярным среди отдельных разработчиков и небольших коллективов [9], ведь Unity предоставляет понятную в освоении документацию и ряд обучающей литературы, что ускоряет и упрощает дальнейшую разработку, а кроссплатформенность позволяет без препятствий перенести проект на Raspberry Pi. В основе реализации скриптов данный движок предоставляет возможность выбора между C# и JavaScript, первый из которых уже ранее использовался членами нашей команды. [11]

В итоге мы остановились на Unity, так как он является наиболее доступным и оптимальным вариантом для нашего проекта.

В качестве программы для рисовки графики мы выбрали бесплатную программу Pixel Art, которая позволяет создавать спрайты в пиксельном стиле. Данная программа была выбрана из-за того, что ранее использовалась нашей командой для разработки видеоигр.

Наша консоль работает на базе популярной мобильной операционной системы Android, так как является оптимальной платформой для запуска игр на Unity. Технические характеристики Raspberry Pi позволяют без каких-либо проблем работать на данной операционной системе.

#### *Разработка игры*

На данном этапе проект игры поделился на две части: прототип, первая рабочая версия. Для каждой части был составлен отдельный план, в котором были прописаны все необходимые задачи согласно роли каждого участника в проекте. Будаев Никита реализовал основную логику генерации уровней, системой прорисовки мини-карты, Святный Иван разработал основную логику комнаты с врагами, систему шансов появления каждого робота, инвентарь игрока, пользовательский интерфейс, главное меню игры. Пономаренко Никита проработал поведение роботов, систему стрельбы, взаимодействие с разнообразным оружием, систему музыкального сопровождения, сохранение статистики, сценами экрана проигрыша и окончания игры. Ангелина Матвеева – прорисовка спрайтов каждого игрового объекта.

#### *Тестирование игры*

Среди наших одноклассников и знакомых мы выбрали нескольких добровольцев для тестирования нашей игры на консоли. В ходе тестирования были выявлены и исправлены недочеты и баги, доработана система появления роботов, настроены уровни сложности.

#### *Первые пользователи*

Среди тестировщиков и просто играющих был проведен опрос для оценки проделанной нами работы. (рис. 1, табл. 2)

## Консоль на Raspberry Pi

Оцените, пожалуйста, наш проект

0 | 1 | 2 | 3 | 4 | 5

### Консоль

Дизайн							
Производительность							

### Игра

Графика							
Музыка							
Интерфейс							
Геймплей							
Сложность							

Пожелания:

---



---



---



---



---

*Рисунок 1. Опросный лист*

Robot: Aocalypse 4	Оформление (итог): 4.2	Графика: 3.9	Музыка: 4.5	Ин
Итог:	Механика (итог): 3.8	Геймплей: 3.9	Сложность: 3.7	
Консоль 3.9	Дизайн: 3.7			
Итог:	Производительность: 4.1			

*Таблица 2 - Итоги проведенного опроса среди пользователей (по 5-ти бальной шкале)*

Согласно полученным результатам, игра и консоль получили оценку выше среднего. Основными недостатками проекта оказались дизайн и недостаточно разнообразный геймплей.

Для улучшения дизайна необходимо разработать удобный корпус для переноса консоли. Имеющийся джойстик необходимо заменить на более качественный вариант с более хорошим откликом кнопок.

Для добавления разнообразия в игру необходимо внедрить дополнительный контент: новые классы роботов, дополнительные задания по ходу игры.

### *Заключение*

В процессе разработки проекта были получены навыки командной работы, изучены методы долгосрочного планирования проекта, способы реализации игровых механик. Были проведены тесты игры и опрос, по результатам которых определены стороны проекта, нуждающиеся в корректировке. Итоговая цель работы была достигнута – собрана и настроена собственная модель игровой консоли, разработана видеоигра. Игровой процесс получился довольно увлекательным, первые пользователи консоли с удовольствием испытали нашу видеоигру.

Стоимость реализации проекта:

Наименование	Цена
7-дюймовый IPS-дисплей с сенсорным экраном	4,673.00 Р
Пластиковый корпус	414.00 Р
Игровой геймпад	799.00 Р
Внешний аккумулятор	1,500.00 Р
Raspberry Pi 4 model B	17,999.00 Р
<b>Итого:</b>	
Полная стоимость всех комплектующих:	25,385.00 Р
С вычетом уже имеющихся компонентов:	5,886.00 Р

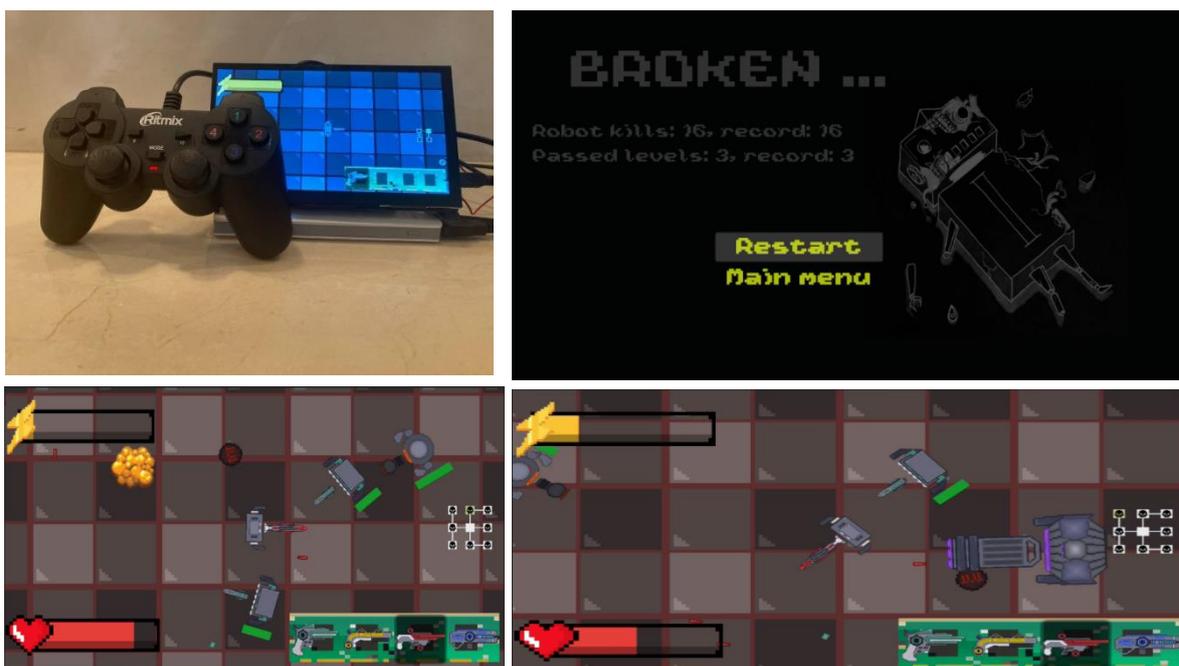


Рисунок 2, 3, 4, 5. Проектный продукт

#### Предложения по внедрению результатов

Одним из дальнейших этапов разработки является расширение возможностей консоли посредством адаптации нашего устройства для запуска мобильных игр. Видеоигру необходимо разнообразить дополнительным внутриигровым контентом и настроить для комфортной игры на мобильных устройствах, после чего она может быть загружена на платформу Google Play, что способствует её продвижению в массы потребителя.

#### СПИСОК ЛИТЕРАТУРЫ

- Ефимова Н. В., Богдан Е. И., Жешко Е. И. Компьютерные игры в жизни современной молодежи [Электронный ресурс] - Режим доступа: <https://elib.bsu.by/handle/123456789/249757> (дата обращения: 18.12.2022)
- Общий объем продаж игровых консолей по всему миру [Электронный ресурс] - Режим доступа: <https://www.vgchartz.com/charts/platform-totals/Hardware.php> (дата обращения: 18.12.2022)

4. Игровая индустрия за 2022 год в цифрах [Электронный ресурс] - Режим доступа: <https://dtf.ru/gameindustry/1518124-igrovaya-industriya-za-2022-god-v-cifrah-infografika-ot-gamesindustry-biz> (дата обращения: 18.12.2022)
5. 2022-й в цифрах: самые продаваемые игры и рост ПК-гейминга [Электронный ресурс] - Режим доступа: [https://stopgame.ru/newsdata/56350/2022\\_y\\_v\\_cifrah\\_samye\\_prodavaemye\\_igry\\_i\\_rost\\_pk\\_geyminga](https://stopgame.ru/newsdata/56350/2022_y_v_cifrah_samye_prodavaemye_igry_i_rost_pk_geyminga) (дата обращения: 18.12.2022)
6. Описание Nintendo Switch [Электронный ресурс] - Режим доступа: <https://www.nintendo.ru/-/Nintendo-Switch/Nintendo-Switch/Nintendo-Switch-1148779.html> (дата обращения: 18.12.2022)
7. Описание Steam Deck [Электронный ресурс] - Режим доступа: <https://store.steampowered.com/steamdeck?l=russian> (дата обращения: 18.12.2022)
8. What is Raspberry Pi? [Электронный ресурс] - Режим доступа: <https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi> (дата обращения: 18.12.2022)
9. Raspberry Pi Trading Ltd, Raspberry Pi 4 Computer Model B Product Brief. – 2021 – С.3.
10. What are games built with and what technologies do they use? [Электронный ресурс] - Режим доступа: <https://steamdb.info/tech/> (дата обращения: 18.12.2022)
11. GameMaker Manual [Электронный ресурс] - Режим доступа: <https://manual-ru.yoyogames.com/#t=Content.htm> (дата обращения: 18.12.2022)
12. Руководство Unity [Электронный ресурс] - Режим доступа: <https://docs.unity3d.com/ru/530/Manual/> (дата обращения: 18.12.2022)

## **РАЗРАБОТКА КОМПЬЮТЕРНОЙ ИГРЫ «ЗАВОД» НА МАТЕРИАЛЕ ИСТОРИИ САРАТОВСКОГО АВИАЦИОННОГО ЗАВОДА В ГОДЫ ВЕЛИКОЙ ОТЕЧЕСТВЕННОЙ ВОЙНЫ В СРЕДЕ UNITY**

*Ромащенко Денис*

*Государственное автономное общеобразовательное учреждение  
Саратовской области «Лицей-интернат 64», 8 класс  
г. Саратов*

Руководитель: Ромащенко Мария Александровна, канд. филос. наук

Актуальность работы. Современный мир очень быстро меняется. Дети в этом мире чувствуют себя отлично. Они хорошо владеют компьютером, с детства смотрят мультфильмы в интернете и играют в компьютерные игры. Однако в школе на уроках компьютер используют мало или, если используют, то это часто скучные и неинтересные сайты и картинки. Дети ничего не делают самостоятельно, только презентации. В школе учат так, как учили еще тогда,

когда не было компьютеров у каждого дома. Именно для того, чтобы исправить это положение дел в школе сегодня активно разрабатывают геймификацию или игрофикацию обучения. То есть в изучение предметов школьного курса включают компьютерные игры.

Также сегодня особенно актуальным является сохранение своих традиций, ценностей, объединение общества на основе самобытности, уникальности русской культуры и русской истории, а для этого необходимо изучать историю мира и историю родного края, потому что именно так человек учится понимать и ценить опыт предков, любить свою малую родину и ее традиции. Мы наблюдаем, что бережное сохранение русской истории и культуры становится очень значимой социальной проблемой. Эта тема звучит в выступлениях многих публичных деятелей Российской Федерации. докладах и выступлениях руководителей Российской Федерации. 25 мая 2022 года президент Российской Федерации В.В. Путин особенно подчеркнул в своих поручениях правительству РФ, что необходимо создавать качественный информационный контент для укрепления традиционных ценностей, патриотизма и гражданской ответственности, в том числе и среди молодого поколения. (Пр-1049 ГС, п.2 а: <http://kremlin.ru/acts/assignments/orders/68659>).

Таким образом, изучение краеведения сегодня позволяет не просто изучать историю родного края, но и то, как эта история вписывается в исторические события всего мира. Кроме того, изучая историю своей малой родины, знакомишься не только с самой историей, но и с географией, техническими науками, биологией, социологией и многими другими науками.

Цель работы состоит в том, чтобы использовать современную геймификацию в изучении истории и краеведения и тем самым сделать процесс обучения более интересным и захватывающим и создать качественный цифровой контент для сохранения исторической памяти русского народа.

Чтобы достичь данной цели, необходимо решить ряд задач:

1. Рассмотреть геймификацию как основу обучения школьника;
2. Выбрать значимое событие в истории мира, России и моего родного края, которое могло бы стать основой для создания компьютерной игры и провести исследование этого события, чтобы создать правдоподобный игровой проект;
3. Создать бета - версию компьютерной игры на основе значимое событие в истории мира, России и моего родного края.

Теоретическое значение работы состоит в том, чтобы рассказать об игрофикации в обучении школьников, показать то, как компьютерная игра помогает сделать интересным и захватывающим школьный предмет для ученика, позволяет сформировать у игрока не только интерес к истории, но и чувство гордости за свою страну, помочь ему стать участником важных исторических событий. Именно поэтому целевой аудиторией проекта являются, преимущественно, школьники средних и старших классов, а также учителя истории в школе, которые могут включать данный контент в обучение.

Практическое значение моей работы состоит в том, что по результатам исследования истории Саратовского авиационного завода создана бета-версия компьютерной игры. Эта игра посвящена Саратовскому авиационному заводу и помогает школьнику почти своими глазами увидеть историю. В игре будет показан Саратовский авиационный завод 24 июня 1943 года во время фашистского налета. Это позволит не только продемонстрировать значительное промышленное предприятие, которое сыграло важную роль в победе над Третьим Рейхом, но и показать игроку героический труд людей в тылу, погрузиться в историю самых трагичных и героических событий в нашей стране.

Работа над игрой «Завод» проходила в несколько этапов.

Первый этап – это этап, на котором была определена концепция игры и ее тип. В самом начале было понятно, что игра разрабатывается для учебного предмета «История», в частности, на материалах краеведческого характера.

История многих семей Саратова была связана с авиационным заводом. Роль Саратовского авиационного завода в и его вклад в победу в Великой отечественной войне трудно переоценить. К сожалению, сегодня завод прекратил свое существование. Именно поэтому необходимо сохранить память о нем. По типу игра «Завод» - это игра – квест, бродилка, в которой игрок от первого лица может ходить по цеху завода времен Великой отечественной войны и рассматривать внимательно локацию, решая небольшие задания.

На втором этапе был выбран движок для игры, то есть программа, которая обеспечивает физику игры и на которой строится вся игра. Выбран был движок Unity, потому что он подходит к проекту и прост в использовании.

На третьем этапе для того, чтобы сделать правдоподобную локацию в игре, а именно, цех завода времен Великой отечественной войны, были исследованы исторические материалы этого периода, изучена история САЗ в период с 1941 по 1945. Также была проведена работа со старыми архивными фотографиями цехов завода, чтобы сделать локацию игры наиболее реальной. Кроме того, понадобились исследования музыки того периода, архивы программ радиопередач, которые могли транслироваться в цеху. Также подробно изучены изображения и конструкционные особенности самолета ЯК -1 и его технические характеристики. Получив разнообразную историческую информацию об интересующем меня событии, я начал создание 3D моделей объектов, которые меня интересовали. Например, смоделирован ангар для сборки самолетов, в котором происходит действие игры и различные мелкие объекты, которые там находятся, а также модель самолета ЯК – 1.

Четвертый этап работы – это сборка игры на движке. Игра «Завод» ведется от первого лица. 24 июня 1943 года игрок оказывается в ангаре Саратовского авиационного завода, где строятся самолеты ЯК – 1. Игрок может внимательно осмотреть ангар, на стенах которого висят плакаты, взаимодействовать с различными предметами, слушать радио военного времени. Бродит игрок по ангару пока не нажимает на спрятанную кнопку. После нажатия кнопки слышится сирена воздушной тревоги, звук падающих бомб, тряска от удара и

взрыв. На этом игра заканчивается. Пока игра «Завод» существует только в бета-версии, то есть тестируется.

В дальнейших планах разработать полномасштабный сценарий игры «Завод», где будет несколько уровней сложности, игрок будет не только перемещаться и выполнять задания внутри одного из цехов завода, но и увидит историю этого завода, будет, согласно сценарию, помогать в героическом восстании этого важного стратегического объекта.

В ходе исследований документов и научных работ по истории Саратовского авиационного завода в годы Великой отечественной войны, были обнаружены некоторые интересные личные истории сотрудников и директора завода, свидетельствующие об их героических подвигах. В полной версии игры планируется включить эти истории в сценарий, чтобы игрок смог сопереживать, сочувствовать участникам тех событий и своими глазами увидеть их героизм. Кроме того, в перспективе разработки полной версии игры, будет улучшена графика и изменен движок, на котором будет работать игра.

Таким образом, подобный проект не просто поможет мотивировать ученика изучать историю и сделать ее интересной, но и покажет ему предметы того исторического периода, который он изучает, поможет понять мысли и поступки тех людей, которые жили очень давно. Геймификация в школьном обучении имеет большое будущее, потому что представляет интерес не только для тех, кто будет играть в игру, но и для того, кто эту игру будет придумывать и создавать.

Демо-версия проекта «Разработка компьютерной игры «ЗАВОД» на материале истории Саратовского авиационного завода в годы Великой отечественной войны в среде Unity» представлена по ссылке: <https://rutube.ru/video/e97c66dd1a23aa4240fb33d5eced8e95/>

## **ЦИФРОВОЙ ВОЛЬТМЕТР НА ARDUINO**

*Самойлов Фёдор, Астахов Владимир*

*Коммунальное государственное учреждение «Средняя многопрофильная школа №37» отдела образования по городу Усть-Каменогорску управления образования ВКО, Казахстан, 11 класс*

Руководитель: Клименко Елена Сергеевна, учитель информатики, категория «педагог-исследователь»

Робототехника — это современная прикладная наука, занимающаяся разработкой автоматизированных технических систем и являющаяся важнейшей технической основой развития производства. Вместе с тем робототехника становится неотъемлемой частью нашей жизни. Компьютеры, автомобили, ... — каждый день мы взаимодействуем с роботами. Но теперь они стали технологией, занявшей особое место в современной жизни. В школе имеются наборы Arduino, которые нас заинтересовали. Пришла идея создать полезный прибор, который можно будет использовать в повседневной жизни. Тема нашего проекта «Цифровой вольтметр на Arduino».

Цель нашего проекта: собрать цифровой вольтметр, используя конструктор на основе платформы Arduino.

Задачи:

1. Изучить комплектацию набора конструктора.
2. Рассмотреть возможности платформы Arduino для сборки вольтметра.
3. Собрать вольтметр, используя конструктор.
4. Провести испытание работы вольтметра на автомобиле.

Объект исследования: возможности конструктора на основе платформы Arduino. Предмет исследования: прибор для точной визуализации показателей напряжения в машине. Гипотеза: если использовать конструктор на основе платформы Arduino, то возможно собрать цифровой вольтметр, который позволит контролировать показатели напряжения в машине.

Arduino – это инструмент для проектирования электронных устройств (электронный конструктор), с возможностью приема сигналов от различных цифровых и аналоговых датчиков, которые могут быть подключены к нему, и управления различными исполнительными устройствами. Проекты устройств, основанные на Arduino, могут работать самостоятельно или взаимодействовать с программным обеспечением на компьютере. Arduino — небольшая плата с собственным процессором и памятью, которая плотно взаимодействует с окружающей средой. На данной плате есть десятки контактов, к которым можно подключать всевозможные компоненты: датчики, моторы, исполнительные устройства, и платы расширения.

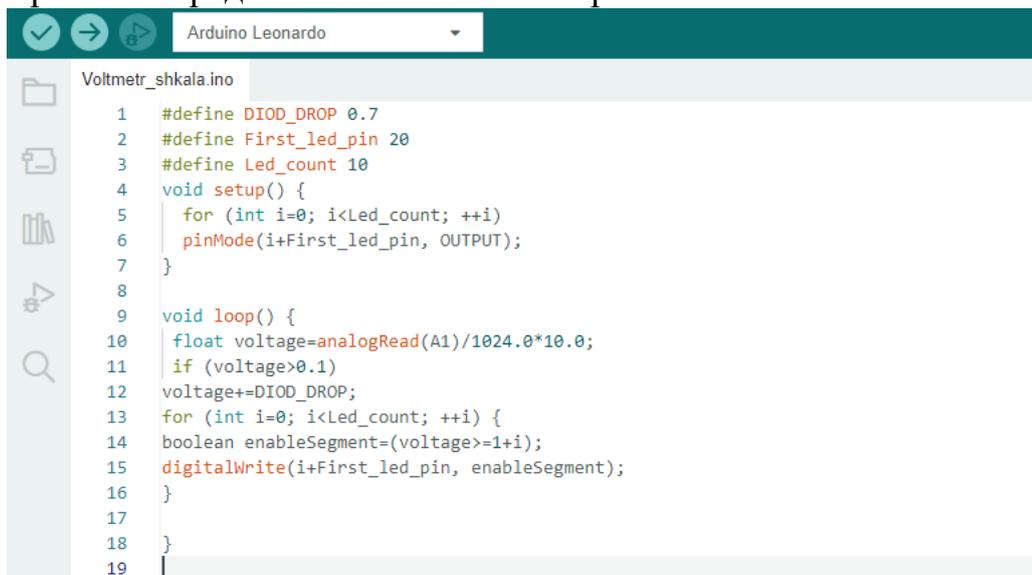
Вольтметр — это электроизмерительный прибор, который предназначен для измерения электрического напряжения на полюсах источника тока или на каком-нибудь участке электрической цепи. Вольтметр в машине нужен в первую очередь для точного фиксирования показателей постоянного тока в автомобиле. Среднее значение которого является общим стандартом 12 вольт.

Для сборки цифрового вольтметра в домашних условиях нам понадобятся такие детали конструктора как сам контроллер Arduino, монтажная площадка, резисторы, провода, светодиоды, светодиодная шкала, Bluetooth модуль.

В первую очередь мы установили резисторы на монтажную плату. Интегрировали контроллер Arduino. Соединили две платы при помощи проводов по нужным пинам. Подключили батарейки к плате. Затем перешли к программному коду. Для этого воспользовались приложением Arduino ADE.

Работа над нашим проектом состоит из 3 этапов. На первом мы апробировали простой тестер аккумуляторов и батареек. Принцип работы такого тестера довольно прост. Батарея подключается к аналоговому входу A0, который считывает напряжение на этой батарее. К цифровым линиям подключены зеленый, красный и желтый светодиоды соответственно. Они позволяют индцировать уровень заряда. Для каждого уровня можно запрограммировать свои минимальное и максимальное значения напряжения.

На втором этапе мы усложнили задачу. Значение напряжения загорается количеством сегментов на светодиодной шкале. Для каждого сегмента запрограммировано определенное значение напряжения.



```
1  #define DIOD_DROP 0.7
2  #define First_led_pin 20
3  #define Led_count 10
4  void setup() {
5      for (int i=0; i<Led_count; ++i)
6          pinMode(i+First_led_pin, OUTPUT);
7  }
8
9  void loop() {
10     float voltage=analogRead(A1)/1024.0*10.0;
11     if (voltage>0.1)
12         voltage+=DIOD_DROP;
13     for (int i=0; i<Led_count; ++i) {
14         boolean enableSegment=(voltage>=1+i);
15         digitalWrite(i+First_led_pin, enableSegment);
16     }
17 }
18 }
19 |
```

Рисунок 1. Программный код

Нам пришла идея усложнить вольтметр, сейчас мы приступили к 3 этапу – Bluetooth вольтметру. При помощи ардуино считывается напряжение на аккумуляторе. Сигнал через Bluetooth подается на телефон, на котором в свою очередь отображается показатель напряжения аккумулятора. Нам нужны написать две относительно независимых друг от друга программы: скетч для ардуино и приложение для андроид. Для создания приложения для андроид используем среду визуальной разработки App Inventor.

В автомобиле мы имеем аккумулятор и генератор. В любой машине есть установленная в зависимости от комплектации, допустим, штатная аудиосистема, на которую рассчитан генератор. К примеру: когда молодые люди обновляют систему для более громкого звучания: сабвуферы, усилители, мощные динамики, то соответственно штатный генератор не справляется с количеством потребляемого тока от всей машины, к тому же идет большая нагрузка из-за музыки. Соответственно подключается аккумулятор, с которого начинает расходоваться заряд для того, чтобы обеспечить автомобиль нужным количеством электроэнергии.

И так, наш прибор - цифровой вольтметр. Он визуализирует, когда происходит падение напряжения, мы наглядно видим, что если дальше усердно продолжать слушать музыку, то на следующее утро могут возникнуть проблемы с запуском автомобиля по причине разряженного аккумулятора. Когда значение падает ниже 12 вольт, то имеем критическое значение. Следовательно, благодаря этому вольтметру можно отслеживать то, что происходит сейчас с машиной, и иметь полное понимание и контроль благодаря визуальным показателям.

Испытаем работу вольтметра на автомобиле. Для подключения вольтметра необходимо взять два провода, далее подключить его к аккумулятору

автомобиля. Один провод подключаем к клемме с положительной полярностью, второй к клемме с отрицательным. Далее запускаем автомобиль и наблюдаем значения напряжения.

Таким образом, подключив цифровой вольтметр в автомобиле, мы имеем возможность фиксировать показатели постоянного тока и при необходимости принять меры для того, чтобы не разрядился аккумулятор.

Следовательно, наша гипотеза подтверждается, используя школьный конструктор на основе платформы Arduino, возможно собрать прибор цифровой вольтметр, который позволит контролировать показатели постоянного напряжения в машине и будет полезным не только для автолюбителей, но и тех, кому необходимо определить напряжение любого аккумулятора.

## **ПРОЕКТИРОВАНИЕ ЭЛЕКТРОННОГО ЛИСТА ПЕРСОНАЖА PATHFINDER**

*Сафьянников Артём Игоревич, МБОУ Лицей при ТПУ, 11 класс  
г. Томск*

Руководитель: Шкляр Алексей Викторович, к.т.н., доцент ОАР ТПУ.

Данный проект направлен на создание Web-приложение для настольной игры Pathfinder, на котором игроки смогут создавать своих персонажей и прописывать им характеристики. В ходе работы над проектом изучены потребности в цифровом формате листа персонажа, изучена среда разработки Django и реализовано Web-приложение на котором игроки смогут комфортно заполнить лист персонажа.

Ключевые слова: электронный лист персонажа, настольная игра, Pathfinder, цифровая таблица, пространства для игры, Web разработка, Web-приложение.

Цель: Создать Web-приложение для настольной игры Pathfinder, на котором игроки смогут прописывать характеристики персонажа.

Проблематика: Одним из важных элементов игры является заполнение параметров персонажа на листе бумаги, что вызывает неудобства из-за большого количества данных и их исправлений. Также для русскоговорящего комьюнити нет простого цифрового способа для заполнения листа персонажа.

Актуальность: в настоящий момент популярная ролевая настольная игра Pathfinder, у которой свыше 100 тысяч поклонников только в России, имеет лишь физический вариант заполнения полей для игры на листе бумаги, ручкой. Создание цифрового ресурса упростит процесс создания персонажа и привлечёт ещё больше новых игроков, на что и направлен мой проект.

Аналоги: существует зарубежный сайт ROLL20 [1] разработанный для создания игр Pathfinder в режиме онлайн. На нём много информации по самой игре, форум игроков, рынок и много другого. Из минусов можно отметить: английский язык; нет мобильного приложения для взаимодействия с листом персонажа; всю информацию в листе персонажа приходится вписывать вруч-

ную. По данным интернет-ресурсов в России аналогов нет, есть только руководства и вспомогательные информационные источники, но полноценного листа заполнения персонажа – нет. Поэтому мой проект является уникальным на Российском рынке.

Этапы выполнения проекта: 1) Исследование потребностей рынка, изучение аналогов; 2) Изучение среды разработки Django [2]; 3) Создание бэкенда; 4) Создание фронтенда и разработка дизайна страниц сайта.

Технические решения: Проект был разработан в фреймворке Django, который позволяет создавать Web-приложения и интернет ресурсы. Основа проекта (скелет) и вся его логика были написаны на python. Внешний вид сайта (Web-страница) были написаны на языке разметки html и языке JavaScript, так же были использованы каскадные таблицы стилей CSS. Для упрощения создания проекта был использован набор инструментов для создания сайтов Bootstrap [3].

Архитектура проекта: в качестве архитектуры сервиса можно выделить основные элементы проекта: бэкенд и фронтенд, которые являются главными составляющими работоспособности Web-приложения. Бэкенд: важной частью является файл «views» в котором прописана вся логика проекта, которая связывает его элементы, в частности модели и базу данных с веб страницей (html кодом), views отвечает за перенаправление данных от базы данных sqlite, к страницам и полям. Переход на страницы происходит через файла “urls”, в котором указывает путь (ссылка) для определённой страницы. Также важной частью бэкенда являются сами модели в которых прописаны поля базы данных, в которых будет храниться информация, которую пользователь будет добавлять в ходе использования Web-приложения и таблицы персонажа, а также «формы», которые позволяют создавать поля, в которых в дальнейшем пользователь будет вводить информацию, связанную с персонажем. Второй важной частью проекта является - фронтенд: html, css, JavaScript файлы, которые реализуют отрисовку страниц, полей и самой таблицы и отвечающие за внешний вид страниц сайта, позволяющие комфортное взаимодействие между пользователем и сайтом. В ходе проведения исследования потребности в таком приложении было выявлено, что целевая аудитория – это люди, которые пользуются сайтом с смартфонов. Исходя из этого разработана адаптивность сайта под мобильные устройства.

Также в Web-приложении реализована авторизация пользователя и сохранение всех данных персонажей, тем самым игрок сможет зайти на сайт спустя время и все внесённые данные будут сохранены, что позволит продолжить игру в удобное время. Ещё одной важной «новинкой» этого проекта является «система автоподсчёта», которая сильно упростит процесс подсчёта изменения характеристик, а в игре их очень много. Пример: изменил силу, поменялась ловкость персонажа и т.д. На рисунках 1-3 приведены примеры кода, который прописан в проекте, для решения поставленных задач.

```
blog > views.py > edit_1.2
14 #страница abc
15 @csrf_exempt
16 def saveforms (request, id=None):
17
18     form = CharacterForm(request.POST or None)
19     context = {'id': None}
20     if request.method=="POST" and form.is_valid():
21         #a = character.objects.get()
22         a = form.save(commit=False)
23         auto_correct(a) #передаёт значения из формы в функции
24         a.author = request.user
25         a.save()
26         context = {
27             'id' : a.id,
28             'form': form,
29         }
30         #if id==None:
31             #redirect('edit')
32         return redirect("saveforms")
33     if context:
34         return render(request,"main.html", context )
35     else:
36         return render(request,"main.html" )
37
38 #основная страница
39 def index (request):
40     return render(request,"index.html",{})
```

*Рисунок 1. Код из файла “views”, в котором прописана вся логика проекта*

```
class character(models.Model):
    author = models.ForeignKey(User, on_delete=models.CASCADE)
    date = models.DateTimeField("Дата изменения, mm/dd/yy", auto_now_add=False, auto_now=True, blank=True)
    date_create = models.DateTimeField(auto_now_add=True, auto_now=False, blank=True)
    # description - описание
    character_name = models.CharField(blank=True, null = True, max_length=255)
    worldview = models.CharField(blank=True, null = True, max_length=255)
    nameperson = models.CharField(blank=True, null = True, max_length=255)
    lvlcharacter = models.IntegerField(blank=True, null = True)
    thedeity = models.CharField(blank=True, null = True, max_length=255)
    homeland = models.CharField(blank=True, null = True, max_length=255)
    race = models.CharField(blank=True, null = True, max_length=255)
    size = models.IntegerField(blank=True, null = True)
    floor = models.CharField(blank=True, null = True, max_length=255)
    age = models.IntegerField(blank=True, null = True)
    height = models.IntegerField(blank=True, null = True)
    hair = models.CharField(blank=True, null = True, max_length=255)
    weight = models.IntegerField(blank=True, null = True)
    eye = models.CharField(blank=True, null = True, max_length=255)
```

*Рисунок 2. Код из файла “models”, в котором прописана структура хранения данных*

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">

  <link href="https://fonts.googleapis.com/css2?family=Lobster&display=swap" rel="stylesheet">
  <!-- Только CSS -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css" rel="stylesheet" i
  <link rel="stylesheet" type="text/css" href="http://www.htmlbook.ru/main.css">
  {% block title %} <title> Главная страница</title> {% endblock %}
  {% load static %}
  <link rel="stylesheet" type="text/css" href="{% static 'css/index.css' %}">
</head>
<body style="background-color:white">
  <!-- Пакет JavaScript с Popper -->
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js" crossorigin="anon
  <form method="post">
    {% csrf_token %}

    {% if request.user.is_authenticated %}
    <nav class="navbar navbar-expand-lg bg-light">
      <div class="container-fluid">
        <a class="navbar-brand" style="position: relative; margin-top: -3px" href="#"> <h3>Vailable</h3></a>
        <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupport
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="navbarSupportedContent">
        <ul class="navbar-nav me-auto mb-2 mb-lg-0">
          <li class="nav-item">
            <a style="class='nav-link active" aria-current="page" href="{% url 'index' %}">ГЛАВНАЯ</a>
          </li>
        </ul>
      </div>
    </div>
  </nav>
  </body>
</html>
```

Рисунок 3. Код главной страницы приложения (index)

Разработка дизайна приложения в стилистике настольной игры Pathfinder позволило создать целостный визуальный образ для пользователей. Приложение разработано для использования на компьютере и мобильном устройстве, адаптация рассчитана под размеры современных смартфонов. На рисунках 4 и 5 представлены скриншоты интерфейса.

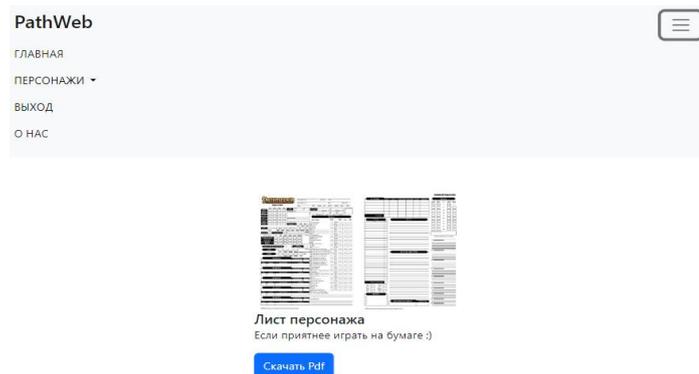


Рисунок 4. Главная страница, вид с мобильного устройства

ХАРАКТЕРИС.	ЗНАЧЕНИЕ	МОДИФИКАТОР	ВРЕМ ИЗМЕНЕНИЕ	ВРЕМ МОДИФИКАТОР
СИЛА	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
ЛОВКОСТЬ	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
ВЫНОСЛ.	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
ИНТЕЛЛЕКТ	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
МУДРОСТЬ	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
ХАРИЗМА	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Рисунок 5. Таблица листа заполнения персонажа

Основные потребители и заказчики: пользователи, играющие в Pathfinder, издатели игр, магазины настольных игр, интернет магазины игр (Steam).

Стратегию продвижения сервиса к потребителю: реклама в социальных сетях, игровых сообществах, далее люди начнут играть в компаниях и тем самым расскажут о сервисе другим игрокам, что повлияет на ещё большее продвижение.

Коммерческая составляющая проекта: возможность введения рекламы на сайт, в частности использование готовых ресурсов позволяющих размещать рекламу на интернет страницах, так же возможное введение подписок на использование сайта, что увеличит прибыль проекта.

Временные и материальные затраты: на создание проекта потребовалось 2 года работы, начиная с января 2021г и до настоящего времени. Использовался ПК и сервисы позволяющие удалённо работать с проектом. Материальные затраты только на аренду хостинга для размещения проекта (в пределах 5000 рублей), и в дальнейшем возможно на оплату курсов или обучающих материалов, связанных с технологиями разработки проекта (Django [2], html, css [4], python [5]).

В дальнейшем планируется доработка системы «автоподсчёта», косметические и функциональные улучшения страниц сайта и таблицы, для более комфортного использования игроком. Проект необходимо продолжать и развивать поскольку на данный момент игра популярна среди молодёжи и людей в возрасте до 40 лет, и данный сайт является уникальным и востребованным со стороны игрового комьюнити.

## СПИСОК ЛИТЕРАТУРЫ

1. Сайт ROLL20. [Электронный ресурс]. URL: <https://roll20.net/> (дата обращения: 20.01. 2023).
2. Вспомогательный инструмент для создания страницы сайта. [Электронный ресурс]. URL: <https://bootstrap-4.ru/> (дата обращения: 20.01. 2023).
3. Курс по Django [Электронный ресурс]. URL: <https://tutorial.djangogirls.org/ru/> (дата обращения: 20.01. 2023).
4. Документация Django [Электронный ресурс]. URL: <https://django.fun/ru/docs/django/4.1> (дата обращения: 20.01. 2023).  
Документация Html [Электронный ресурс]. URL: <http://htmlbook.ru/> (дата обращения: 20.01. 2023).
5. Самоучитель python [Электронный ресурс]. URL: <https://pythonworld.ru/samouchitel-python> (дата обращения: 20.01. 2023).

## **КОМПЬЮТЕРНЫЙ ПОДХОД К РЕШЕНИЮ И ГЕНЕРАЦИИ ШАШЕЧНЫХ ЗАДАЧ**

*Сокольников Михаил Юрьевич*

*г. Иркутск, МБУДО г. Иркутска ЦДТТ, МБУДО г. Иркутска Лицей ИГУ, 9  
класс*

Руководители: Рейнгольд Григорий Борисович, Рудых Александр  
Николаевич

Данная работа посвящена компьютерному решению и генерации задач игры «Шашки».

Цель работы: создать компьютерную программу для решения шашечных задач, на основе которой, после, создать и генератор шашечных задач.

Задачи:

- Произвести анализ аналогичных разработок.
- Сделать программу «демонстрационную доску», реализующую соблюдение всех правил при игре человек-человек.
- Сделать программу «решатель задач».
- На основе «решателя» создать «генератор задач»

За основу были выбраны русские шашки, но на поле произвольных размеров.

На первом этапе была реализована демонстрационная доска, возможность игры с автоматическим контролем за соблюдением правил.

Затем, было реализовано автоматическое решение задач. Сперва сделано решение одноходовок, а на их основе – многоходовок.

И, наконец, самая трудная и интересная часть работы – автоматическая генерация задач случайным образом. При этом используется алгоритм решения задач «Выигрыш в несколько ходов». Порядок действий:

- Как основа для будущей задачи берется положение, в котором можно выиграть за один ход.
- Составляется список всех возможных положений из которых проигрывающая сторона может прийти к позиции, полученной в пункте 1 за один ход
- В положениях, полученных в 2, проверяются все возможные ходы проигрывающей стороны. Если все ходы приводят к позиции, из которой выигрывающая сторона гарантированно побеждает за 1 ход (проверяется алгоритмом «Выигрыш в один ход»), то полученное в 2 положение помечается как «перспективное»
- Для всех «перспективных» позиций, полученных из 3, проверяется, есть ли хоть один ход у решающей стороны, приводящий к данной. Если такой ход есть, то позиция считается задачей k-го порядка.
- Данный алгоритм повторяется, повышая k за каждый проход алгоритма. Когда k принимает нужное пользователю значение, работа алгоритма оканчивается, а на экран выводится полученная позиция и выводятся все выигрышные последовательности ходов для решающей стороны.

Задачи, поставленные на первом этапе, в целом, выполнены. Создан программный продукт на языке Python.

Планируется продолжить работу по следующим направлениям:

- Дальнейшее усовершенствование генератора задач.
- Реализация решения и генерации задач не только на выигрыш, но и на наилучший ход
- Проведение исследования с целью выяснить, какой из игроков имеет преимущество в начальной позиции на досках разных размеров. Исследование имеет смысл, поскольку на доске размеров 4\*4 при обоюдной правильной игре всегда могут свести в ничью.
- Провести дополнительные исследования с вариацией игры в «поддавки»
- Попробовать решить вопрос, существуют ли такие положения, когда неоспоримое преимущество у одного из игроков имеется как в обычной игре, так и в поддавках.

После окончания имеются перспективы для использования в шашечных кружках.

## **МОДЕЛИРОВАНИЕ СВЯЗИ ХИМИЧЕСКОГО СТРОЕНИЯ И АНТИМИКРОБНОЙ АКТИВНОСТИ ФОРМАЗАНОВ МЕТОДОМ QSAR**

*Спиридонова Анна Андреевна*

*Муниципальное бюджетное общеобразовательное учреждение – средняя общеобразовательная школа № 23 с углубленным изучением английского языка г. Орла*

Научный руководитель: А.Ю.Винокуров, педагог дополнительного образования, БУ ОО ДО «Дворец пионеров и школьников им. Ю.А. Гагарина», детский технопарк «Кванториум», г. Орёл

Введение. По последним данным ВОЗ, три пункта в списке 10 ведущих причин смертности в мире занимают бактериальные инфекции или болезни, ими опосредованные (респираторные инфекции нижних дыхательных путей, диарейные болезни и туберкулез). И от них умирает каждый десятый человек в мире.

Глобальную ежегодную смертность от самых распространенных лекарственно-устойчивых штаммов инфекций оценивают минимум в 700 000 человек. Если ситуация не изменится, 100 млн человек к 2030 году умрут преждевременно, а через 35 лет эта цифра достигнет 300 млн. Мировая ежегодная смертность из-за АБР достигнет к 2050 году 10 млн

Одним из последних способов является метод компьютерного моделирования(QSAR).

QSAR Quantitative Structure-Activity Relationship – количественные и качественные модели, связывающие структуру вещества и его активность.

Стадии моделирования: формирование «обучающей» выборки соединений с известным свойством (активностью) или набором свойств; описание молекулярной структуры соединений рассматриваемой выборки с помощью дескрипторов; установление взаимосвязи «структура-биологическая активность» с последующим созданием устойчивой прогностической модели для предсказания свойств новых соединений.

Цель проекта: разработка математической модели связи строения, химических и физических свойств формазанов и их антимикробной активности

Задачи: Проанализировать исследования в области антимикробной активности формазанов. Составить обучающую выборку формазанов с известной антимикробной активностью. Представить структуры формазанов обучающей выборки в принятом для машинной обработки формате. Выполнить расчет дескрипторов для формазанов обучающей выборки. Выявить дескрипторы с наибольшей связью с антимикробной активностью. Построить модели связи химической структуры и антимикробной активности.

В последние годы появляются работы, в которых исследовано влияние формазанов на бактериальные клетки (некоторые из них представлены на слайде). По результатам данных статей можно говорить о том, что формазаны действительно способны оказывать угнетающее действие на бактерии. Однако, учитывая невероятное разнообразие возможных соединений, идея использования формазанов в этой области только начинает развиваться. Фармазоны – это соединения, содержащие азогидразонную систему связей.

Основная часть На первом этапе работы нами проводился анализ научной литературы, чтобы отобрать экспериментальные данные. После была создана база экспериментальных данных. Затем нами была проведена работа по обработке соединений в химическом редакторе, готовые файлы были сохранены в формате смайлс. После был произведен расчёт дескрипторов в программе e-DRAGON, в результате этого была получена база данных дескрип-

торов для каждого формазана.. Затем следует процесс выбора значимых дескрипторов для биологического вещества. Выбор значимых дескрипторов осуществляется при помощи коэффициента корреляции. Мы используем полученную ранее базу дескрипторов, далее при помощи коэффициента корреляции произведем расчет между дескрипторами и формазанами биологической активности, где находим коэффициент для каждого дескриптора. Нам необходим коэффициент корреляции, модуль которого больше 0.3, после мы создаем список значимых дескрипторов при заданном условии. Выбираем по 1-2 дескрипторам, и после получаем базу данных, где представлены 15-16 значимых дескрипторов для каждой группы.

На следующем этапе работы при помощи коэффициента регрессии в приложении MS Excel была получена связь коэффициентов регрессии связи антибактериальной активности и дескрипторов для каждого микроорганизма. Также стоит отметить, что нулевые значения икса было исключены из математической модели. На последнем этапе я получила математическую модель.

Итоговый продукт: модель связи дескрипторов и биологической активности.

Выводы: Ввиду разнообразия химического строения антимикробная активность формазанов может значительно изменяться, что делает необходимым применение эффективных методов скрининга, в частности QSAR, для поиска соединений с максимальной эффективностью.

2. Анализ литературы позволил составить обучающую выборку формазанов с известной антимикробной активностью, выраженной в одинаковых единицах.

3. С использованием онлайн-редактора представлены структуры формазанов обучающей выборки в принятом для машинной обработки формате.

4. Выполненный с помощью программы e-Dragon расчет выявил значительное количество значимых дескрипторов, особенно в случае патогенных микроорганизмов.

5. Методом регрессионного анализа с применением MS Excel разработаны математические модели связи строения, химических и физических свойств формазанов и их антимикробной активности. Эти модели могут быть в дальнейшем использованы для поиска эффективных соединений *in silico*.

#### СПИСОК ЛИТЕРАТУРЫ

1. Preparation and evaluation of the anti-bacterial Activity for some Formazans \*\* and Omar M. AbdulMuhsin \*\*, Fawzi H. Jumaa \* Ibtisam K. Jassim \*Department of chemistry , College of Education –Ibn-Al-Haitham, University of Baghdad. \*\*Department of chemistry , College of Education for Woman, Tikrit University .
2. Journal of Chemical and Pharmaceutical Research ISSN No: 0975-7384 CODEN(USA): JCPRC5 J. Chem. Pharm. Res., 2010, 2(4):60-67 60 Synthesis and Antimicrobial Activity of some novel Formazan Derivatives Amarish B. Samel and Nandini R. Pai\* Department of Organic Chemistry,

D.G. Ruparel College, Senapati Bapat Marg, Mahim, Mumbai-400 016, India

3. 3 QSAR — Википедия ([wikipedia.org](https://ru.wikipedia.org))
4. Nabeel Abed Abdul-Reda et al /J. Pharm. Sci. & Res. Vol. 10(5), 2018, 983-988 Synthesis and Antimicrobial Studying of Some New Formazan Derivatives from (8-Chlorotheophylline)
5. Борьба за выживание. Погибнет ли человечество из-за устойчивости к антибиотикам Анна Петренко 20 ноября 2017 г. Технологии
6. synthesis and Characterization of Formazan Derivatives from Schiff's Base and Studying their Biological Activity Rafid Q. Kmal\*, Shaimaa A. Behget, Meaad N. Husean Department of Chemistry, College of Education, University of Al-Qadisiyah, Al Diwaniyah, Iraq. Received: 12th February, 2022; Revised: 06th April, 2022; Accepted: 20th May, 2022; Available Online: 25th June, 2022
7. 7 Formazan analogous: Synthesis, antimicrobial activity, dihydrofolate reductase inhibitors and docking study Author links open overlay panel Abrar Bayazeed<sup>a</sup>Noof A. Alenazi<sup>b</sup>Amani M.R. Alsaedi<sup>c</sup>Mona H. Ibrahim<sup>d</sup>Nadia T. Al-Qurashi<sup>e</sup>Thoraya A. Farghal Journal of Molecular Structure, 15 June 2022, 132653.

## **РУКА ПОМОЩИ: КАК УСТРОЕН БИОНИЧЕСКИЙ ПРОТЕЗ**

*Спицын Т. А.*

МБОУ Академический лицей г. Томска им. Г. А. Псахье

Руководитель: Иванов Д. В. - учитель робототехники

В современном мире идет активное развитие технологий и роботизации производств, однако уровень травматизма сохраняется. Согласно статистике около 12% людей на планете имеют нарушения структур организма, снижающие его функциональность, что в свою очередь отражается на уровне физической и социальной активности человека, снижает качество жизни и становится препятствием при реализации профессиональной деятельности. Более 50 миллионов человек каждый год приобретают инвалидность по тем или иным причинам, около 300 000 теряют ноги, а 390 тысяч – руки. Утрата конечности не только приводит к физическим ограничениям, но также несет в себе и социальные проблемы для пострадавшего. Это все влияет на качество жизни человека.

Пытаясь уменьшить негативное влияние травмы, человечество изменяет инфраструктуру городов, адаптирует окружающую среду для лиц с ограниченными возможностями, а также прибегает к помощи искусственных конечностей – протезов. С развитием современных технологий выбором все чаще становятся бионические протезы.

В связи с этим мне интересно узнать, действительно ли бионический протез — это полезное изобретение, а также я хочу изучить как он устроен. На

основе собственного макета я исследую принцип работы бионического протеза.

Цель проектно-исследовательской работы:

Изучить принцип работы бионического протеза и самому сконструировать его макет.

Задачи проектно-исследовательской работы:

1. Ознакомиться с историей создания бионического протеза.
2. Понять принцип работы современных бионических протезов.
3. Изучить строение бионического протеза.
4. Исследовать принципы работы бионического протеза и сконструировать его рабочий макет.
5. Сделать выводы.

Гипотеза проектно-исследовательской работы: бионический протез может полностью вернуть функции утраченной конечности.

В 1958 году впервые прозвучал термин «бионический»: его придумал военный врач Джек Стил, занимавшийся медицинскими и аэрокосмическими исследованиями. Он исследовал природные процессы и структуры, а затем использовал их для военных разработок. В том же году в СССР разработали первую микроэлектрическую руку.

В 2007-м канадская Touch Bionics представила i-limb — первый широко доступный бионический протез. Эта рука весила всего 25 кг, обладала тонкими пальцами и открывала больше возможностей для мелкой моторики: от работы с мышкой до завязывания шнурков. Протез крепится на гильзе, легко закручивается и откручивается.

В 2018 году появились первые протезы для глаза — Argus II. Он помогает частично восстановить зрение за счет электростимуляции оставшихся клеток.

Современные протезы используют разработки робототехники, умеют имитировать индивидуальные жесты, передавать тактильные ощущения. Наконец, экзоскелеты — это переходный этап: они не заменяют утраченные органы, а дополняют, расширяя возможности человека. С их помощью люди без физподготовки могут поднимать тяжести, а парализованные — двигаться.

Все протезы состоят из трех основных частей: крепление, компонент заменяемой части и косметическая облицовка.

Современный бионический протез ноги включает несколько обязательных элементов, таких как:

- силиконовая манжета со встроенными датчиками;
- опора – титановый стержень, формой напоминающий голень;
- шарнирный модуль с микродвижками и процессором;
- блок искусственного интеллекта, обрабатывающий все поступающие сигналы.

Управление бионической рукой осуществляется через электроды с помощью электрических потенциалов мышц. Другими словами, протез «улавливает» мышечные сигналы и реагирует на них определенными движениями. Большинство задач решается двумя действиями протеза – хватом и щупом. Первое позволяет взаимодействовать с крупными предметами, второе – с мелкими, например, застегнуть молнию или завязать шнурки. Некоторые производители расширяют возможности бионических протезов, встраивая в них различные датчики и гаджеты, устройства оплаты, фонарики.



*Рисунок 1.*

Сначала я взял микроконтроллер и подключил его к питанию.



*Рисунок 2.*

Затем, я начал программировать контроллер таким образом, чтобы при воздействии импульсов рука то сгибалась, то разгибалась.



*Рисунок 3.*

После этого, я начал делать корпус моего бионического протеза.

И наконец, я приступил к изготовлению пальцев для бионического протеза.

В настоящее время протезирование добилось заметного успеха. Биопротезирование способно значительно улучшить качество жизни человека, перенесшего ампутацию. Научные исследования в области биопротезирования не замедляют темпов, в дальнейшем можно будет ожидать, что протезы получат больший спектр возможностей, станут легкими, сильными, чувствительными, энергоемкими и максимально адаптированными к потребностям человека, чутко отвечающими всем его запросам.

Также не исключено, что в скором времени появятся инновационные разработки с новыми возможностями, которые позволят бионическим протезам превосходить по своей функциональности естественные конечности.

На данный момент существует два основных направления по развитию бионических протезов. Во-первых, дать владельцу возможность почувствовать объект, к которому он прикасается, то есть сделать протез чувствительным. Во-вторых, избавиться от необходимости каждый раз снимать и надевать протез (при приеме душа или перед сном), то есть требуется вживление всех составных частей.

К сожалению, высокотехнологическое протезирование пока мало распространено ввиду высокой стоимости, наиболее известные случаи получения бионического протеза относятся к благотворительности частных организаций или общественному сбору средств. Однако государственные программы, нацеленные на развитие бионического протезирования, уже существуют.

На современном этапе развития протезирования человечество проявляет большой интерес к данной технологии, а многие даже приветствуют ее внедрение, что свидетельствует о готовности социума принять данную ступень технологического прогресса в медицине. Кроме того, данный интерес демонстрирует, что общество активно готовится к внедрению искусственных органов и позитивной адаптации людей с инвалидностью.

В ходе нашей проектно-исследовательской работы были рассмотрены вопросы истории создания бионического протеза, изучено его строение, проанализирован принцип работы биопротеза и сконструирован его рабочий макет.

В процессе создания макета мы выяснили, что бионический протез может полностью вернуть функции утраченной конечности.

Цель моей проектно-исследовательской работы достигнута, гипотеза доказана.

#### СПИСОК ЛИТЕРАТУРЫ

1. <https://rostec.ru/news/ruka-pomoshchi-kak-ustroen-bionicheskiy-protez/>
2. <https://medobr.com/news/istoriya-protezirovaniya-ot-drevneyshikh-vremendo-nashikh-dney/>
3. <https://trends.rbc.ru/trends/industry/5e91e02b9a79474e8cb6d892/>
4. <https://science-education.ru/ru/article/view?id=28969/>
5. <http://tm.spbstu.ru/%D0%A4%D0%B0%D0%B9%D0%BB:KovalevHand1.jpg/>
6. <https://vk.com/@wowfacy-bionicheskii-protez-ustroistvo-ustanovka-princip-raboty>
7. <https://cyberleninka.ru/article/n/bionicheskoe-protezirovanie-konechnosti>
8. <https://elibrary.ru/item.asp?id=35220173>
9. <https://rostec.ru/news/ruka-pomoshchi-kak-ustroen-bionicheskiy-protez/>

#### ЭЛЕКТРОННЫЙ ЖУРНАЛ ДЛЯ ПЛАЗМОХИМИЧЕСКОГО РЕАКТОРА

*Стариков Арсений*

*Муниципальное бюджетное общеобразовательное учреждение лицей при ТПУ, 10 класс  
г. Томск*

Руководитель: Герасимов Роман Дмитриевич, младший научный сотрудник ЛПМЭО ТПУ

Ключевые слова: плазмохимический реактор, электронный журнал, Google Drive, Google Drive API, C#

Во многих лабораториях ведётся журнал экспериментов. Письменный журнал не всегда практичен: его неудобно использовать при большом количестве экспериментов, неудобно заполнять, он может затеряться или прийти в негодное состояние. Поэтому большинство лабораторий также использует электронный способ хранения данных экспериментов. Современные технологии позволяют автоматизировать и централизовать хранение и обработку данных. Одним из методов работы с данными являются базы данных[1]. Такой метод отлично подходит для каталогизирования, поиска и обработки данных. Однако для ведения журналов экспериментов данный подход слишком слож-

ный в реализации и представлении данных. В качестве альтернативного варианта выступают Excel таблицы. Они удобны тем, что позволяют наглядно получать информацию (в том числе и через графики), а также их можно хранить как на персональном компьютере, так и на облачном сервере. Преимущество облачных хранилищ заключается в том, что они позволяют централизовать все данные в одном месте с доступом из любой точки, где есть интернет.

Исходя из представленных данных, целью данной работы является внедрение системы электронного журнала в интерфейс управления плазмохимическим реактором[2].

Задачи:

- Выбрать облачное хранилище;
- Изучить API облачного хранилища;
- Разработать меню подключения к облачному серверу;
- Разработать меню заполнения формы и отправки эксперимента;
- Разработать функцию создания Excel таблиц;
- Разработать функцию загрузки Excel таблиц на облачный сервер;

В качестве облачного хранилища используется Google Drive. Данный выбор обусловлен рядом особенностей: имеется бесплатный тарифный план, бесплатная библиотека Drive API Client Library для платформы .NET, а также хорошо задокументированное API. Всё это в совокупности упрощает процесс разработки и взаимодействие пользователей с журналом.

Для того, чтобы интерфейс мог взаимодействовать с Google Drive API, нужно пройти авторизацию. На рисунке 1 изображена схема авторизации. Сначала приложение запрашивает токен. Затем пользователь даёт согласие на работу с данным приложением. После соглашения пользователя приложение получает код авторизации, с помощью которого можно получить токен. Токен позволяет получить доступ к Google Drive хранилищу (далее диск) и взаимодействовать с ним.

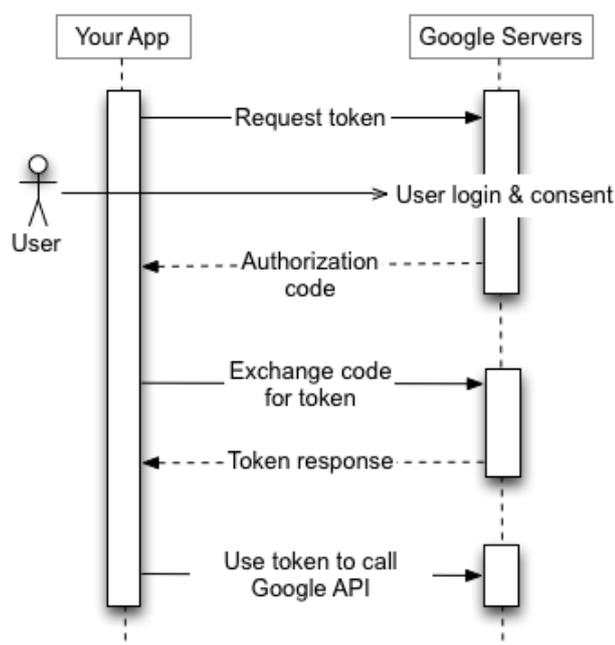


Рисунок 1. Схема авторизации и идентификации [3].

Чтобы использовать хранилище как электронный журнал, нужно его настроить. В официальной документации[4] показано, как настроить Google API и получить client id и client secret. Благодаря ним сервера Google определяют хранилище, с которым будет работать интерфейс. Также есть важное замечание: в целях безопасности, если приложение в меню соглашения имеет лого или запрашивает право на информацию об аккаунте или право на скачивание файлов, то оно должно быть верифицированным. Этот процесс может занять несколько дней. Интерфейс использует только право на загрузку файлов в хранилище. Поэтому, чтобы приложению не требовалась верификация, важно, чтобы не было лого и запрашивалось только право на загрузку файлов (<https://www.googleapis.com/auth/drive.file>).

На рисунках 2.1 и 2.2 представлены меню создания и настройки подключения к Google Drive. Здесь указываются следующие данные: Название диска (не больше 12 символов), для удобства работы разных операторов, client id и client secret, которые нужны для авторизации. Таким образом сервера Google смогут определить приложение и отправить код авторизации. После создания диска производится подключение. Если во время подключения произошла ошибка из-за некорректно введенного client id и client secret, то программа отобразит ошибку. Также программа отображает ошибку при отсутствии подключения к интернету. Аналогичное происходит при изменении уже созданного диска (за исключением случая, когда пользователь меняет только имя).

Рисунок 2.1. Меню создания диска.

Рисунок 2.2. Меню настройки диска.

На рисунке 3 изображено меню заполнения формы и отправки эксперимента. В левом верхнем углу пользователь может изменить номер серии эксперимента, а также выбрать, на какой диск будет загружаться эксперимент. При нажатии на кнопку «Сохранить на диск» создается Excel файл (рисунок 4). При возможности сохранить на диск приложение создаёт папку (если её еще нет), названием которой выступает серия экспериментов, и загружает в неё Excel таблицу с названием “категория материалов”\_“серия экспериментов”\_”номер эксперимента”. В случае, когда Excel таблица с таким же именем существует, программа просит пользователя изменить имя файла.

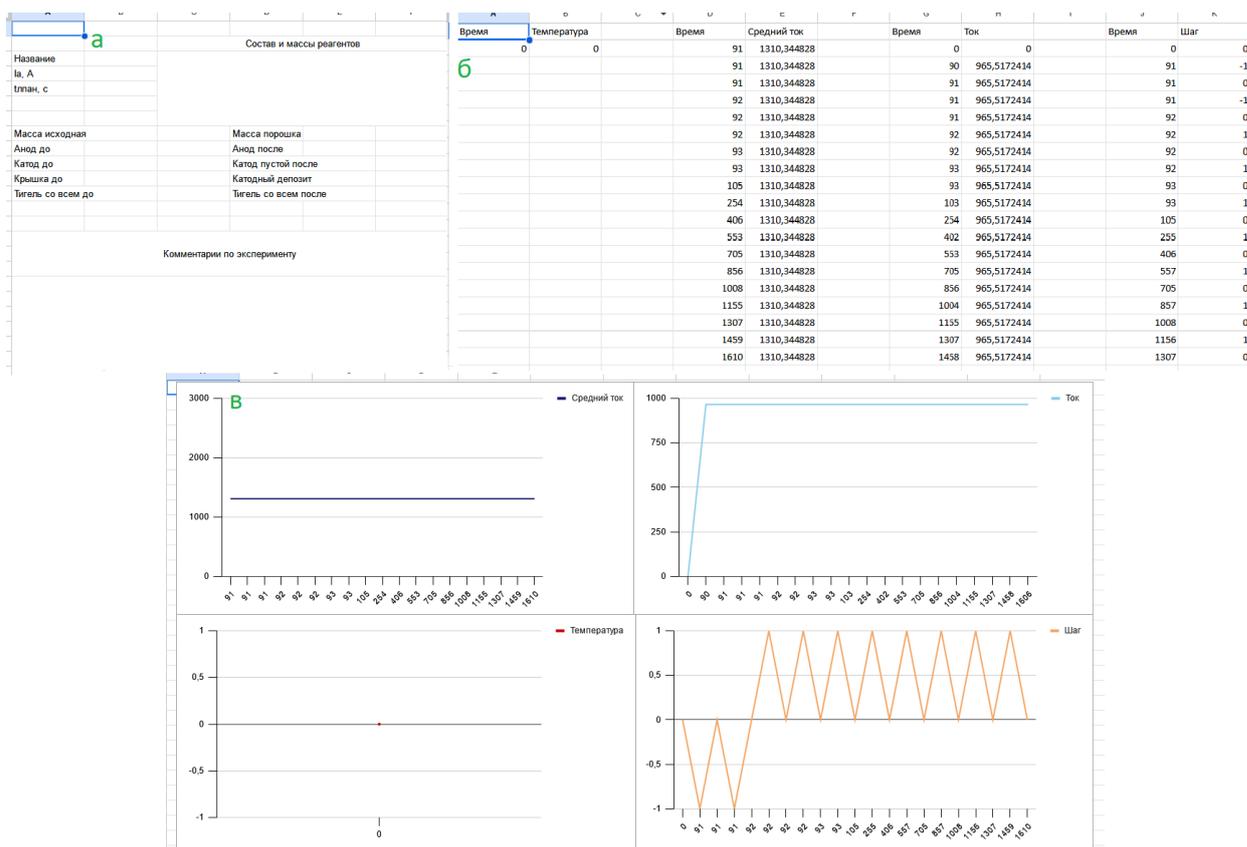
Рисунок 3. Меню заполнения формы и отправки эксперимента.

Формат Excel таблиц (рисунок 4):

На первом листе расположены общие данные об эксперименте, которые указываются в меню заполнения формы и отправки экспериментов.

На втором листе сгруппированы данные, по которым были построены графики.

На третьем листе находятся графики данных с реактора и внешнего устройства.



**Рисунок 4. Excel таблица эксперимента:**  
*а - первый лист с общими данными;*  
*б - второй лист с данными реактора и внешнего устройства;*  
*в - третий лист с графиками*

В результате проделанной работы был разработан электронный журнал, позволяющий лабораториям упростить ведение журналов экспериментов. Данный проект может быть интересен не только для лабораторий, обладающих плазмохимическим реактором с автоматическим управлением, но и для лабораторий, имеющих реактор с ручным управлением, так как журнал может сохранять данные с внешнего оборудования.

Стоит также отметить, что формат сохранения данных позволяет в дальнейшем использовать журнал как базу данных. В перспективе эти данные могут быть использованы для обучения нейронных сетей в целях создания системы поддержки принятия решений пользователя при выборе режима синтеза.

### СПИСОК ЛИТЕРАТУРЫ

1. Mu M., Tang Q., Han S., Liu X., Cui H. // Using GRand Database and Surface Water Data to Constrain Area–Storage Curve of Reservoirs // Water. 2020. V. 12. P. 1242
2. Стариков А. Д. Шаблон интерфейса для определенной категории оборудования / Стариков А. Д., Герасимов Р. Д. // Современные проблемы машиностроения сборник трудов XV Международной научно-технической

конференции, г. Томск, 22-25 ноября 2022 г.: / Национальный исследовательский Томский политехнический университет (ТПУ); ред. кол. Е. Н. Пашков [и др.]. — Томск: Томский политехнический университет, 2022. — С. 380-381

3. Using OAuth 2.0 to Access Google APIs : сайт. – URL: <https://developers.google.com/identity/protocols/oauth2> (дата обращения: 15.03.2023)
4. Develop on Google Workspace : сайт. – URL: <https://developers.google.com/workspace/guides/get-started> (дата обращения: 15.03.2023)

## **ТЕЛЕГРАММ БОТ ПО ПРОФОРИЕНТАЦИИ**

*Стручаева Ксения Максимовна*

*Муниципальное автономное общеобразовательное учреждение Заозерная средняя общеобразовательная школа с углубленным изучением отдельных предметов №16 г. Томска, 10 класс, г. Томск*

Руководитель: Осинцев Артем Викторович, педагог доп. образования, АНО ДО Технопарк "Кванториум".

Аннотация. В работе представлен прототип чат бота выполняющего роль помощника для выпускников школ при выборе профиля и направления дальнейшего обучения в высшее учебное заведение. Для прототипа выбран Telegram, как один из самых популярных мессенджеров на территории России. Разработан алгоритм и реализован прототип чат бота, который позволит оперативно и в доступной форме найти подходящие варианты для поступления в высшее учебное заведение.

Ключевые слова: профориентация, чат бот, университет, специальность.

Актуальность темы подтверждается тем, что задача о выборе профессии всегда остро стояла у выпускников школ [1–4]. Так, например, по результатам опроса сайта «Поступи Онлайн [5]» среди двух тысяч опрошенных выпускников школ 58% пока не решили, на какую специальность вуза или колледжа они подадут документы. Определились со специальностью лишь 42% опрошенных. Таким образом, можно заметить, что большая часть школьников, прошедших опрос, нуждается в доступном способе получения информации по вопросу поступления.

Разработка чат-бота для профориентации имеет несколько преимуществ, которые делают его актуальным в настоящее время. Во-первых, чат-боты являются эффективным способом коммуникации с пользователем, особенно для молодого поколения, которое привыкло общаться в мессенджерах и социальных сетях. Чат-боты могут помочь пользователям получить информацию о различных профессиях и специальностях, задавая им вопросы и предоставляя индивидуальную информацию в зависимости от ответов. Во-вторых, чат-боты могут работать круглосуточно, что позволяет пользователям получать информацию в любое время, даже если они не могут обратиться к консультанту или специалисту в реальном времени. В-третьих, разработка чат-

бота для профориентации может сэкономить время и ресурсы организаций, занимающихся профориентацией. Вместо того чтобы общаться с каждым пользователем индивидуально, чат-бот может отвечать на часто задаваемые вопросы и предоставлять информацию о профессиях и образовательных программах. Наконец, разработка чат-бота может помочь улучшить качество профориентационных услуг, поскольку бот может использовать машинное обучение и аналитику данных для анализа обратной связи от пользователей и улучшения своей работы в соответствии с потребностями пользователей. В целом, разработка чат-бота для профориентации имеет много преимуществ, которые делают ее актуальной и эффективной в настоящее время.

Цель работы: Разработать прототип чат-бота с возможностями предоставления справочного материала о ВУЗах для помощи выпускнику школы при выборе направления дальнейшего образования.

Основная часть: В ходе реализации проекта на первом этапе был составлен сценарий работы чат бота для профориентации, его пункты могут варьироваться в зависимости от конкретной реализации, но в общем виде сценарий включает в себя следующие действия:

- Приветствие пользователя и запрос на ввод информации о себе: имя, возраст, уровень образования и другие параметры, которые могут помочь определить наилучшее решение.
- Задание вопросов пользователю о его интересах, увлечениях, навыках и предпочтениях в работе.
- Анализ ответов пользователя на вопросы и определение наиболее подходящих профессий или образовательных программ.
- Предоставление информации о подходящих профессиях, образовательных программах и других связанных с ними вопросах, таких как зарплата, требования к квалификации, перспективы карьерного роста и т.д.
- Возможность задать дополнительные вопросы и запросить дополнительную информацию.
- Предоставление рекомендаций по тому, как действовать дальше в зависимости от интересов пользователя.
- Завершение сеанса чата и возможность сохранения истории диалога.

Затем был разработан алгоритм и прототип телеграмм бота с функциями поиска по городу, сдаваемым предметам, специальности, рейтингу и ключевым словам. Прототип был реализован на языке программирования Python. Целевой аудиторией такого бота будут являться выпускники школ. Заинтересованными лицами данного проекта могут являться образовательные учреждения различного уровня: школа, колледж, ВУЗ, поскольку выпускники учебных заведений могут лучше узнать об особенностях и достоинствах различных образовательных программ учебного заведения, требования к абитуриентам, проходные баллы, и различную дополнительную информацию о учебном заведении. Самыми главными преимуществами бота являются его удобный интерфейс и расположение в популярном и доступном мессенджере. Также, к

достоинствам данного чат бота относится низкая стоимость и простота его реализации, в отличие от разработки отдельного мобильного приложения которое требует полного цикла разработки программного обеспечения. Такие как: разработка UI/UX дизайна, разработка алгоритма и сценариев, поддержка версий для платформ под Android и iOS, его тестирование и публикация в маркет-плейсе.

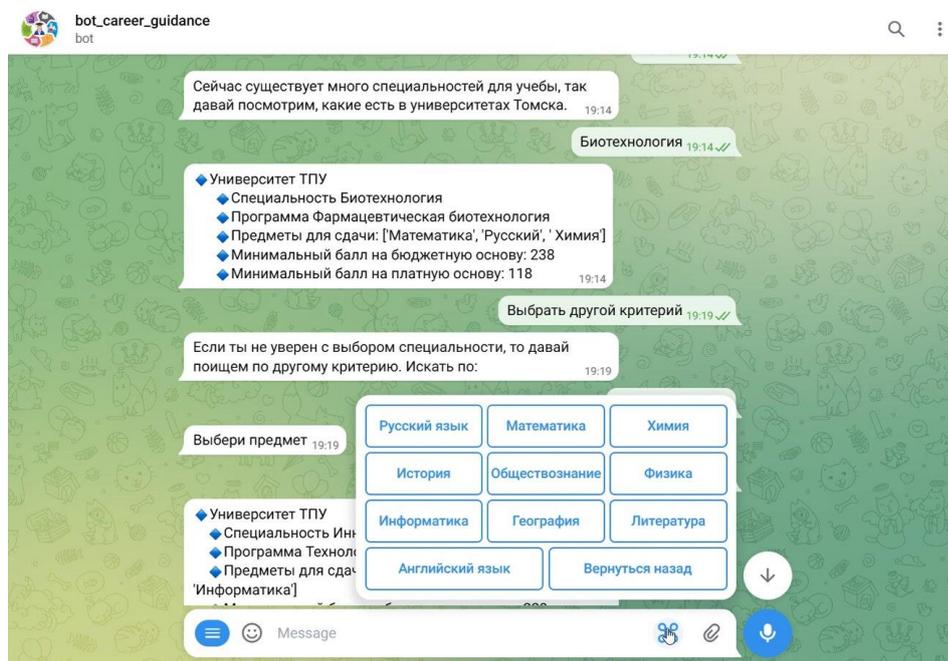


Рисунок 1.

Этот алгоритм может быть улучшен с помощью машинного обучения и анализа данных, что позволит более точно определять наиболее подходящие профессии и образовательные программы для пользователя. Чат-бот для профориентации может взаимодействовать со сторонними сервисами, чтобы получить дополнительную информацию и улучшить свою работу. Некоторые из способов, которыми бот может взаимодействовать со сторонними сервисами, включают в себя:

- Интеграция с базами данных о профессиях и образовательных программах: бот может использовать данные из сторонних баз данных для предоставления пользователю более полной и точной информации.
- Интеграция с сервисами онлайн-тестирования: бот может направлять пользователя на тесты, которые помогут определить его интересы, способности и личностные качества, а затем использовать результаты тестирования для определения подходящих профессий.
- Интеграция с рекомендательными системами: бот может использовать рекомендации от сторонних сервисов, чтобы предложить пользователю наиболее подходящие профессии и образовательные программы.
- Интеграция с платежными системами: если бот предлагает платные услуги, он может взаимодействовать со сторонними платежными системами, чтобы обеспечить безопасную и удобную оплату услуг.

- Интеграция с социальными сетями: бот может использовать информацию о пользователе из социальных сетей, чтобы предложить более персонализированные рекомендации и сделать опыт взаимодействия с ботом более удобным.
- Интеграция с API сторонних сервисов: бот может использовать API сторонних сервисов, чтобы получить доступ к дополнительным данным и функциям, таким как курсы обучения или информация о вакансиях.

Это лишь несколько примеров того, как бот может взаимодействовать со сторонними сервисами. В зависимости от конкретной реализации, бот может использовать различные сервисы и API, чтобы обеспечить более полную и точную информацию для пользователей.

Выводы и дальнейшие перспективы проекта:

Для создания чат бота необходимо провести анализ и структурную обработку данных посредством сбора информации с сайтов. В процесс будут вовлечены его заказчики и создатель бота. В будущем чат бот может преобразоваться в мультифункциональный сервис. Также возможна его реализация на нескольких языках, на которых он будет функционировать. В дальнейшем есть возможность адаптировать чат бот для конкретных целей в зависимости от желаний заказчика. Необходимость усовершенствования телеграмм бота обусловлена острой потребностью поступающих в получении в доступной форме информации по поступлению в высшее учебное заведение.

Работа с чат-ботом для профориентации имеет перспективы, так как он может предоставлять эффективную и доступную помощь в выборе профессии и образовательной программы.

Ниже приведены некоторые перспективы работы с чат-ботом профориентации:

- Удобство использования: чат-бот доступен для использования 24/7 и не требует посещения консультанта в определенное время. Это значительно упрощает процесс профориентации и делает его более удобным для пользователей.
- Персонализация: чат-бот может предоставлять персонализированные рекомендации, основанные на ответах пользователя и анализе его интересов, навыков и способностей. Это увеличивает вероятность того, что пользователь получит наиболее подходящие рекомендации.
- Эффективность: чат-бот может обрабатывать большое количество запросов одновременно и быстро предоставлять информацию. Это позволяет сэкономить время пользователя и повысить эффективность процесса профориентации.
- Низкая стоимость: использование чат-бота для профориентации может быть намного дешевле, чем оплата услуг консультанта. Это делает процесс профориентации доступным для большего числа людей.

- Повышение качества услуг: чат-бот может использовать аналитические инструменты, чтобы определять области, в которых пользователи испытывают наибольшие трудности, и улучшать качество услуг на основе этих данных.
- Расширение деятельности: чат-бот может быть использован не только для профориентации, но и для других целей, например, для оказания поддержки в выборе карьеры или для предоставления информации о вакансиях. Это расширяет возможности использования бота и может привести к дополнительным преимуществам для пользователей.

Таким образом, использование чат-бота для профориентации обладает большими перспективами в развитии и интеграции в цифровую инфраструктуру образовательных учреждений, а также может применяться как эффективный справочный инструмент для помощи людям в выборе карьеры. Работа по данному проекту будет продолжена. Будет реализована возможность выгрузки анонимной статистики пользователей чат бота, а также возможность фиксирования ботом интересов пользователя для его удобства.

#### СПИСОК ЛИТЕРАТУРЫ

1. Читайло, К. С. Возможности анализа социальных сетей в профориентационной работе / К. С. Читайло, М. С. Можаров // Информационно-коммуникационные технологии в педагогическом образовании. – 2020. – № 2(65). – С. 122-125. – EDN YZUTVB.
2. Грейман, А. А. Применение цифровых технологий в профориентации для учеников старших классов / А. А. Грейман // . – 2022. – № 1. – С. 139-155. – EDN WXANCL.
3. Мобильные приложения как инструмент профессиональной ориентации / Т. А. Блатова, В. В. Макаров, М. Э. Легких, А. В. Федоров // Труды ЦНИИС. Санкт-Петербургский филиал. – 2022. – Т. 2, № 14. – С. 28-39. – EDN FEWOQW.
4. Заболотнов, Д. А. Мобильное приложение, как средство профориентации при подготовке специалистов высоких технологий / Д. А. Заболотнов, И. Е. Жунусова, Н. А. Шумилина // Университетский комплекс как региональный центр образования, науки и культуры : Материалы Всероссийской научно-методической конференции, Оренбург, 23–25 января 2019 года. – Оренбург: Оренбургский государственный университет, 2019. – С. 752-757. – EDN YZBFXN.
5. Траектория поступления в вузы России. Режим доступа [онлайн]. URL: <https://postupi.online>.
6. Качкан, А. Р. Разработка серверной части web-приложения для профориентации / А. Р. Качкан, Н. Б. Осипенко // Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях : Материалы XXII Республиканской научной конференции студентов и аспирантов, Гомель, 25–27 марта 2019 года. –

- Гомель: Гомельский государственный университет им. Франциска Скорины, 2019. – С. 165-166. – EDN BFAPRK.
7. Готовы ли старшеклассники к выбору профессионального образования. [электронный ресурс] URL: <https://postupi.online/journal/issledovaniya-obrazovanie/gotovy-li-starsheklassniki-k-vyboru-professionalnogo-obrazovaniya>
  8. Чат-боты, AR и еще три тренда интерактивного маркетинга в 2021 году. URL: <https://www.sostav.ru/publication/smit-studio-48135.html>

## **ОСОБЕННОСТИ РЕШЕНИЯ НЕКОТОРЫХ ЗАДАЧ ПО ИНФОРМАТИКЕ НА ЯЗЫКЕ PYTHON**

*Трусов Иван Владимирович, Тараскин Дмитрий Константинович*  
*МБОУ «Тогурская СОШ имени С.В.Маслова», 11 класс, село Тогур*  
*Колпашевского района Томской области*  
Руководитель – Несмелова Нина Николаевна

Аннотация. Данная работа посвящена поиску и анализу эффективных способов решения задач по информатике, связанных с обработкой целочисленной информации, а именно — задач с масками. Рассматриваются виды задач с масками и способы их решения.

Ключевые слова: информатика, экзамен, задачи, программирование, маски, способы решения, язык Python.

Введение. Единый государственный экзамен по информатике, который должны сдавать выпускники школ, желающие связать свою дальнейшую жизнь с информационными технологиями и получить высшее образование в этой области, содержит задачи, для решения которых необходимо показать навыки программирования.

Одним из видов задач, которые предлагаются старшеклассникам в пособиях для подготовки к ЕГЭ по информатике, являются задачи с масками. Маской числа в таких задачах называется последовательность цифр, в которой могут встречаться символы «?» и «\*». При этом символ «?» может заменяться любой цифрой от 0 до 9, а символ «\*» любой последовательностью цифр произвольной длины, в том числе длина этой последовательности может быть равна нулю.

Целью нашей работы стал анализ возможных способов решения задач с масками на языке Python. Для исследования способов решения были взяты задания из сборника типовых экзаменационных вариантов для подготовки к ЕГЭ 2023 года [1], для проверки предлагаемых способов решения были использованы задачи с масками из электронного ресурса «СДАМ ГИА : РЕШУ ЕГЭ» [2].

Обзор аналогов. В работах [3,4,5] предлагается для программирования на ЕГЭ использовать язык Python. В этих работах приводятся примеры решения на языке Python задач, связанных с поиском делителей, но отсутствуют

примеры с проверкой целых чисел на соответствие маске. Поэтому мы в своей работе решили рассмотреть способы решения задач с масками на языке Python.

Основная часть. В пособии [1] задачи с масками — это задачи № 25 из вариантов с первого по четвертый. В этих задачах необходимо найти в определенном диапазоне числа, которые соответствуют заданной маске, а также в отношении этих чисел должны выполняться некоторые условия. Задания этого типа решаются практически одинаково, они очень похожи по своей сути и различаются только видом маски и условиями, которые должны выполняться для найденных чисел. Как правило, эти условия касаются делимости чисел.

Самый простой подход к решению таких задач предполагает использование одновременно двух вариантов представления информации: целочисленного и строкового. Сначала надо определить максимально возможный диапазон значений, определяемых маской и условиями задачи. Затем следует осуществлять перебором в цикле чисел из диапазона с проверкой каждого числа на соответствие условию. Найденные числа проверять на соответствие маске. Для этого число представляется в виде строки и срезы этой строки сравниваются с фрагментами маски, не содержащими символов “?” и “\*”. Этот алгоритм не предполагает вложенных циклов, что позволяет проверить даже большой диапазон чисел довольно быстро, так как для проверки миллиона чисел требуется примерно одна секунда.

Существуют несколько типов масок, которые отличаются между собой количеством знаков “?” и “\*” (ниже представлен список показательных масок, которые нужны для того, чтобы разобрать методы решения):

- 1) два вопросительных знака, это самый простой тип заданий с маской, например: «12345?7?8»;
- 2) звездочка и вопросительный знак, например: «123\*7?8»;
- 3) маска с двумя звездочками, между которыми находится цифра, отличная от других элементов маски, например: «123\*7\*123»;
- 4) маска с двумя звездочками и цифрой между звездочками, которая совпадает с элементами маски, например: «123\*3\*123».

Далее рассмотрим методы решения для каждого типа задач.

Решение маски без звездочек. В этом случае числа из диапазона даже не следует перебирать и проверять на принадлежность маске потому, что “?” — единственный символ, цифра. Мы знаем заданную нам последовательность и достаточно вместо “?” подставлять цифры. Ниже приведен фрагмент кода для решения такой задачи (рис.1).

```
l = int(input())
for x in range(10):
    for y in range(10):
        s = '12345' + str(x) + '7' + str(y) + '8'
        if int(s) % l == 0:
            print(s, int(s)//l)
```

*Рисунок 1. Решение задачи с маской без знаков «\*»*

В первой строке программы вводится произвольный делитель, далее с помощью вложенного цикла перебираются все значения для двух «?», таким образом рассматриваются все возможные варианты чисел, соответствующих маске. Сначала нужная последовательность цифр формируется в переменной строкового типа, затем с помощью функции `int()` полученная строка, соответствующая маске, превращается в целое число и это число проверяется на соответствие условию делимости. Если условие выполняется, число и его делитель выводятся в требуемом формате.

Для решения маски второго типа, включающей знаки «\*» и «?», требуется проверять число на принадлежность маске с помощью индексации и срезов (рис.2).

```
l = int(input())
for i in range(100005, 10**9 + 1, l):
    s = str(i)
    if s[:3] == '123' and s[-1] == '8' and s[-3] == '7':
        if int(s) % l == 0:
            print(s, int(s) // l)
```

*Рисунок 2. Решение задачи с маской, включающей «\*»*

В первой строке задаем произвольный делитель, затем в цикле перебираем все числа нужного диапазона. Каждое число проверяется на соответствие маске. Для того, чтобы определить нижнюю границу диапазона, достаточно в данной маске заменить знак «\*» на пустую строку «», а знак «?» на минимальное возможное значение - «0». Верхняя граница диапазона задается условиями задачи. Конвертируем числа из цикла в строковый формат, чтобы их можно было проверять на принадлежность маске с помощью индексации. Это гораздо проще, чем если бы мы проверяли принадлежность делением. Проверяем наше число на принадлежность маске. Так как «\*» – это последовательность цифр произвольной длины, мы будем проверять маску с разных концов, рассматривая последовательности цифр слева и справа от «\*». Чтобы проверить фрагмент справа используем отрицательные индексы. Если число соответствует маске, проверяем условие делимости и выводим ответ.

Решение маски типа «123\*7\*123», с двумя знаками «\*», между которыми находится цифра, отличная от всех, было бы простым, если бы эти знаки «\*» стояли рядом. Тогда бы принцип решения не отличался бы от предыдущего типа. Но в данном случае это не так. Поэтому необходимо немного модифицировать решение задачи (рис. 3).

```

l = int(input())
for i in range(1000000, 10**9):
    s = str(i)
    if s[:3] == '123' and s[-3:] == '123' and ('7' in s):
        if int(s) % l == 0:
            print(s, int(s)// l)

```

*Рисунок 3. Решение задачи с маской, включающей две «\*» и уникальную цифру между ними*

В этом случае можно проверить начало и конец маски с помощью срезов, а вхождение цифры 7 (на этом месте может стоять любая другая уникальная для данной строки цифра) проверяем стандартной конструкцией поиска вхождения.

Решение задачи с маской типа «123\*3\*123» с двумя знаками «\*» и цифрой между звездочками, которая совпадает с одним из элементов маски приведено на рисунке 4.

```

l = int(input())
for i in range(1000000, 10**9):
    s = str(i)
    if s[:3] == '123' and s[-3:] == '123' and (s.count('3') > 2):
        if int(s) % l == 0:
            print(s, int(s)// l)

```

*Рисунок 4. Решение задачи с маской, включающей два знака «\*» и число между ними, совпадающее с одним из элементов маски*

В отличие от предыдущего типа задач, наличие нужной цифры между знаками «\*» можно проверить с помощью функции count(). Следует заметить, что в маске присутствуют три тройки, и чтобы между звездочками точно присутствовала цифра 3, необходимо добавить условие, проверяющее, что наше число содержит больше двух троек.

Универсальный метод решения заданий с маской предполагает использование библиотеки fnmatch и одноименной функции из этой библиотеки. С таким подходом нам не страшны даже самые сложные маски, ведь проверку на принадлежность маске за нас делает функция fnmatch. Разбирать саму программу и синтаксис будем сразу на примерах из сборника. Рассмотрим решения задачи 25 из первого варианта сборника [1]. В этой задаче необходимо найти все числа, не превышающие сто миллионов, которые соответствуют маске «11\*223» и делятся на 149. Решение задачи с помощью функции fnmatch приведено на рисунке 5.

```

import fnmatch
for x in range(0, 10**8 + 1, 149):
    stroka = str(x)
    if fnmatch.fnmatch(stroka, '11*223'):
        if int(stroka) % 149 == 0:
            print(stroka, int(stroka) // 149)

```

Рисунок 5. Решение задачи с маской с помощью функции *fnmatch*

В первую очередь мы должны импортировать библиотеку `fnmatch`. Далее перебираем все числа, которые можно брать по условию задачи с шагом 149. Или же можно оформить начало перебора с помощью нехитрых вычислений, описанных выше, скорость работы программы от этого почти не зависит. Формируем из числа строку, ведь функция `fnmatch` работает исключительно со строками. Проверяем, соответствует ли строка нашей маске с помощью функции `fnmatch`. Следует заметить, что в параметрах функции первый параметр соответствует нашей строке, а второй – маске. Причем маска переписывается из задания такой, какая она есть. Нужно лишь заключить ее в апострофы. Как обычно, проверяем делимость и выводим ответ.

Выводы и дальнейшие перспективы исследования. Целью исследования стал поиск способов решения задач на обработку целочисленной информации с масками чисел. Для решения задач разрабатывались программы на языке Python. В ходе исследования выделено четыре типа задач с масками и предложены способы решения задач каждого типа, а также найден универсальный способ решения с помощью функции `fnmatch` из одноименной библиотеки. Показано, что язык программирования Python содержит удобные средства для решения задач с масками. В дальнейшем предполагается рассмотреть способы решения других видов задач и разработать собственное пособие для подготовке к ЕГЭ по информатике. Материалы пособия планируется представить в виде электронного курса, который можно разместить на платформе Stepik. От года к году курс будет развиваться и дорабатываться следующими поколениями старшеклассников, с учетом обновлений задач ЕГЭ. На собственном опыте мы убедились в том, что подготовка к ЕГЭ с применением исследовательской и проектной работы оказывается более интересной и эффективной, по сравнению с традиционными способами.

#### СПИСОК ЛИТЕРАТУРЫ

1. Крылов С.С., Чуркина Т.Е. Единый государственный экзамен “Информатика”. Типовые экзаменационные варианты. М: Национальное образование. - 2023, - 251 с.
2. СДАМ ГИА: РЕШУ ЕГЭ. Образовательный портал для подготовки к экзаменам: информатика. Электронный ресурс. Режим доступа: <https://inf-ege.sdamgia.ru/>. Дата обращения: 16.03.2023.

3. Ильченко О.Ю., Сырицына В.Н., Кадеева О.Е. Решение задач ЕГЭ по информатике средствами языка PYTHON // Высшее образование сегодня. - 2021. - № 11-12. - С.42-54.
4. Сорочинский М.А., Белолубский М.М. Подготовка к ЕГЭ по информатике и ИКТ: обзор заданий и решение задач на основе языка программирования PYTHON // МНИЖ. - 2021. - №8-3 (110). - С. 114-117.
5. Фирсова С.А. Особенности решения некоторых задач компьютерного ЕГЭ по информатике // Вестник Таганрогского института имени А. П. Чехова. - 2021. - №2. - С. 47-52.

## **БОТ-ПОМОЩНИК**

*Терецук Руслан*

*Муниципальное бюджетное общеобразовательное учреждение «Средняя школа № 18», 11 класс, г. Ачинск*

Руководитель: Жерносек Оксана Николаевна, учитель информатики

Актуальность. Человек всегда стремится быть осведомлённым. Ему необходимо знать об окружающих его событиях и грядущих делах как можно больше информации. Согласитесь, что иногда очень огорчает факт несбывшихся ожиданий. Речь идёт о переносе встреч, изменении в расписании автобусов и так далее. И процесс обучение в школе не исключение. Каждый обучающийся знает не по наслышке, как сильно осложняет школьную жизнь неосведомлённость в грядущем мероприятии или же изменении в расписании.

В мире существует большое количество средств, форм и способов информирования. Немалая часть из них так или иначе связана с современными техническими возможностями и с использованием глобальной компьютерной сети.

В школе, помимо электронных дневников, используется большое количество различных мессенджеров для передачи и получения информации. К ним относятся социальные сети: Telegram, Viber, WhatsApp и так далее.

Постановка и формулировка проблемы. Сейчас существуют электронные дневники, задача которых оптимизировать процесс получения информации, напрямую зависящую от наличия Интернета. Иногда возникают нестандартные ситуации и у пользователей появляются сложности с процессом получения изменений в расписание каждого класса.

Поэтому требуется рассылать информацию в рамках определённого мессенджера об изменениях классным руководителям, которые в свою очередь пересылают её обучающимся. И хотя данный процесс не сложен технически, но он довольно длителен и к тому же создаёт дополнительную нагрузку на классных руководителей во внеурочное время.

Именно эти проблемы легли в основу создания бота-помощника, который оптимизирует процесс получения школьной информации в мессенджере Telegram. Им могут воспользоваться не только обучающиеся и учителя, но и родители.

Разработанность исследуемой проблемы. В сети Интернет очень большое количество разнообразных ботов-помощников и чат-ботов. Но большая часть из них предназначены для общения, реламмы, доставок, магазинов, сервисов, для консультации и поддержки клиентов и другое [1;2]. При этом, зачастую, предоставляют свои услуги за определённую плату, взимание которой реализовано с помощью подписки [3].

Данный бот-помощник позволит оптимизировать процесс получения актуального расписания уроков и звонков, грядущих школьных мероприятий, карты школы, информации о школьных социальных сетях.

Гипотеза: можно создать бота-помощника для уведомления обучающихся об разноплановых изменениях с помощью языка программирования Python.

Цель: создать бота-помощника для оповещения обучающихся о разноплановых изменениях.

Задачи:

1. Провести опрос и обработать полученные результаты.
2. Подготовить среду разработки и установить необходимые для работы библиотеки.
3. Написать код основного функционала бота на языке программирования Python и для парсинга.
4. Подключить базу данных SQL для работоспособности функции рассылки по ID пользователей.
5. Провести апробацию работы программы.

Методы исследования: аналитический, информационное моделирование, компьютерное моделирование.

Основная часть

Среди обучающихся 5-11-х классов нашей школы провёл опрос по созданию бота-помощника. Используя сервис Google - формы сформировал новую страницу опроса и конвертировал ссылку в QR-код. Обучающимся было предложено ответить на вопросы:

1. Из какого вы класса?
2. Вам нужен бот-помощник для отслеживания школьной информации, касательно:
  - А) Актуального расписания в школе.
  - Б) Расположения кабинетов по этажам.
  - В) Расписания звонков.
  - Г) Отслеживания новостной ленты школы.
  - Д) В нём нет необходимости
  - Е) Свой вариант

Используя метод обработки статистических данных результаты опроса обработал и представил в виде диаграммы (Рисунок 1):



*Рисунок 1. «Результаты»*

В результате опроса большая часть обучающихся ответили, что бот-помощник нужен для отслеживания актуального расписания в школе.

**Этапы создания бота-помощника**

*Первый этап.* Используя компьютерное моделирование выбрал язык программирования Python для создания бота-помощника [4].

*Второй этап.* Создал API токен в интегрированном боте мессенджера Telegram «BotFather», необходимый для работы с протоколами бота, а также в качестве средства авторизации для работы с ним (Рисунок 2).



*Рисунок 2. «API токен»*

*Третий этап.* Создал основную часть бота с функциями вывода актуального расписания, расписания звонков, грядущих школьных мероприятий, карты школы, информации о школьных социальных сетях (Рисунок 3, 4, 5).

```

@bot.message_handler(commands=['start'])
def start_main_message():
    markup = types.ReplyKeyboardMarkup(resize_keyboard=True)
    butt_1 = types.KeyboardButton('Расписание')
    butt_2 = types.KeyboardButton('Назад в главное меню')
    butt_3 = types.KeyboardButton('Страница расписания школы')
    butt_4 = types.KeyboardButton('Почему? Почему? Почему?')
    butt_5 = types.KeyboardButton('Вопрос + комментарий')

    markup.add(butt_1, butt_2, butt_3, butt_4, butt_5)
    bot.send_message(message.chat.id, 'Привет! Я бот для отслеживания и учета успеваемости учащихся школы. Вы можете задать вопрос или комментарий по поводу работы бота.')
    reply_markup=markup)

```

Рисунок 3. «Главное»

Рисунок 4. «Расписание»

```

@bot.message_handler(content_types=[text])
def schedule_menu(message):
    if message.text == 'Расписание':
        markup1 = types.ReplyKeyboardMarkup(resize_keyboard=True)
        butt_11 = types.KeyboardButton('10-11 класс (старшая школа)')
        butt_21 = types.KeyboardButton('5-9 класс (средняя школа)')
        butt_31 = types.KeyboardButton('Назад в главное меню')

        markup1.add(butt_11, butt_21, butt_31)
        bot.send_message(message.chat.id, 'В каком ты классе учишься?', reply_markup=markup1)

    elif message.text == '10-11 класс (старшая школа)':

```

```

elif message.text == 'Вопрос в социальных сетях':
    markup_social = types.InlineKeyboardMarkup()
    vk = types.InlineKeyboardButton(text='Вопросы (группа в ВК)', url='https://vk.com/lab20337024')
    vk_st = types.InlineKeyboardButton(text='Счет ученика школы в ВК', url='https://vk.com/school13community')

    markup_social.add(vk, vk_st)
    bot.send_message(message.chat.id, 'Социальные сети школы', reply_markup=markup_social)

elif message.text == 'Назад в главное меню':
    start_menu(message)

```

Рисунок 5. «Социальные сети школы»

**Четвёртый этап.** Добавил функцию обновления актуальных расписаний путём его отправки в текстовом формате авторизованным в программе лицом.

**Пятый этап.** Создал отдельный код для парсинга информации из школьной новостной ленты социальной сети ВК (Рисунок 6).

**Шестой этап.** В основном коде бота добавил функцию сохранения уникального ID аккаунта ученика, необходимого для таргетированной рассылки информации, полученной с помощью парсинга (Рисунок 7).

```

import requests

token = '194301fd194301fd194301fd7a51cd511943194301fd7ab0ee246aa09a73819ae27f'
ver = 5.131
u_id = -203370024

response = requests.get('https://api.vk.com/method/wall.get',
                        params={
                            'access_token': token,
                            'v': ver,
                            'owner_id': u_id
                        })

```

Рисунок 6. «Парсер ленты»

```

@bot.message_handler(commands=['start'])
def start_main_message():
    connect = sqlalchemy.create_engine('mysql://root:root@localhost:3306/school13')
    cursor = connect.cursor()

    cursor.execute('CREATE TABLE IF NOT EXISTS user_id (username VARCHAR(255))')
    connect.commit()

    user = [message.chat.id]
    cursor.execute('INSERT OR IGNORE INTO user_id VALUES(?)', user)
    connect.commit()

```

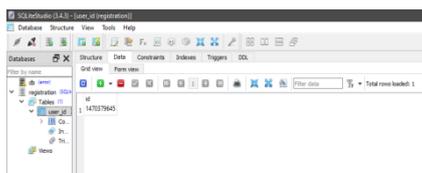


Рисунок 7. «ID пользователя»

**Седьмой этап.** Готового бота-помощника загрузил на хостинг.

**Восьмой этап.** После создания бота-помощника в течение нескольких недель проводил апробацию среди обучающихся нашей школы. Им было предложено протестировать работоспособность бота-помощника. Далее обучающимся было предложено ответить на вопросы:

1. Испытали ли вы трудности при работе с ботом-помощником?
2. Что бы Вы добавили ещё в бота-помощника?

Используя метод обработки статистических данных результаты опроса представил в виде диаграмм:



В результате апробации стало ясно, что работа с ботом-помощником вызвала интерес у большинства обучающихся. Ими было предложено интегрировать взаимодействие со столовой и преобразовать оформление.

#### Заключение

В результате проделанной работы был написан Bot для мессенджера Telegram на основе библиотеки pyTelegramBotAPI (Telebot), который, в свою очередь, облегчает мониторинг онлайн уроков для обучающихся.

Проведён опрос, результаты обработаны и представлены в виде диаграмм

Следовательно, была подтверждена выдвинутая гипотеза, достигнута цель и поставленные задачи.

Работа над проектом будет продолжена, через экосистему бота-помощника можно интегрировать работу школьной столовой.

#### СПИСОК ЛИТЕРАТУРЫ

1. SCHOOL STARS: <https://schoolstars.ru/item-work/2022-0358/?ysclid=lemq1fosp821161977>
2. Старт в науке: <https://school-science.ru/8/4/42931/?ysclid=lempqoj12k586164677>
3. Конструктор Telegram ботов: <https://puzzlebot.top/?lang=ru>
4. Википедия: <https://ru.wikipedia.org/wiki/Python>

#### СКРИПТ ДЛЯ АНАЛИЗА КРИПТО-ГРАФИКОВ

*Федосеев Степан*

*Муниципальное автономное общеобразовательное учреждение “Гимназия № 3 в Академгородке”, 10 класс  
г. Новосибирск*

Руководитель: Жакупова Стелла Анатольевна

В наше время криптовалютный рынок очень популярен, и многие люди увлекаются трейдингом<sup>1</sup> с целью заработка в этом рынке. У каждой криптовалюты есть свой график цены, который может дать нам много информации о

<sup>1</sup> Трейдинг — купля-продажа финансовых активов (инструментов) на бирже.

том, что будет происходить с данной нам валютой. Самым популярным сайтом с информацией о криптовалютных рынках является TradingView.com.

На этом сайте можно создавать собственные скрипты и индикаторы, которые могут облегчить работу с рыночными графиками. Такой скрипт/индикатор я хочу создать, так как мой отец увлекается Трейдингом и данный продукт оптимизирует его работу.

Мне захотелось разобраться в вопросах анализа крипто-графиков и написать уникальный и полезный скрипт, который в дальнейшем можно было бы использовать локально или продать.

Цель: написать скрипт/индикатор, который поможет оптимизировать работу трейдерам, используя язык скриптов “PineScript”.

Задачи:

- Узнать, что сейчас нужно трейдерам<sup>2</sup> от новых скриптов
- Изучить язык скриптов “PineScript” на том уровне, чтобы можно было написать скрипт/индикатор.
- Написать скрипт/индикатор.
- Сделать оповещение по индикатору.
- Автоматизация торговли.

Целевой аудиторией моего проекта будут сами трейдеры, которым мой скрипт/индикатор облегчит работу.

Инструменты разработки:

- Язык скриптов “PineScript”.
- Сайт TradingView.com, и его раздел “Скрипты”.
- Яндекс Cloud “Cloud function”.
- Python 3.9.

Этапы выполнения проекта:

1. Анализ графика, то есть написание индикатора.
2. Написание Telegram-бота для оповещения пользователя о срабатывании индикатора.
3. Автоматизация торговли.

Анализ графика, то есть написание индикатора:

Код написан на языке скриптов “PineScript”. Этот язык встроен в сайт TradingView. Язык был изучен с помощью документации и справочника, которые были указаны на сайте.

Суть индикатора заключается в том, чтобы находить моменты, когда новая свеча на графике больше предыдущей, но при этом ее объем меньше.

---

<sup>2</sup> Трейдер — это торговец на финансовом/фондовом рынке, который стремится извлечь прибыль.



Код индикатора:

```

//@version=5
indicator("rect", overlay=true, max_lines_count = 50)
if (math.abs(open[1]-close[1])<math.abs(open[0]-close[0]) and volume[1]>volume[0]) and open[1]-close[1]>0 and close[0]-open[0]>0 and barstate.ishistory
    line.new(bar_index[0], low[0], bar_index[1], low[1], color=color.blue)
if (math.abs(open[1]-close[1])<math.abs(open[0]-close[0]) and volume[1]>volume[0]) and open[1]-close[1]<0 and close[0]-open[0]<0 and barstate.ishistory
    line.new(bar_index[0], low[0], bar_index[1], low[1], color=color.rgb(61, 250, 108))

```

Рисунок 1. Написание Telegram-бота для оповещения пользователя о срабатывании индикатора.

Код для бота написан на языке Python 3.9, но при этом использован сервис от “Яндекс” под названием “Cloud Function”, таким образом бот всегда активен и лежит на серверах Яндекса бесплатно.

С помощью оператора “alertcondition” (оповещение с условием) были задействованы встроенные на сайте оповещения, которые в свою очередь использовали вебхук для триггера моего бота, и тот отправлял сообщение в Telegram.

```

//@version=5
indicator("rect", overlay=true, max_lines_count = 50)
if (math.abs(open[1]-close[1])<math.abs(open[0]-close[0]) and volume[1]>volume[0]) and open[1]-close[1]>0 and close[0]-open[0]>0 and barstate.ishistory
    line.new(bar_index[0], low[0], bar_index[1], low[1], color=color.blue)
if (math.abs(open[1]-close[1])<math.abs(open[0]-close[0]) and volume[1]>volume[0]) and open[1]-close[1]<0 and close[0]-open[0]<0 and barstate.ishistory
    line.new(bar_index[0], low[0], bar_index[1], low[1], color=color.rgb(61, 250, 108))
alertcondition(math.abs(open[1]-close[1])<math.abs(open[0]-close[0]) and volume[1]>volume[0]) and open[1]-close[1]>0 and close[0]-open[0]>0, title="Va upra a wpr", message="{{exchange}}({ticker)} LONG!!!")
alertcondition(math.abs(open[1]-close[1])<math.abs(open[0]-close[0]) and volume[1]>volume[0]) and open[1]-close[1]<0 and close[0]-open[0]<0, title="Va owra a wpr", message="{{exchange}}({ticker)} SHORT!!!")

```

```

1 import os
2 import requests
3 import json
4 import base64
5
6 def send_message(text):
7     url = f"https://api.telegram.org/bot%s/sendMessage" % os.environ['BOT_TOKEN']
8     data = {"chat_id": os.environ['CHAT_ID'], "text": text}
9     r = requests.post(url, data=data)
10    return r
11
12 def run(event, context):
13    try:
14        message = base64.b64decode(event['body']).decode('utf-8')
15        send_message(message)
16        r = {'statusCode': 200, 'body': 'Working'}
17
18    except Exception as e:
19        r = {'statusCode': 404, 'body': 'Some error'}
20
21    return r
22

```

Редактировать оповещение для  
AXSUSDTPERP, 1Д
✕

Настройки
Уведомления

Уведомления в приложении

Всплывающее окно

Отправить e-mail

URL вебхука ?

https://functions.yandexcloud.net/d4e1ff9h9cagbgvu1vvn

Звуковой сигнал

Звонок

30 секунд

Отправить email-по-SMS

Отмена
Сохранить

### Автоматизация торговли:

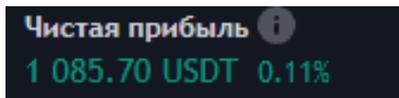
Для автоматизации торговых операций была использована стратегия, в которой есть необходимые операторы, организующие взаимодействие с самой биржей. Скрипт закупает ордера, используя уже написанное мною условие, и продает, если цена станет выше (при лонг позиции), либо ниже (при шорт позиции) на 5% от цены закупки.

```

//@version=5
strategy("Моя стратегия тест", overlay=true, margin_long=1, margin_short=-1)
var float ffl=0
var float ffs=0
if (math.abs(open[1]-close[1])<math.abs(open[0]-close[0]) and volume[1]>volume[0] and open[1]-close[1]>0 and close[0]-open[0]>0 and barstate.ishistory)
strategy.entry("test long", strategy.long)
ffl:=open[0]
if (math.abs(open[1]-close[1])<math.abs(open[0]-close[0]) and volume[1]>volume[0] and open[1]-close[1]<0 and close[0]-open[0]<0 and barstate.ishistory)
strategy.entry("test short", strategy.short)
ffs:=open[0]
if (close[0] > 1*ffl + 0.05*ffl) or (close[0] < 1*ffs + 0.05*ffs)
strategy.close_all()

```

Таким образом, если бы я запустил этот скрипт в 2017 году, то сейчас у меня была бы прибыль, равная 1085 (одна тысяча восемьдесят пять) долларов (USDT).



Чистая прибыль  
1 085.70 USDT 0.11%

### Заключение

Написан запланированный скрипт/индикатор на языке скриптов “PineScript”

Работа в данной прикладной области оказалась интересной и полезной, потому хотелось бы в дальнейшем глубже вникнуть в вопросы крипто-бирж и разработать несколько новых скриптов.

В перспективе планируется монетизировать свой продукт, используя площадки для распространения скриптов на TradingView.

### СПИСОК ЛИТЕРАТУРЫ

1. Справочник по языку PineScript // Электронный ресурс <https://ru.tradingview.com/pine-script-reference/v5/> (дата обращения 15 марта 2023)
2. PineScript v5 User Manual // Электронный ресурс <https://ru.tradingview.com/pine-script-docs/en/v5/Introduction.html> (дата обращения 14 марта 2023)

### МАСШТАБНАЯ МОДЕЛЬ БУЕРА

*Филипков Артём*

*Муниципальное автономное общеобразовательное учреждение города*

*Новосибирска*

*Гимназия №3 в Академгородке, 10 класс*

*г.Новосибирск*

Научный руководитель — П.Ю. Комков

В настоящее время все больше становится популярным альтернативные источники энергии. В связи с чем новый смысл приобретают старые способы передвижения. Было решено воспользоваться силой ветра. Благодаря ветру можно сделать лодку на коньках для перемещения по льду или плотному снегу. А именно буер. И вот в мои руки попала книга про буеры. Там было подробно описано как создавать буер своими руками. после прочтения этой книги у меня захотелось изучить подробнее, как же все-таки устроен буер. Поэтому я загорелся идеей создания 3d-модели и уменьшенной копии буера.

Буер — судно (ранее), позже лёгкая лодка или платформа, установленная на особых металлических коньках, предназначенная для скольжения по льду и оснащенная мачтой с парусами.

История буеров уходит корнями в Голландию, в которой ещё задолго до Петра Первого применяли коньки и лыжи для адаптации судов к зимним условиям.

Первый буер российской постройки был спущен на лёд в 1819 году. Строился он на Адмиралтейской верфи в Санкт-Петербурге. К сожалению, об этом буере нам ничего не известно, хотя формально он числился в различных реестрах вплоть до начала XX века. По-настоящему массовое строительство буеров началось в конце 19 века. Тогда, примерно в одно и то же время (около 1876 года) были построены два знаменитых буера — «Метель» и «Елка». «Метель» была 7 метровой и обладала площадью парусности в 30 кв. м, а «Елка» была самым большим на тот момент в мире буером: её максимальная длина достигала 15 метров, а площадь парусности составляла 190 кв. метров. О том, как выглядели эти суда мы, к сожалению, можем только догадываться, так как их изображения мне, по крайней мере, достать не удалось.

Актуальность. Буеры очень практичные для передвижения по льду на больших скоростях поэтому было решено построить средство передвижения на альтернативном (природном) источнике энергии.

Это актуально для многих в отдаленных регионах нашей страны.

Также в настоящее время набирают популярность беспилотные транспортные средства поэтому было решено добавить автопилот к буеру, а именно автоматическое регулирование паруса по ветру, и проехать от одной точки до другой.

Целью данной работы является конструирование модели буера на радиоуправлении.

Задачи:

- Создать модель буера в программе КОМПАС-3D.
- Проработать механику конструкции.
- Подобрать материалы и технологию изготовления.
- Изготовить детали модели с использованием станка с ЧПУ и 3D-принтера .
- Собрать и испытать изделие.
- Спланировать электронику для управления.
- Смонтировать.

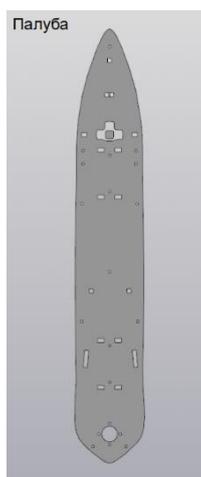
В данном проекте выполнена модель буера в программе КОМПАС-3D; проработана механика конструкции; подобраны материалы и технология изготовления; изготовлены детали модели с использованием станка с ЧПУ и 3D-принтера; спланирована электроника для управления; собрано и испытано изделие в режиме дистанционного управления.

Размеры модели 400x350x350 мм.

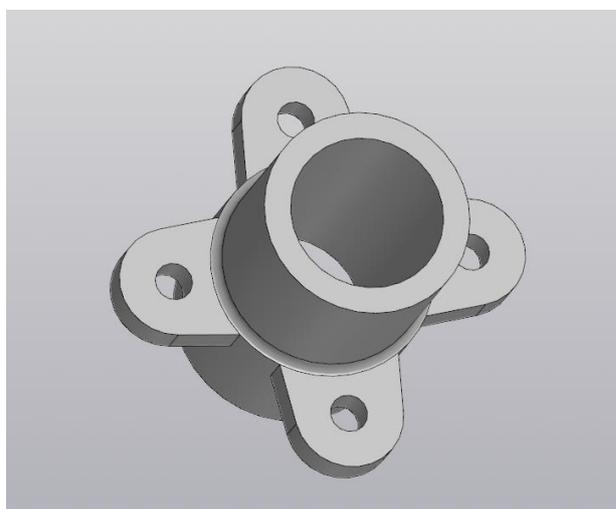
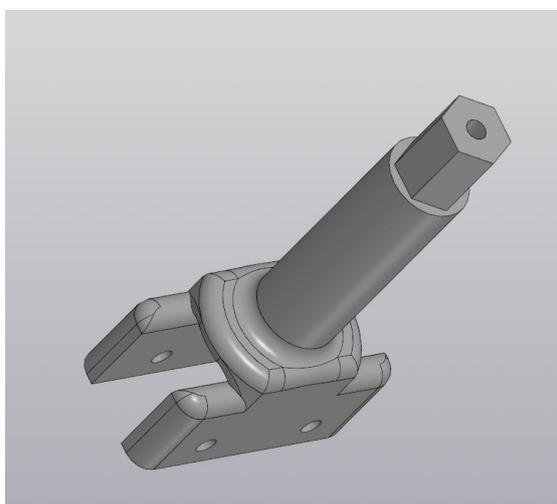
Этапы выполнения:

- Построение 3D модели в программе КОМПАС-3D
  - Сборка
- Построение 3D модели в программе КОМПАС-3D:

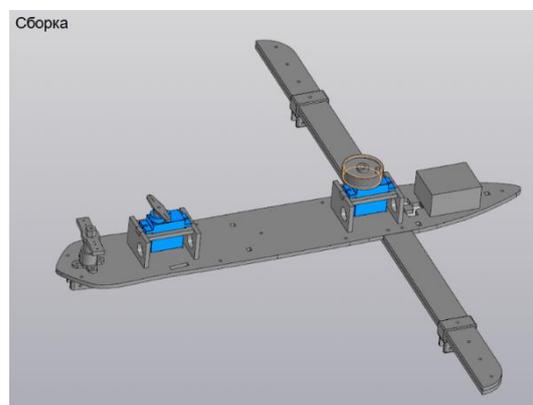
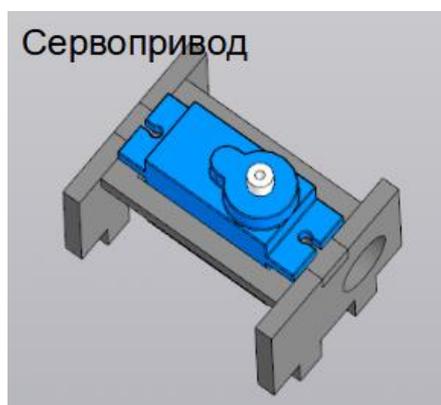
Сначала была построена палуба, а также крылья. Затем были спроектированы детали для крепления и управления рулевого конька.



Держатель рулевого конька и обтекатель.

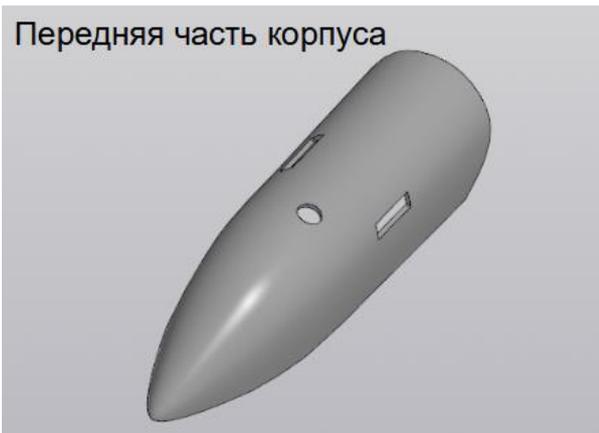


Внутри корпуса расположено 2 сервомашинки, отсек для батареи и радиомодуль.

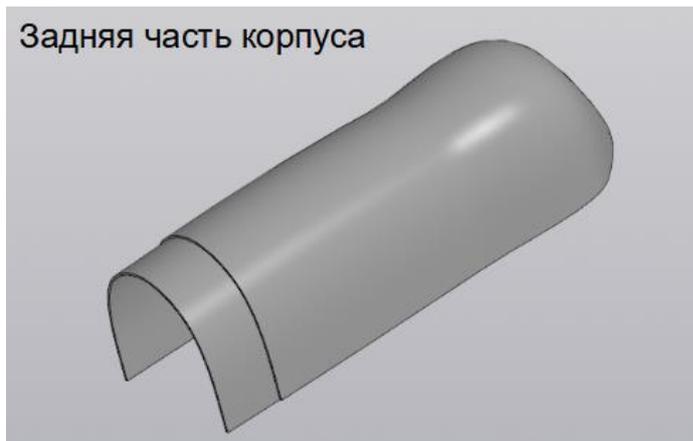


Обтекатель выполнен из двух частей для лучшего доступа к электронике.

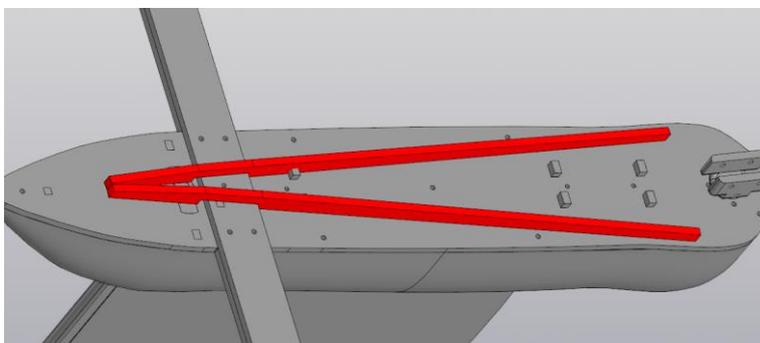
Передняя часть корпуса



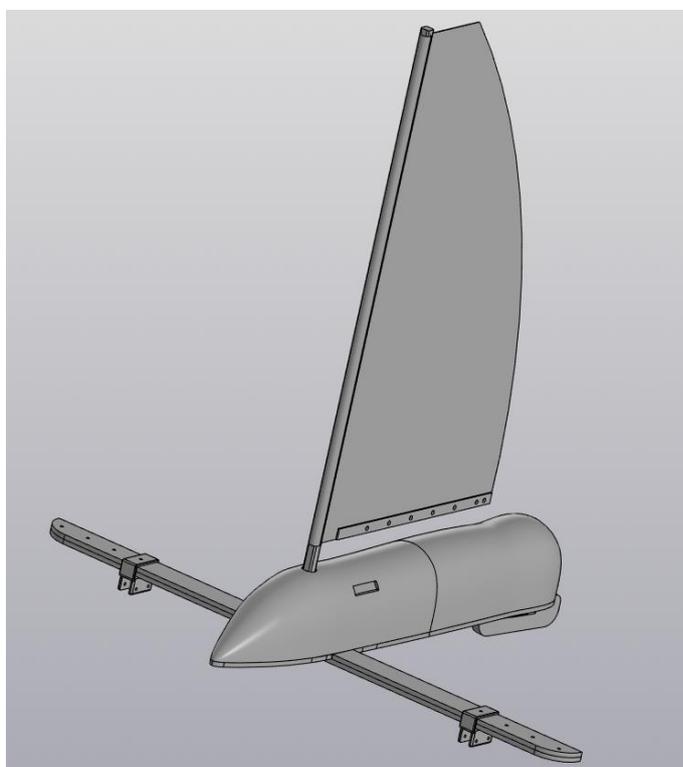
Задняя часть корпуса



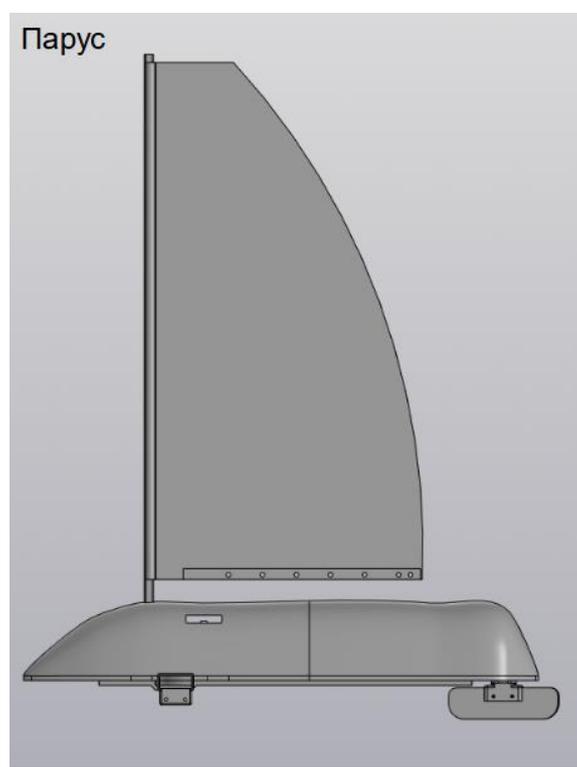
Также в конструкцию были добавлены ребре жёсткости для увеличения прочности конструкции.



После ребер жесткости мы добавили парус из листового материала.

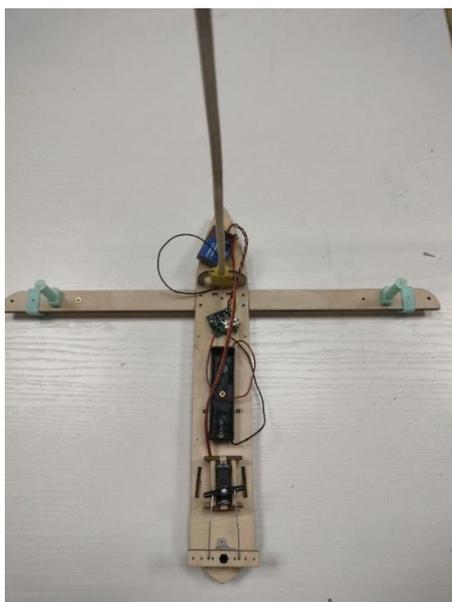


Сборка.



Парус

Далее все детали для буера были изготовлены на станке лазерной резки и напечатаны на 3D-принтере. А затем собраны в единую конструкцию.



#### Заключение

Была спроектирована и создана масштабная модель буера.

Планируется: запрограммировать систему автопилота и провести тестирование прототипа в автономном виде.

#### СПИСОК ЛИТЕРАТУРЫ

1. Буер // электронный ресурс <https://ru.wikipedia.org/wiki/%D0%91%D1%83%D0%B5%D1%80> (Дата обращения 15 марта 2023 года)
2. История Буера // электронный ресурс <https://dnrussia.ru/dn-class-info/history-ice-yacht/> (Дата обращения 15 марта 2023 года)

#### СОЗДАНИЕ ИНТЕРАКТИВНЫХ ТЕСТОВ В MS POWERPOINT

*Ходыревский Алексей*

*Муниципальное бюджетное общеобразовательное учреждение*

*«Лицей № 21», 11 класс*

*г. Курск*

Руководитель: Меньшикова Любовь Александровна, учитель информатики

Интерактивный тест один из наиболее актуальных видов тестирования на современном этапе развития информационного общества. Несмотря на большое разнообразие средств для создания интерактивных тестов, создание тестов с помощью MS PowerPoint является легко осваиваемой и доступной программой, которая входит в офисный пакет приложений Microsoft Office и не требуют дополнительной установки и затрат на приобретение. Поэтому создание интерактивного теста средствами программы MS PowerPoint, а также

оценка преимуществ и возможностей применения MS PowerPoint для создания интерактивных тестов является актуальным направлением исследования.

Разработанные в Ms PowerPoint различные виды интерактивных тестов могут эффективно применяться для проведения самоконтроля и контроля знаний и умений учащихся по предмету «Метрология, стандартизация и сертификация».

Цель исследования: создание интерактивного теста в Ms PowerPoint и доказательство возможностей и преимущества применения программы для создания интерактивных тестов.

Задачи исследования:

1. проанализировать программные средства для создания интерактивных тестов;
2. изучить возможности программы Ms PowerPoint для создания интерактивных тестов;
3. разработать и представить продукт проекта – интерактивный тест в Ms PowerPoint;
4. оценить возможности и преимущества применения программы Ms PowerPoint для создания интерактивных тестов.

Анализ имеющихся на рынке и в сети Интернет программных продуктов показывает, что создание и применение интерактивных тестов могут производиться с использованием, как специализированных программ, так и программ общего назначения [1].

В первом случае обеспечивается наибольшая простота создания и использования тестов, поскольку именно в этом и заключается назначение этих программ. Вместе с тем их применение ограничивает творчество преподавателей. Например, далеко не все программы этого класса поддерживают тесты типа «множество из множества», в некоторых случаях имеются ограничения на использование в тестах графической информации и т.д.

Кроме того, большинство специализированных программ не обеспечивают универсальности, поскольку файлы тестов сохраняются в собственном формате. Таким образом, возникает необходимость установки соответствующих программ на все ПК, используемые для проведения тестирования. Для тестирования в условиях аудитории это означает дополнительные затраты на покупку лицензии, а тестирование обучаемых в ходе самоподготовки (в домашних условиях) становится в ряде случаев невозможным.

Для создания интерактивных тестов не требуется специальных знаний. Простота изготовления и применения тестов не требует специального программного обеспечения. Существует ряд программ, в которых создаются интерактивные тесты. Рассмотрим некоторые программы, в которых создаются интерактивные тесты.

Для сравнения возможностей программных продуктов для тестирования было проанализировано двадцать четыре программы, достаточно часто используемые для создания интерактивных тестов (см. таблицу).

Предлагаемые варианты программ могут помочь в выборе программ для создания интерактивных тестов с учетом целей и задач тестирования, а также с учетом технических требований, предъявляемых к тесту. Число программ, с помощью которых можно сделать интерактивные ресурсы самостоятельно огромно. Одни из них требуют уровня продвинутого пользователя, другие доступны и начинающим пользователям.

Также можно использовать как готовые электронные тесты на дисках, так и скачанные из интернета. Однако вопросы в представленных готовых тестах могут не подойти по сложности, содержанию, количеству и т. д. Поэтому и было решено создать свой интерактивный тест. Было проанализировано много программ, в которых можно создать тесты: тестовые оболочки, Excel, PowerPoint, с помощью языка HTML, но выбор был остановлен на программе PowerPoint [2, 3].

Основой для создания разработанного интерактивного теста в MS PowerPoint являются макросы, написанные на языке программирования VBA (Visual Basic for Application). Это позволяет оптимизировать процесс создания интерактивного теста за счёт автоматизации повторяющихся действий и объединения работы нескольких команд.

Программа MS PowerPoint позволила создать интерактивный тест с различными вариантами заданий:

- Задания с выбором единственного правильного ответа.
- Задания с выбором нескольких правильных ответов.
- Задание на ввод текстового ответа.
- Задание на заполнение в тексте пропусков слов, букв или чисел.
- Задания на установление соответствий (с перемещаемыми объектами), а также на установление правильной последовательности (с перемещаемыми объектами).
- Задание на выбор активных областей.

Разработанный тест является демонстрацией возможностей приложения Microsoft PowerPoint для создания интерактивных тестов:

*Таблица – Сравнение возможностей программ для создания интерактивных тестов*

Программы тестирования	Возможности									
	Дополнительные виды тестовых вопросов, кроме	Возможности настойки	Возможности импорта	Возможности экспорта	Защита ключей теста	Бесплатное распространение	Возможности загрузки доработки модулей программы, интеграция	Возможности изменения	Настройка расписания времени проведения тестирования	Установление дополнительных возможностей

	е осно вных	ок		рез ульт та- там и		ной верс ии	ие собств енных блоков	са тест иру- емой прог рамм ы		время тести рован ия
MyTestXPro <a href="http://mytestx.pro/">http://mytestx.pro/</a>	X	X	X	X	X	-	-	-	-	-
x-TLS	-	X	X	-	X	X	X	-	-	-
INDIGO	-	X	X	X	X	-	-	X	X	-
Moodle	-	X	X	X	X	-	-	X	X	X
OpenTest	-	X	X	X	X	-	-	X	X	X
Let`s test	-	X	X	X	X	-	-	X	X	X
AnsTester <a href="http://ansoft.net.ru">http://ansoft.net.ru</a>	X	X	X	X	X	X	-	-	-	-
Tests2009 <a href="http://tests2009.ucoz.ua">http://tests2009.ucoz.ua</a>	X	X	X	X	X	X	-	-	X	-
TestBuilder <a href="http://testbuilder.narod.ru">http://testbuilder.narod.ru</a>	X	X	X	X	X	-	-	-	X	-
ADSoft Tester <a href="http://adtester.org">http://adtester.org</a>	-	-	X	-	X	X	X	-	-	-
NetTest <a href="http://kpolyakov.narod.ru">http://kpolyakov.narod.ru</a>	-	X	X	-	X	X	X	-	-	-
МастерТест <a href="http://tvy.narod.ru">http://tvy.narod.ru</a>	-	X	X	X	X	X	-	-	X	-
STS <a href="http://taalge.o.narod.ru">http://taalge.o.narod.ru</a>	-	X	X	-	X	X	X	-	-	-
TestMan <a href="http://rafsoft.narod.ru">http://rafsoft.narod.ru</a>	-	X	X	-	X	X	X	-	-	-
Surveyor	-	-	-	-	-	X	-	-	-	-

http://seesoft.ru										
RichTest http://maestro-kit.ucoz.ru	-	X	X	-	X	X	X	-	-	-
Шёлковый тест http://labfretech.org	-	X	X	X	X	-	-	-	X	-
eTest http://etest.ru	-	X	X	-	X	-	X	-	-	-
PikaTest http://kripexx.narod.ru/pikatest	-	X	X	-	X	X	X	-	-	-
Экзаменатор http://pisoft.ru	X	X	X	X	X	-	-	-	-	-
MyTest http://mytest.klyaksa.net	X	X	X	X	X	-	-	-	-	-
Knowing http://www.globalpage.ru	X	X	X	X	X	-	-	-	-	-
TestYourSelf http://www.alonewolfsoft.ru	-	X	X	-	X	-	X	-	-	-
TestDel http://gromdel.net	-	X	X	-	X	X	X	-	-	-

1. MS PowerPoint 2003 и MS PowerPoint 2007, 2010 и 2013 (как в 32-разрядных, так и в 64-разрядных версиях), используя огромные мультимедийные возможности этой среды (фото, видео и т.д.).

2. Позволяет создавать как проверочные тесты, так и обучающе-контролирующие ресурсы.

3. Количество заданий – от одного до тысячи и более.

4. Может содержать информационные слайды и слайды с заданиями:

- – с выбором единственного правильного ответа (с переключателями);
- – с выбором нескольких правильных ответов (с флажками);

- – на ввод текстового ответа;
- – на заполнение в тексте пропусков слов, букв или чисел;
- – на установление соответствий (с перемещаемыми объектами);
- – на установление правильной последовательности;
- – на выбор активных областей.

5. В любой момент разработки теста можно добавлять или удалять слайды с заданиями и информационные слайды, произвольно менять их порядок следования.

6. Количество вариантов ответов для выбора – от двух до шести, а на слайдах с перемещаемыми объектами – до десяти, и может быть различным на разных слайдах.

7. Допускается выбор шкалы оценки от 5-балльной до 100-балльной с прямым или обратным порядком.

8. Простота и единообразие задания верных ответов и настроек, в том числе выбора требовательности к оценке и учета неполных ответов при множественном выборе.

9. Можно применять шаблоны оформления и цветовые схемы.

10. Все элементы слайдов (в том числе переключатели и флажки) допускается перемещать, менять их порядок, изменять размеры, цвет контуров и заливки, форматировать шрифт, редактировать текст.

11. Допускается удаление большинства объектов на слайде, за исключением ограниченного набора, часть элементов которого может быть скрыта с помощью настроек.

12. Ведется учет времени, затраченного на прохождение теста, которое можно ограничить, включив таймер обратного отсчета. Время на информационных слайдах можно остановить.

13. До истечения времени тестирования можно вернуться к предыдущим слайдам и исправить ответ.

14. Предусмотрена возможность вывода подробных итогов тестирования в скрытый текстовый файл, что позволяет проконтролировать результаты тестирования не только на рабочем месте, но и удаленно через локальную сеть, с последующей автоматической их обработкой.

Основные преимущества MS PowerPoint для создания интерактивных тестов:

- оснащенность образовательных учреждений программами MS Office;
- оснащенность компьютерами рабочего места учителя, преподавателя, методиста и т.д.;
- относительная простота реализации эффектов анимации в среде MS PowerPoint по сравнению с Flash-технологиями или с помощью программирования в среде алгоритмических языков высокого уровня;
- широкий спектр использования анимационных эффектов.

В результате выполнения исследовательской работы:

- 1) проанализировано 24 программы для создания интерактивных тестов;

2) изучены возможности программы Ms PowerPoint для создания интерактивных тестов;

3) разработан интерактивный тест для тестирования по предмету «Метрология, стандартизация и сертификация»;

4) оценены возможности и преимущества применения программы Ms PowerPoint для создания интерактивных тестов.

Таким образом, среда Microsoft PowerPoint предоставляет прекрасные возможности для реализации тестового контроля знаний. С использованием среды программирования Visual Basic for Applications возможно реализовывать различные формы заданий в тестовой форме.

Кроме этого, тесты, реализованные в среде Microsoft PowerPoint, соответствуют таким требованиям как: удобный внешний вид; простота процесса тестирования; объективное оценивание тестовых результатов; снижение затрат времени на проверку знаний.

В данной среде возможно создавать тестовые задания по любым дисциплинам для обучающихся любого возраста.

Нами был разработан интерактивный тест в программе Microsoft PowerPoint по дисциплине «Метрология, стандартизация и сертификация». Ознакомиться с видео демонстрацией разработанного интерактивного теста можно по ссылке [https://drive.google.com/file/d/1piagP7eV3nzQVSd5wJ89crZ1xaRt5\\_8l/view](https://drive.google.com/file/d/1piagP7eV3nzQVSd5wJ89crZ1xaRt5_8l/view).

#### СПИСОК ЛИТЕРАТУРЫ

1. Бойко Г., Зотов Н., Полуэктов М. Классификация и особенности создания электронных тестов // Высшее образование в России. 2008. №12. URL: <https://cyberleninka.ru/article/n/klassifikatsiya-i-osobennosti-sozdaniya-elektronnyh-testov> (дата обращения: 19.04.2021).
2. Берман Н. Д. MS PowerPoint 2010: учебное пособие / Н. Д. Берман, Т. А. Бочарова, Н. И. Шадрин. – Хабаровск: Изд-во Тихоокеан. гос. ун-та, 2013. – 88 с.
3. Базовый курс PowerPoint. Изучаем Microsoft Office. - М.: Современная школа, 2015. - 669 с. 15. Кокс, Джойс Microsoft PowerPoint 2013. Русская версия / Джойс Кокс, Джоан Ламберт. - М.: ЭКОМ Паблишерз, 2016. - 496 с.

#### **РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ ОПРЕДЕЛЕНИЯ СЕРДЕЧНО-СОСУДИСТЫХ ЗАБОЛЕВАНИЙ С ПРИМЕНЕНИЕМ ОСНОВ МАШИННОГО ОБУЧЕНИЯ**

*Питецкий Михаил Константинович, Хрипачев Иван Александрович*  
*КГАОУ Школа Космонавтики, 10 класс*  
*Железногорск*

Руководитель: Вичканова Ирина Александровна, магистр 1 курса, ИШНКБ,  
НИ ТПУ

На сегодняшний день искусственный интеллект (далее ИИ) находит широкое применение в повседневной и профессиональной сферах человеческой жизни. С 2005 по 2020 год внедрение технологий искусственного интеллекта в медицинские процессы выросло почти в 62 раза, что говорит об успешном внедрении и отличных результатах работы ИИ [1]. Данная работа направлена на определение сердечно-сосудистых заболеваний с применением основ машинного обучения. Работа заключается в реализации программного кода для обучения математической модели на основе обработки тестовых сигналов ЭКГ, взятых из общей базы биологических сигналов [2]. После обучения модель способна прогнозировать результаты анализа реальных ЭКГ сигналов самостоятельно.

Ключевые слова: Искусственный интеллект, ЭКГ, машинное обучение, сердечно-сосудистые заболевания.

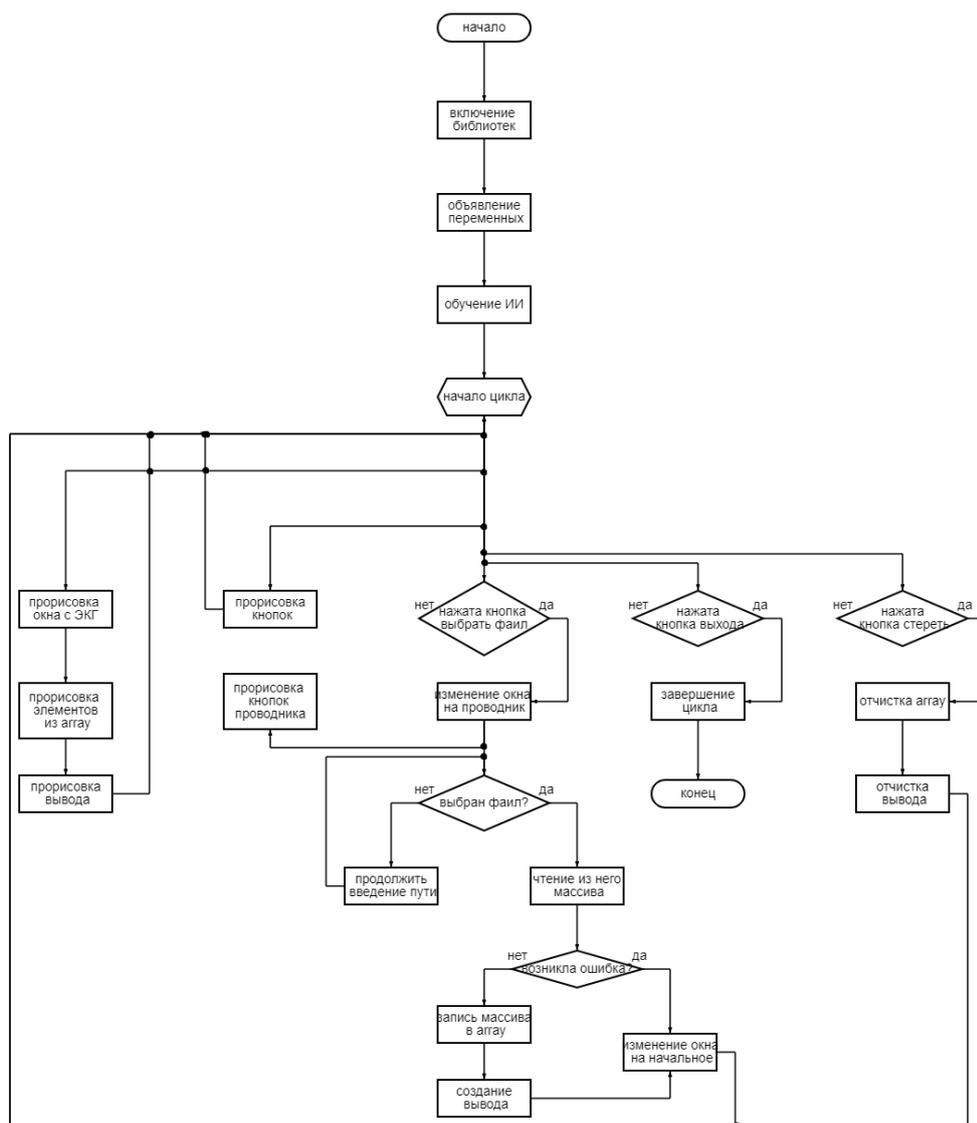
По данным Росстата, за 2017-2018 год более миллиона человек умирают от различных заболеваний сердечно-сосудистой системы. Для получения неотложной помощи в случае таких заболеваний важно быстро и качественно поставить диагноз пациенту, основанный, чаще всего, на ЭКГ. Для этого мы реализовали программный код, который облегчит и ускорит работу кардиологов и, вместе с этим, получение пациентами результатов, что в некоторых случаях может спасти их жизни. Для обучения ИИ мы используем библиотеку `skit learn`, который, в отличие от аналогов, прост в написании кода и добавляет множество различных способов обучения ИИ, и линейную модель обучения логистической регрессии. Логистическая регрессия является линейным классификатором и из-за этого наиболее эффективна в данном случае.

Разработанная программа отличается от подобной услуги, предоставляемой компанией, например, Сбером тем, что она имеет удобный интерфейс для пользователя, не засоренный лишними кнопками и функциями, которые только запутают человека, работающего с ними [3]. Так как в России многие компании работают над созданием приложений для телемедицины, представленное в данной статье приложение может стать конкурентоспособным на российском рынке.

При выполнении работы данные с ЭКГ сигналами были взяты из открытой базы данных PhysioNet. Далее была выбрана и изучена подходящая библиотека для работы с ИИ. С помощью библиотеки (`ScikitLearn`) был реализован программный код для определения патологии через исследуемый ЭКГ сигнал. С помощью графического редактора `Paint` были реализованы фон и изображения. С помощью библиотеки `PyGame` был отрисован дизайн приложения. Заказчиком данной работы может являться Министерство Здравоохранения и частные здравоохранительные организации. Данная работа отличается от остальных применением новейших алгоритмов для обучения ИИ.

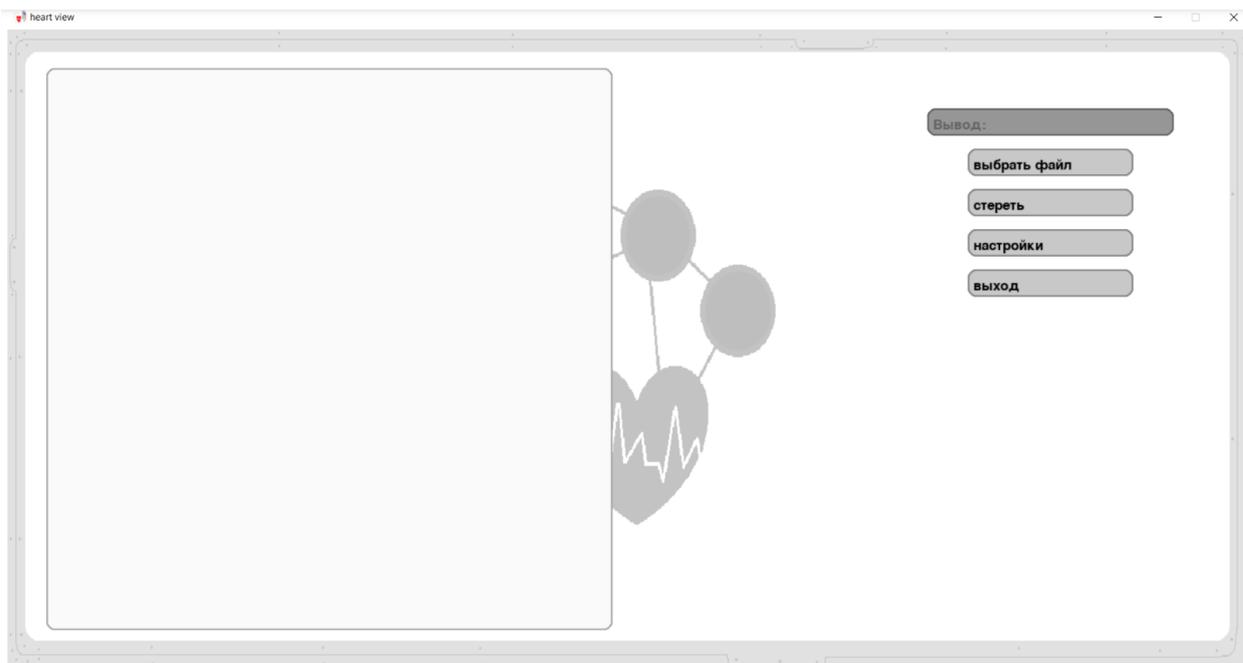
Потребителем может быть каждый, кто хоть раз обследовался в кабинете кардиолога.

Алгоритм выполнения программного кода для определения диагноза с помощью разработанного приложения представлен на рисунке 1.

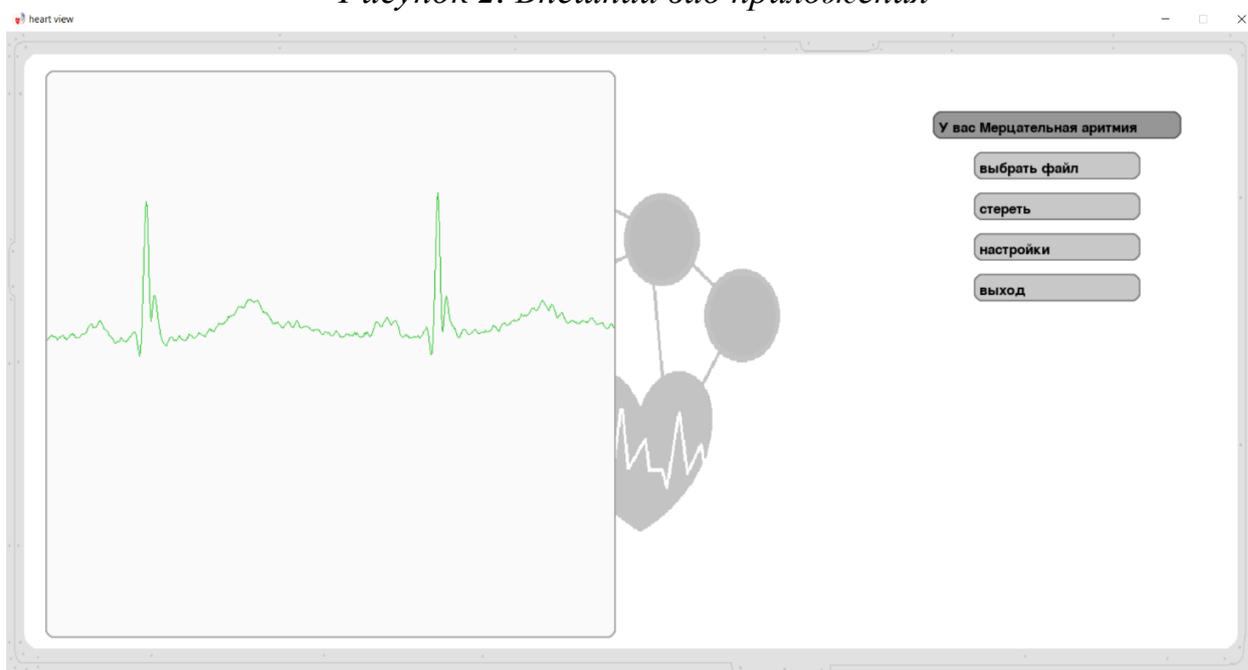


*Рисунок 1. Алгоритм выполнения программного кода*

Внешний вид приложения и представление диагноза представлены на рисунках 2,3 соответственно.



*Рисунок 2. Внешний вид приложения*



*Рисунок 3. Результаты обработки тестового сигнала ЭКГ*

Небольшое отличие построенного ЭКГ-сигнала от настоящего можно объяснить недостаточным количеством данных. Проблема решится, если в дальнейшем обучении модели использовать данные реальных ЭКГ-сигналов, а не усреднённой.

Данный проект необходимо продолжать, поскольку он имеет огромный потенциал не только для определения диагнозов по результатам ЭКГ, но также может стать общедоступной и надёжной платформой, где будут храниться как недавние анализы пациента, так и история его болезней, которая может быть автоматически определена и занесена в базу. Это поможет улучшить качество предоставляемых больницами услуг, потому что по результатам исследований ЕвроМед, программа определяет диагноз более точно и может спасти больше

людей, чем обычный врач, который может, например, не заметить что-нибудь [4].

#### СПИСОК ЛИТЕРАТУРЫ

1. Искусственный интеллект в медицине: сферы, технологии и перспективы [Электронный ресурс] / URL: [www.habr.com/ru/company/first/blog/682516/](http://www.habr.com/ru/company/first/blog/682516/), свободный. – Загл. с экрана. – Яз. рус. Дата обращения: 9.03.23
2. PhysioNet. The Research Resource for Complex Physiologic Signals [Электронный ресурс] / URL: [www.physionet.org/](http://www.physionet.org/), свободный. – Загл. с экрана. – Яз. англ. Дата обращения: 9.03.23
3. Сайт ИИ от Сбера: [Электронный ресурс]/URL: <https://sbermed.ai/iskusstvenniy-intellekt-v-analize-ekg/>, свободный.-Загл. с экрана- яз.рус. Дата обращения: 14.03.23
4. Исследование ЕвроМед: [Электронный ресурс]/URL: <https://euromed.ru/news/puls-vremeni-iskusstvennyj-intellekt-v-kardiologii/>, свободный.-Загл. с экрана- яз.рус. Дата обращения: 14.03.23

#### АЛГОРИТМ МОНТЕ-КАРЛО В КРЕСТИКАХ НОЛИКАХ

*Хузеев Кирилл Юрьевич, Бакеев Тимур Рустамович*

*Муниципальное бюджетное общеобразовательное учреждение лицей при  
ТПУ г. Томска, 10 класс*

Руководитель: Кривошеев Николай Анатольевич, ассистент ОИТ ИШИТР

Актуальность: крестики-нолики - это одна из самых простых и популярных игр, которая часто используется в качестве тестовой задачи для различных алгоритмов и методов искусственного интеллекта. Одним из таких алгоритмов является алгоритм Монте-Карло, который позволяет находить оптимальные ходы в игре на основе случайных выборок.

Цель работы: разработка и тестирование алгоритма Монте-Карло для игры в крестики-нолики. Для достижения этой цели необходимо изучить алгоритм Монте-Карло, а также разработать и протестировать свой собственный алгоритм. Результаты работы могут быть использованы для улучшения алгоритмов искусственного интеллекта в области игр.

Задачи:

1. Изучить основы алгоритма Монте-Карло и его применение в играх.
2. Разработать и реализовать алгоритм для игры в крестики-нолики на основе случайного поиска.
3. Разработать и реализовать алгоритм Монте-Карло для игры в крестики-нолики.
4. Провести эксперименты для оценки эффективности и точности алгоритма.
5. Сравнить результаты работы алгоритма с другими алгоритмами, используемыми в играх.

- б. Проанализировать полученные результаты и сделать выводы о применимости алгоритма Монте-Карло в играх и других областях, где требуется принимать решения на основе вероятностных расчетов.

Теоретическая часть.

Обзор существующих решений задачи:

Одним из самых простых решений является использование алгоритма минимакс, который позволяет компьютеру просчитывать все возможные ходы и выбирать оптимальный вариант. Однако, этот алгоритм может быть сложным для понимания и реализации для начинающих программистов.

Случайный поиск:

В случайном поиске компьютер делает случайные ходы в игре, пока не найдет выигршный вариант или не закончатся все свободные ячейки на поле. Этот метод прост в реализации, но не всегда эффективен и может не давать оптимального результата.

Алгоритм Монте-Карло:

Алгоритм Монте-Карло - это метод численного моделирования, который использует случайные числа для решения задач. В контексте игры в крестики-нолики, алгоритм Монте-Карло используется для выбора оптимального хода компьютера.

Алгоритм состоит из следующих шагов:

1. Компьютер делает случайный ход на доске.
2. Затем производится случайный выбор ходов, чтобы определить, какая из оставшихся ячеек будет заполнена следующей.
3. Игра продолжается до конца, и результат сохраняется.
4. Шаги 1-3 повторяются множество раз.
5. Компьютер выбирает ход, который дал наибольшую вероятность победы.

Разработка алгоритма Монте-Карло для игры в крестики-нолики.

Для разработки алгоритма Монте-Карло для игры в крестики-нолики необходимо выполнить следующие шаги:

1. Определить текущее состояние игрового поля. Это позволит определить доступные ходы для компьютера.
2. Сгенерировать случайные ходы на основе доступных ходов, чтобы симитировать несколько возможных игр.
3. Запустить симуляцию игр на основе случайных ходов, чтобы определить вероятность победы для каждого из возможных ходов.
4. Выбрать оптимальный ход на основе вероятности победы, рассчитанной на предыдущем шаге.
5. Продолжать игру, повторяя шаги 1-4 до тех пор, пока не будет достигнуто окончательное состояние игры.

Пример работы алгоритма Монте-Карло для игры в крестики-нолики:

1. Начальное состояние игрового поля определено.
2. Сгенерировано несколько случайных ходов на основе доступных ходов.

3. Для каждой из симулированных игр была определена вероятность победы для каждого из возможных ходов.
4. Оптимальный ход был выбран на основе вероятности победы, рассчитанной на предыдущем шаге.
5. Игра продолжается, повторяя шаги 1-4 до тех пор, пока не будет достигнуто окончательное состояние игры.

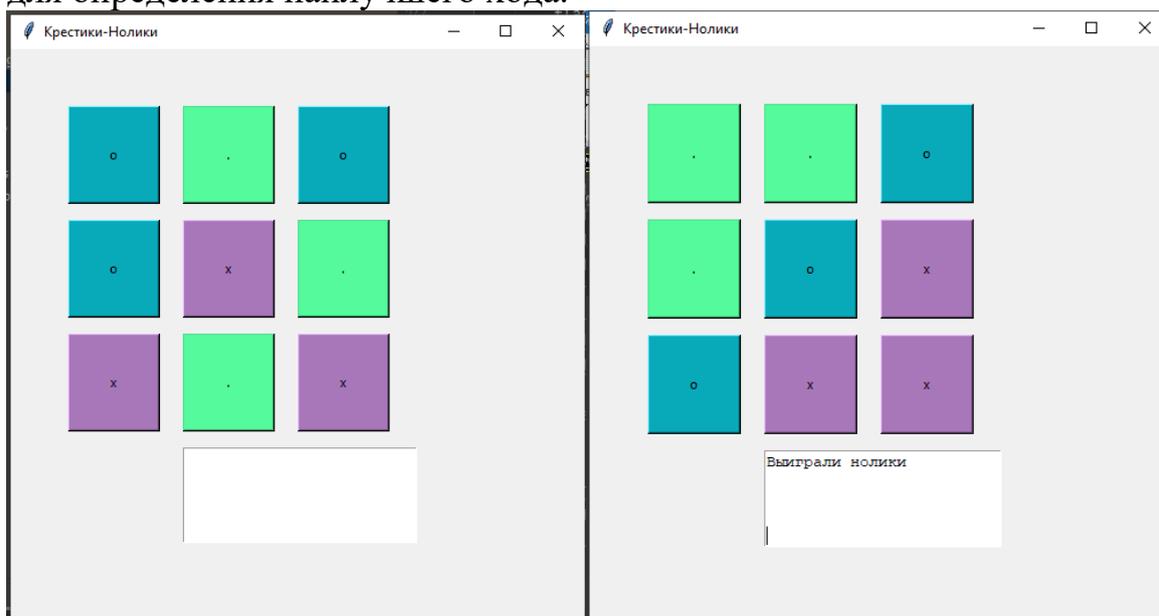
Анализ результатов показывает, что алгоритм Монте-Карло для игры в крестики-нолики является эффективным и может быть использован для создания компьютерных игр. Он позволяет компьютеру выбирать наиболее оптимальный ход на основе вероятности победы, что увеличивает шансы на победу в игре. Однако, чтобы алгоритм работал наилучшим образом, необходимо подобрать оптимальные параметры, такие как количество симулируемых игр и количество случайных ходов, которые должны быть выполнены в каждой игре.

Реализация графического интерфейса.

Для реализации графического интерфейса для игры в крестики-нолики с использованием алгоритма Монте-Карло можно использовать различные библиотеки для создания пользовательского интерфейса на языке программирования Python, например, PyQt, Tkinter, wxPython и др.

Одной из наиболее распространенных библиотек для создания графических интерфейсов на Python является Tkinter. Она позволяет создавать красивые и функциональные пользовательские интерфейсы с помощью графических элементов, таких как кнопки, поля ввода, таблицы и др.

Результатом графического интерфейса для игры в крестики-нолики с использованием алгоритма Монте-Карло будет являться окно, в котором будет отображаться игровое поле, на котором пользователь сможет делать свои ходы, а компьютер будет делать свои ходы, используя алгоритм Монте-Карло для определения наилучшего хода.



Такой интерфейс будет обеспечивать удобство и комфорт при игре в крестики-нолики, а использование алгоритма Монте-Карло позволит компьютеру играть более эффективно и принимать лучшие решения в игре.

Заключение

В заключении можно сделать общие выводы о проделанной работе по теме "Алгоритм Монте-Карло в крестиках-ноликах". Были рассмотрены основные шаги алгоритма Монте-Карло, включая определение текущего состояния игры, генерацию случайных ходов, запуск симуляций на случайных ходах для определения вероятностей победы, выбор хода с наибольшей вероятностью победы и продолжение игры до достижения конца игры.

Возможными направлениями дальнейшей работы по теме "Алгоритм Монте-Карло в крестиках-ноликах" могут быть:

1. Разработка более сложных игровых стратегий на основе алгоритма Монте-Карло, например, с использованием машинного обучения.
2. Разработка графического интерфейса для игры в крестики-нолики с использованием алгоритма Монте-Карло, который позволит пользователям играть против компьютера.
3. Оптимизация параметров алгоритма Монте-Карло для достижения наилучших результатов, таких как количество симуляций и случайных ходов, которые необходимо выполнить в каждой игре.

## МЕТОДЫ ОБРАБОТКИ ИЗОБРАЖЕНИЙ

*Хусаинов Ильяс*

*Муниципальное бюджетное общеобразовательное учреждение лицей при  
ТПУ г. Томска, 10 класс*

**Руководитель:** Кривошеев Николай Анатольевич

В условиях стремительного развития интернета, а вследствие и увеличения количества данных их обработка является главной проблемой. Важной задачей же ученых состоит оптимизировать и уменьшить затраты на этот процесс.

Существует несколько методов обработки изображений, мы будем рассматривать сегментацию и классификацию, отличительной особенностью их является то, что человек не участвует в самой работе, или принимает минимальное участие. Ручная обработка данных в больших объемах занимает огромное количество времени.

Целью работы было выявление лучшего метода обработки изображений.

Задачи:

- Реализовать алгоритм ближайшего соседа и нейронную сеть на библиотеке Keras для задач классификации.
- Сравнить скорость и качество работы этих алгоритмов на задачах классификации.
- Реализовать сверточную нейронную сеть для задач сегментации.

Для сравнения качества работы наших алгоритмов необходимы данные, с которыми мы будем работать. Для этого мы будем использовать базы данных, созданных специально для этого. Это нереальные базы данных, с которыми мы сталкиваемся, но благодаря им, можно на примере посмотреть работу наших алгоритмов и в будущем использовать их на реальных задачах.

При написании кода программ была использована информация из источников [1-2].

Алгоритм ближайшего соседа.

На начальном этапе мы использовали базу данных mnist. База данных содержит 70000 изображений, сами изображения — это черно-белые цифры от 0 до 9 размерами 28 на 28 пикселей.



Рисунок 1. Визуальное представление базы данных mnist

Алгоритм ближайшего соседа считается простым методом классификации изображений. Сам алгоритм заключается в нахождении расстояния между пикселями изображений. Мы искали его с помощью евклидова расстояния.

Для того чтобы проверить эффективность работы алгоритма, мы взяли 100 изображений и каждый проводили через этот алгоритм. Если он правильно определял цифру, то точность росла, если нет, то падала. Результат показал 83% точности, это значит, что шанс того, что алгоритм правильно определит на изображении цифру равен примерно 83%.

Далее мы будем использовать более сложные базы данных для обработки. Cifar-10 это база данных представляющая собой набор изображений, которые чаще всего используют для решения задач классификации. Она содержит 60000 цветные изображений размерами 32 на 32 пикселя.

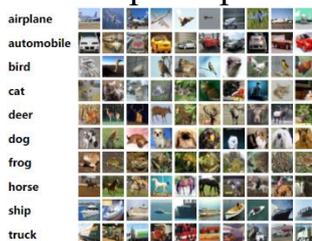


Рисунок 2. Визуальное представление базы данных cifar-10

Алгоритм ближайшего соседа несмотря на то, что и сможет правильно классифицировать такие изображение, однако, его точность будет крайне мала, так как, мы просто искали максимально схожие изображения по пикселям, и отличить кота от собаки он не сможет, в следствии такие базы данных как cifar-10 он обработать не может.

Переходя к более сложным задачам классификации, обычно прибегают к машинному обучению.

Нейронные сети

Понятие, нейронные сети, появилось благодаря изучению процессов, происходящих в коре головного мозга, ведь оно, построено по принципу работы головного мозга. Так понятие, без которого невозможно представить программирование пришло прямиком из биологии.

Нейронная сеть — это последовательность нейронов, соединенных между собой синапсами. Создается адаптивная система, с помощью которой компьютер учится на своих ошибках и постоянно совершенствуется. Таким образом, искусственные нейронные сети и пытаются решать проблемы, с которыми человек не смог бы справиться, с минимальным участием самого человека.

Нейронная сеть состоит из нескольких слоев: входной, скрытый, выходной. Каждый из слоев же состоит из нейронов, которые принимают взвешенную сумму входных данных, а после выдают эти данные, принимая на них функцию активации. Взвешенную, потому что нейроны соединены с помощью синапсов, а они в свою очередь имеют свой вес, который меняется в процессе обучения.

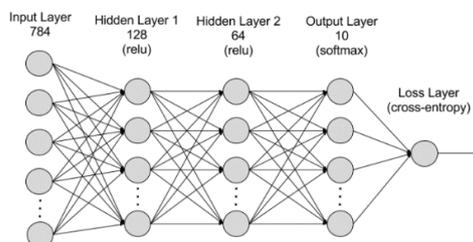


Рисунок 3. Структура нейронной сети

Нейронная сеть на keras.

Keras — библиотека, написанная на языке python и обеспечивающая взаимодействие с ИНС.

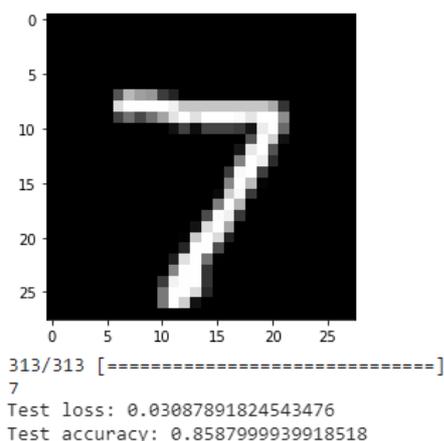
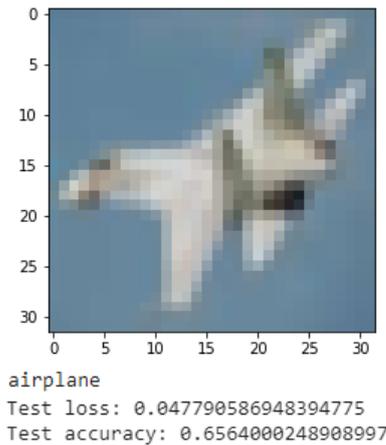


Рисунок 4. Результат классификации на mnist

На рисунке 4 мы видим изображение, подающееся обученной нейронной сети и число, которое распознала нейронная сеть. Для численных оценок результатов классификации использовалась встроенная в код функция тестирования. Результат точности классификации составляет 85%. Это не максимально возможный результат, и, хотя разница с алгоритмом ближайшего соседа не большая, выигрыш в скорости работы огромный.



*Рисунок 5. Результат классификации на cifar-10*

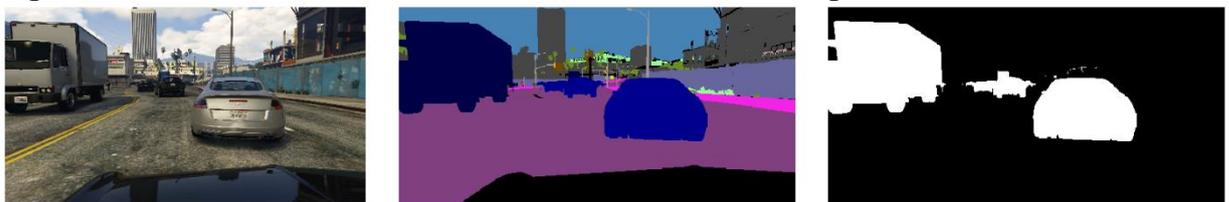
На рисунке 5, аналогично, мы видим изображение и результат, объект, который распознала нейронная сеть. Результат классификации находится ниже и здесь мы видим крайне плохие результаты по сравнению с предыдущей задачей. Это вызвано тем, что мы никак не меняли архитектуру нашей нейронной сети.

Теперь мы знаем, как работают нейронные сети и машинное обучение, но их используют не только для задач классификации. Сегментация – это еще один вид обработки изображения, только теперь мы разделяем ее на несколько сегментов. Необходимо это для упрощения анализа изображения, обычно он выделяет границы объекта или сам объект на изображении.

#### Задачи сегментации

Прежде чем подходить к задачам сегментации необходимо подготовить нашу выборку. Изначально выборка представляет собой огромный массив изображений, сделанных в компьютерной игре, а также уже обработанные изображения, на которых выделены различные объекты разными цветами. База данных скачана из источника [5].

Наша первостепенная задача — это оставить на картинке только машины остальное перекрасить в черный цвет, также встает проблема, что размеры картинок большие и работать с ними тяжело, поэтому перед дальнейшими манипуляциями с изображениями необходимо было их уменьшить в 8 раз. На рисунке 6 представлена полная работа с одним из изображений выборки, такое необходимо было сделать со всей выборкой.

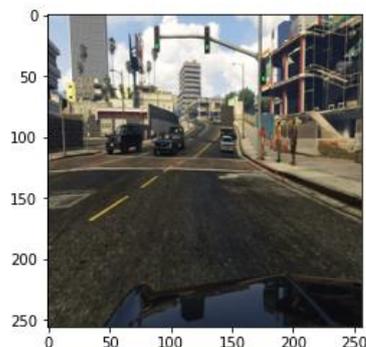


*Рисунок 6. Процесс обработки одного изображения*

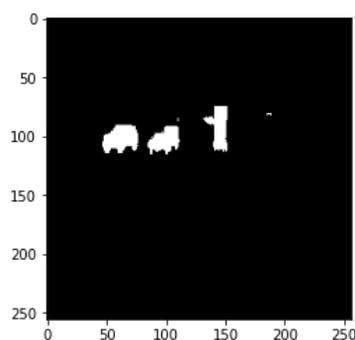
Теперь, когда наша выборка готова можно приступать к обучению нашей нейронной сети. Ей необходимо распознавать на изображении машины. Если мы будем использовать старую архитектуру нашей нейронной сети, как для задач классификации, процесс обучения будет долгим и не эффективным.

Несмотря на то, что рассматриваемая нейронная сеть с глубоким обучением показывала хороший результат на задачах классификации, для сегментации этого недостаточно, поэтому обычно используют сверточные нейронные сети. Для нашего проекта мы использовали архитектуру U-Net. Информация по написанию кода для сверточной нейронной сети была взята из источников [2-4].

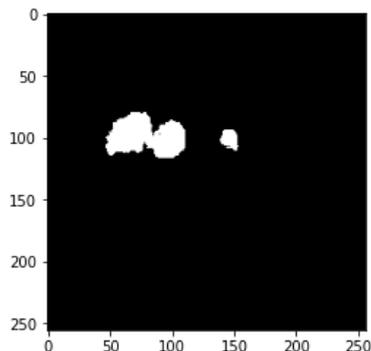
Результаты сегментации. Результаты сегментации тестовых изображений представлены на рисунках 7-9.



*Рисунок 7. Исходное тестовое изображение.*



*Рисунок 8. Сегментированное тестовое изображение.*



*Рисунок 9. Результаты сегментации тестового изображения нейронной сетью.*

В результате была создана сверточная нейронная сеть, реализующая обработку сегментацией изображений. Созданная программа с достаточной точностью способна искать на изображении машины.

Выводы проекта:

1. Реализовали алгоритм ближайшего соседа и нейронную сеть на keras.

2. Сравнение показателей точности работы и времени работы двух алгоритмов показали, что на задачах классификации нейронные сети показывают себя лучше в сравнении с алгоритмом ближайшего соседа.
3. Реализовали сверточную нейронную сеть для сегментации изображений.

#### СПИСОК ЛИТЕРАТУРЫ

1. Документация к библиотеке Keras. [Электронный ресурс]. – URL: <https://keras.io/api/> (дата обращения: 16.12.2022)
2. Стэнфордский курс: лекция 2. Классификация изображений – URL: [https://www.reg.ru/blog/stehnfordskij-kurs-lekciya-2-klassifikaciya-izobrazhenij/?utm\\_source=google.com&utm\\_medium=organic&utm\\_campaign=google.com&utm\\_referrer=google.com/](https://www.reg.ru/blog/stehnfordskij-kurs-lekciya-2-klassifikaciya-izobrazhenij/?utm_source=google.com&utm_medium=organic&utm_campaign=google.com&utm_referrer=google.com/) (дата обращения: 12.01.2023)
3. Погружение в сверточные нейронные сети. [Электронный ресурс]. URL: <https://habr.com/ru/post/456740/> (дата обращения: 27.02.2023)
4. U-Net Image Segmentation in Keras. [Электронный ресурс]. URL: <https://pyimagesearch.com/2022/02/21/u-net-image-segmentation-in-keras/> (дата обращения: 1.03.2023)
5. Датасет для обучения нейросети. [Электронный ресурс]. – URL: [https://download.visinf.tu-darmstadt.de/data/from\\_games/](https://download.visinf.tu-darmstadt.de/data/from_games/) (дата обращения: 5.03.2023)

### **РЕАЛИЗАЦИЯ ЭЛЕКТРОННОЙ ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ УПРАВЛЕНИЯ ПОМЕЩЕНИЕМ НА БАЗЕ УЧЕБНОГО КЛАССА СРЕДНЕЙ ШКОЛЫ**

*Цыгин Максим Леонидович*

*МБОУ Гимназия №1 города Белово*

*г. Белово*

Научный руководитель: Цыгин Леонид Владимирович

МБОУ гимназия №1, Кемеровская область, г.Белово

Качественное и надежное обеспечение нового уровня жизни и удовлетворение потребностей жителей Кемеровской области, реализация интересов России, Сибирского Федерального округа и Кузбасса в сферах социального, культурного, научно-образовательного и экологически ориентированного промышленного развития на долгосрочную перспективу потребовало обоснованной долгосрочной Стратегии. В качестве основополагающих начал разработки и реализации данной Стратегии до 2035 года и на более длительную перспективу, для обеспечения воплощения в жизнь ее стратегических приоритетов обоснованы и использованы следующие принципы такие как безопасность и надежность, социальная ответственность, экологичность и здоровье населения; качество жизни, продукции и услуг; цифровизация и умная специализация. Наш проект является одним из шагов, направленных на достижение поставленных целей. С каждым годом технологии

«умного дома» завоевывают все больше и больше поклонников. И это не удивительно, так как эти технологии подразумевают комплексную автоматизацию дома, обеспечивая максимальный комфорт, безопасность и ресурсосбережение для всех пользователей.

Актуальность данной работы обусловлена тем, что технологии «Умного дома» направлены, в основном, на жилые строения. Лишь небольшую часть «интеллектуальных» помещений составляют коммерческие и производственные объекты, и практически совсем нет учебных классов и школьных зданий. Но «умные классы» - не погоня за веяниями моды, а выгодное приобретение, которое поможет сэкономить затраты на эксплуатацию помещения примерно на 30% , а также окажет здоровьесберегающее воздействие на школьников и педагогов.

Целью работы является создание эффективной системы освещения и контроля за качеством воздуха в учебном классе за счет внедрения контекстно-зависимого алгоритма управления.

Для достижения цели необходимо решить следующие задачи:

- Разработать алгоритм управления освещением и качеством воздуха;
- Получить навыки программирования настроек «умного дома»
- Создать действующий макет «умного класса».

Практическая значимость работы заключается в разработке алгоритма контекстно-зависимой системы управления освещением и контролем качества воздуха и его аппаратной реализации. В результате проделанной работы мы выполнили все поставленные цели и задачи.

Нами изучено достаточное количество литературы для того, чтобы самостоятельно, что называется, «с нуля» создать действующий макет учебного класса, управляемого сервером. Изучены возможности платформы OpenHAB для автоматизации комфортной среды и здоровьесбережения учащихся и педагогов, а также создан алгоритм, проверяющий соответствие параметров освещенности и качества воздуха нормам ГОСТ. В ходе выполнения работы мы изучили характеристики используемых для создания «умных домов» датчиков и силовых блоков, что позволило сделать осознанный выбор необходимых для проекта компонентов.

Известно, что ученик, избавленный от вредных воздействий слабой освещенности и высокого содержания CO<sub>2</sub> в воздухе, намного лучше усваивает материал, который передает учитель. Спокойная музыка для релаксации — это отличное средство, к которому человек подсознательно прибегает, чтобы снять напряжение или отвлечься. Уже один только этот факт, что миллионы людей знают о чудесных свойствах звуков восстанавливать равновесие организма на физическом и психическом уровнях, доказывает нам практическую пользу прослушивания музыки после напряженного умственного труда.

Наш проект – только первый шаг к наиболее комфортному учебному процессу.

Мы верим, что в Кузбассе и в других регионах уже совсем скоро появятся школы, жилые комплексы или даже микрорайоны, в которых изначально, на этапе строительства будет закладываться основа под установку систем «умный дом».

#### СПИСОК ЛИТЕРАТУРЫ

1. Богданов С.В. Умный Дом. Изд. 2е, перераб. и доп.— СПб.: Наука и Техника, 2005. — 208 стр.: ил.
2. Вайгенд М. Raspberry Pi для детей/пер. с нем. Ю.Ю. Энглерт.- М.: ДМК Пресс, 2019. – 564 с.: ил.
3. Гололобов В. Н. Raspberry Pi для любознательных. - Изд. 2е, перераб. и доп.—СПб.: Наука и Техника, 2019
4. Макаров С. Л. Arduino Uno и Raspberry Pi 3: от схемотехники к интернету вещей.- М.: ДМК Пресс, 2018. – 204 с.
5. Харке В. Умный дом. Объединение в сеть бытовой техники и систем коммуникации в жилищном строительстве. Москва: Техносфера, 2006.-288с.

#### ОБНАРУЖЕНИЕ АВТОМОБИЛЕЙ И ДОРОЖНОЙ РАЗМЕТКИ НА КАДРАХ ВИДЕОПОСЛЕДОВАТЕЛЬНОСТИ

*Чурсин Алексей Андреевич*

*Муниципальное бюджетное общеобразовательное учреждение лицей при  
ТПУ, 10 класс  
г Томска*

Руководитель: Кривошеев Николай Анатольевич, аспирант ТПУ

Беспилотный транспорт обладает большим потенциалом и развитие систем автономного вождения является перспективным направлением. Интеллектуальная аналитика видеоданных является важной частью программного обеспечения беспилотных автомобилей. Одной из задач аналитики является обнаружение автомобилей и дорожной разметки на кадрах видеопоследовательности. [3]

Цель проекта: разработка системы обнаружения автомобилей и дорожной разметки на видеопоследовательности.

Задачи:

- Изучить алгоритмы сегментации изображений для обнаружения объектов на кадрах видеопоследовательности.
- Реализовать алгоритм сегментации изображения.
- Разработать алгоритм сегментации сегментации автомобилей и дорожной разметки на основе реализованного алгоритма.

Задача сегментации изображения — задача поиска групп пикселей на изображении, каждая из которых характеризует один смысловой класс [8]. Пример работы алгоритма сегментации представлен на Рисунке 1.



Рисунок 1. Пример работы алгоритма сегментации изображения [8].

Большинство современных решений задачи сегментации изображения, показывающих высокие метрики качества работы, используют нейронные сети, так как у нейронных сетей лучше проявляется способность работать в разных обстановках [4].

Нейронная сеть – это математическая модель, построенная по принципу организации и функционирования биологических нейронных сетей. Основной структурной и функциональной единицей нейронной сети является нейрон. На Рисунке 2 представлена модель нейрона. Нейрон принимает входные сигналы  $x_0, x_1, \dots, x_n$ , обрабатывает их с помощью весовых коэффициентов  $w_0, w_1, \dots, w_n$  и подает на выход сигнал  $y$ .

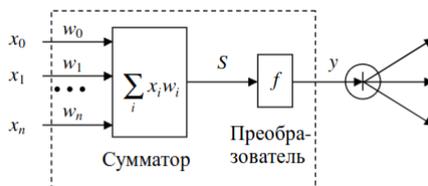


Рисунок 2. Модель нейрона.

Нейроны могут быть объединены так, что выходные сигналы одних нейронов становятся входными для других, образуя искусственную нейронную сеть. В искусственную нейронную сеть входят три типа нейронов:

- Входные нейроны, принимающие входные сигналы для всей сети;
- Выходные нейроны, представляющие результирующие сигналы сети;
- Скрытые нейроны, составляющие скрытые слои сети [9].

Одним из методов настройки весовых коэффициентов нейронов в сети является обучение с учителем, которое заключается в минимизации функции потерь с использованием обучающей выборки. В процессе обучения весовые коэффициенты настраиваются таким образом, что функция потерь стремится к минимуму, а значения метрик качества увеличиваются. [9]

Когда обучение завершено, нейронная сеть тестируется с набором данных, который не был использован ранее. Эффективность нейронной сети на этом тесте более важна, чем на тренировочном, т. к. она демонстрирует ее способность делать выводы и работать с данными, с которыми она встретилась впервые. [1]

На сегодняшний день для решения задачи сегментации изображения наиболее популярны следующие нейронные сети: U-NET – симметричная нейронная сеть, состоящая из равного количества сверточных и разверточных слоёв [6]; серия моделей DeepLab [2] и др.

Для оценки качества решения задачи сегментации изображения могут быть использованы следующие метрики качества:

$$\text{точность (precision)} = \frac{TP}{TP+FP},$$

$$\text{полнота (recall)} = \frac{TP}{TP + FN},$$

где TP – это число верно обнаруженных пикселей класса, FP – число ложно обнаруженных пикселей класса, FN – число необнаруженных пикселей класса [5].

#### Экспериментальная часть

Для решения поставленной задачи была реализована нейронная сеть структуры U-NET. На Рисунках 3 и 4 представлен код реализации сети.

```
def double_conv_block(x, n_filters):
    x = layers.Conv2D(n_filters, 3, padding = "same", activation = "relu", kernel_initializer = "he_normal")(x)
    x = layers.Conv2D(n_filters, 3, padding = "same", activation = "relu", kernel_initializer = "he_normal")(x)
    return x

def downsample_block(x, n_filters):
    f = double_conv_block(x, n_filters)
    p = layers.MaxPool2D(2)(f)
    return f, p

def upsample_block(x, conv_features, n_filters):
    x = layers.Conv2DTranspose(n_filters, 3, 2, padding="same")(x)
    x = layers.concatenate([x, conv_features])
    x = double_conv_block(x, n_filters)
    return x
```

Рисунок 3. Методы, используемые при построении архитектуры нейронной сети.

```
inputs = layers.Input(shape)

f1, p1 = downsample_block(inputs, 8)
f2, p2 = downsample_block(p1, 16)
f3, p3 = downsample_block(p2, 32)
f4, p4 = downsample_block(p3, 64)

bottleneck = double_conv_block(p4, 128)

u6 = upsample_block(bottleneck, f4, 64)
u7 = upsample_block(u6, f3, 32)
u8 = upsample_block(u7, f2, 16)
u9 = upsample_block(u8, f1, 8)

outputs = layers.Conv2D(1, 3, padding="same", activation = "sigmoid")(u9)
model = keras.Model(inputs, outputs, name="U-Net")
```

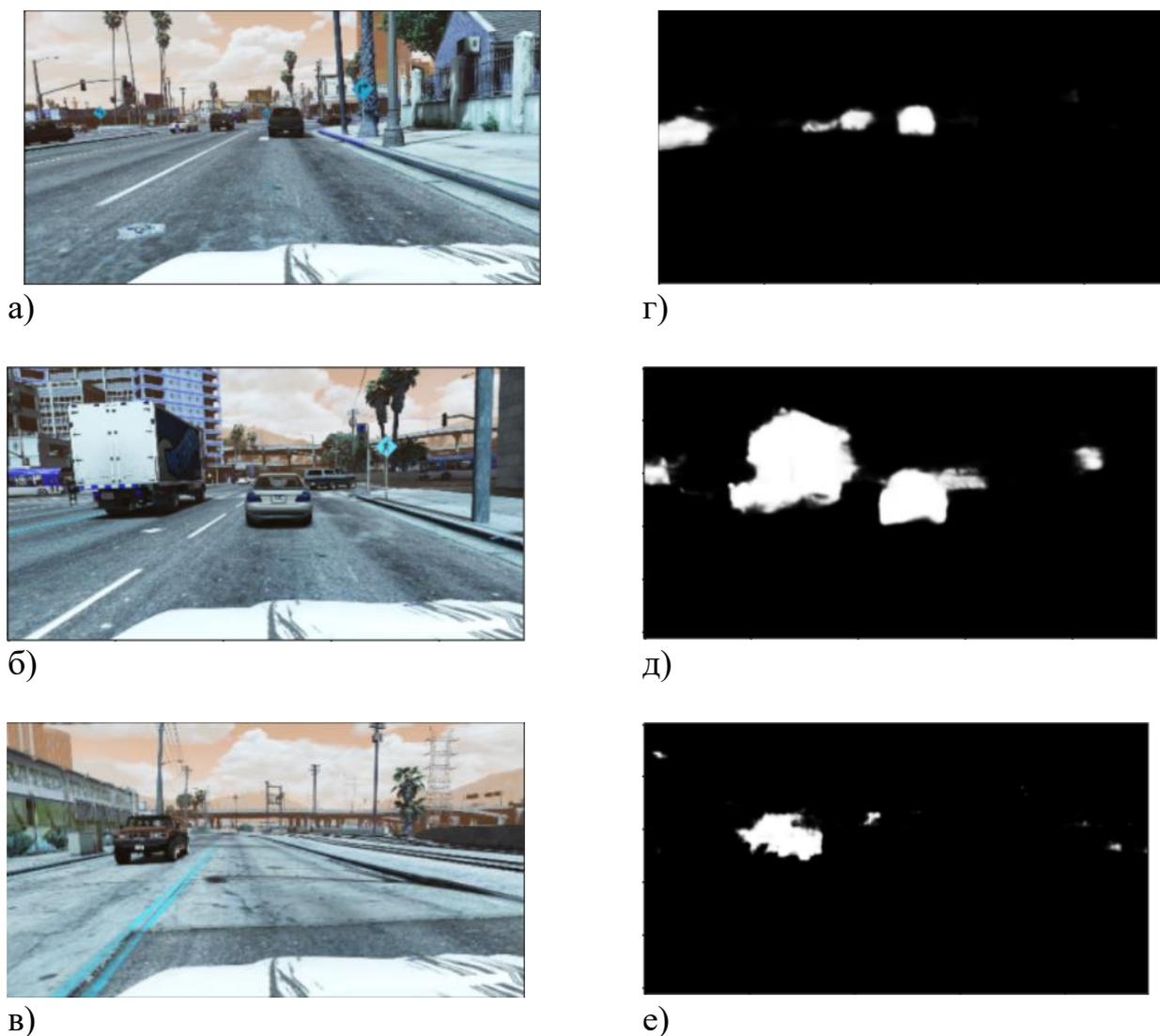
Рисунок 4. Построение архитектуры нейронной сети.

Для обучения модели сегментации автомобилей на изображении были использованы данные из набора gta5, представляющие большое количество изображений игрового мира, схожие с видами реальных улиц [7]. Обучающая и тестовая выборки были сформированы из 500 и 300 пар изображений и аннотаций соответственно. Аннотации были преобразованы таким образом, что для каждого пикселя решалась задача принадлежности к классу автомобилей, т.е. решалась задача бинарной классификации. Также перед обучением входные данные были нормализованы путем приведения значений цветов в диапазон от 0 до 1.

В результате обучения U-NET со скоростью обучения 0.003, функцией потерь бинарной кросс-энтропией и размером батча 16 на 48-ой эпохе обучения была получена модель для обнаружения автомобилей. Были получены следующие следующие метрики качества решения задачи сегментации на тестовой выборке:

- Точность (Precision): 66.5%.
- Полнота (Recall): 65.8%.

Данные значения метрик говорят о том, что 65.8% пикселей автомобилей обнаруживаются с точностью 66.5%, что говорит о том, что достаточно большие площади автомобилей могут быть обнаружены моделью. Результаты работы модели представлены на Рисунке 5.



*Рисунок 5. а, б, в - Исходное изображение;  
г, д, е - Результаты работы нейронной сети.*

Выводы проекта:

В ходе работы были достигнуты следующие результаты:

1. Были изучены алгоритмы машинного и глубокого обучения, применяемые для обработки изображений.
2. Реализована нейронная сеть U-NET для сегментации изображений и получена модель для сегментации машин, показывающая значение точности (precision) и полноты (recall) равное 66.5 и 65.8 процентов соответственно.
3. В дальнейшем модель может быть улучшена путем увеличения размера обучающей выборки, а также может быть создана и обучена модель для обнаружения разметки.

## СПИСОК ЛИТЕРАТУРЫ

1. Autonomous Off-Road Vehicle Control Using End-to-End Learning. Final technical report, 2004. - 55 с.
2. DeepLabV3 [Электронный ресурс] – режим доступа: <https://paperswithcode.com/method/deeplabv3> (дата обращения: 15.01.2023)
3. Introducing autonomous vehicles in logistics: a review from a broad perspective [Электронный ресурс] - режим доступа: <https://lirias.kuleuven.be/1719958?limo=0> (дата обращения: 29.11.2022)
4. Lane Detection [электронный ресурс] – режим доступа: <https://paperswithcode.com/task/lane-detection> (Дата обращения: 10.12.2022)
5. Precision and Recall [Электронный ресурс] – режим доступа: [https://scikit-learn.org/stable/auto\\_examples/model\\_selection/plot\\_precision\\_recall.html](https://scikit-learn.org/stable/auto_examples/model_selection/plot_precision_recall.html) (Дата обращения: 14.01.2023)
6. Olaf Ronneberger, Philipp Fischer, Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation // Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. Lecture Notes in Computer Science – 2015 – С. 9351.
7. Playing for Data: Ground Truth from Computer Games [Электронный ресурс] – режим доступа: <https://bitbucket.org/visinf/projects-2016-playing-for-data/src> (Дата обращения: 27.01.2023)
8. Сегментация изображения [Электронный ресурс] – режим доступа: [https://neerc.ifmo.ru/wiki/index.php?title=%D0%A1%D0%B5%D0%B3%D0%BC%D0%B5%D0%BD%D1%82%D0%B0%D1%86%D0%B8%D1%8F\\_%D0%B8%D0%B7%D0%BE%D0%B1%D1%80%D0%B0%D0%B6%D0%B5%D0%BD%D0%B8%D0%B9](https://neerc.ifmo.ru/wiki/index.php?title=%D0%A1%D0%B5%D0%B3%D0%BC%D0%B5%D0%BD%D1%82%D0%B0%D1%86%D0%B8%D1%8F_%D0%B8%D0%B7%D0%BE%D0%B1%D1%80%D0%B0%D0%B6%D0%B5%D0%BD%D0%B8%D0%B9) (Дата обращения: 20.12.2022)
9. Спицын В.Г., Цой Ю. Р. / Представление данных в информационных системах. Учебное пособие. Томск (2007) - 160 с.

## РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ РОБОТА-МАНИПУЛЯТОРА

*Шаманина Виктория Максимовна, Никитин Павел Николаевич*  
*Муниципальное бюджетное общеобразовательное учреждение*  
*лицей при ТПУ, 10 класс*  
*г. Томск*

Руководитель: Шаманин Олег Максимович, ассистент ТПУ

Цель проекта: разработка системы управления робота-манипулятора для загрузки деталей в станок с ЧПУ.

Актуальность. В настоящее время большое количество людей занимается низко квалифицированным ручным трудом и самостоятельно выполняют процессы производства. Одним из таких процессов является паллетирование, которое подразумевает собой перетаскивание и размещение партий груза в

зоне погрузки и на производственной линии. Автоматизация процессов работы является важным направлением развития производства и обладает преимуществами перед ручным трудом[1].

Процесс размещения деталей в станок с ЧПУ монотонен, поэтому шанс допущения ошибки человеком повышен, так как при длительной однообразной работе люди теряют усидчивость и уделяют меньше внимания своим действиям, в то время как машины могут работать постоянно с небольшой вероятностью того, что может произойти ошибка в выполнении задач.

Обслуживание робота обходится дешевле, чем оплата человеческого труда. При долгосрочной работе стоимость дорогого робота может окупиться за счет сокращения расхода средств на заработную плату сотрудников.

С внедрением роботов увеличивается производительность и эффективность труда[1].

Целевая аудитория. Система предназначена для управления роботами манипуляторами на предприятиях, которые нуждаются в автоматизации своего производства или логистических процессов. Такие роботы манипуляторы могут выполнять задачи по перемещению деталей.

Среди компаний, которые могут заинтересоваться роботами-манипуляторами, можно выделить производственные предприятия, складские комплексы, логистические компании и дистрибьюторов.

Задачи проекта:

- Провести обзор и сравнение аналогов
- Разработать структуру системы управления
- Разработать программное обеспечение для контроллера манипулятора
- Разработать программное обеспечение для встраиваемой панели человеко-машинного интерфейса
- Апробировать созданную систему

Обзор аналогов.

Робот-манипулятор компании Doosan используется для автоматизирования процесса загрузки деталей в токарный станок. Изменение параметров робота можно настроить с помощью доступного для человека, не имеющему опыта программирования, интерфейса. Перед началом работы манипулятора необходимо загрузить в роботизированную ячейку тележку с лотками, на которых находятся заготовки. Робот захватывает деталь, загружает в станок, а после извлекает готовое изделие и ставит его обратно. После того, как все содержимое тележки отработано, нужно извлечь и загрузить ее с новыми заготовками[2].

Другим аналогом является робот-манипулятор компании Haas. Его интерфейс также легок в использовании, человеку лишь потребуется задать положения манипулятора при выполнении загрузки. Перед началом работы также необходимо предоставить лоток с деталями. После загрузки первой детали робот возвращается к лотку, захватывает заготовку, далее свободной кистью извлекает готовую деталь и, меняя положение кистей, сразу же загружает новую, затем цикл повторяется[3].

В ходе работы мы создадим систему управления, которая будет предназначена для робота-манипулятора, выполняющего цикл погрузки деталей по аналогии с роботами компании Doosan и Haas.

Этапы выполнения проекта:

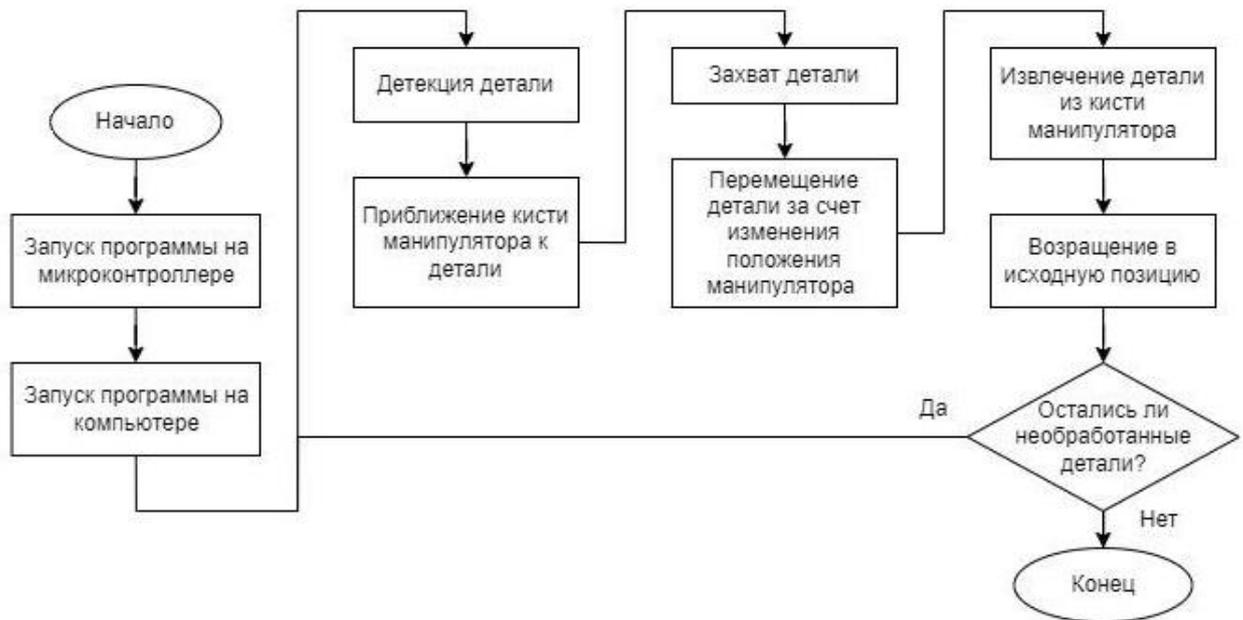
1. Изучение теоретического материала по робототехнике и системам управления;
2. Разработка функциональной схемы системы управления;
3. Выбор используемых компонентов, таких как робот, датчики, микроконтроллеры и программное обеспечение;
4. Разработка программного обеспечения для микроконтроллера манипулятора;
5. Разработка программного обеспечения для встраиваемой панели человеко-машинного интерфейса;
6. Разработка интерфейса связи микроконтроллера и панели управления
7. Настройка и интеграция всех компонентов системы управления в одно приложение;
8. Тестирование системы управления в на установке.

Используемые методы:

- Использование языка программирования Python для написания кода программы
- Применение библиотеки OpenCV для реализации технологии компьютерного зрения
- Применение библиотеки Tkinter для создания графического интерфейса
- Используемое оборудование:
- Механическая рука-манипулятор, необходимая для практического применения системы управления;
- Видеокамера, для реализации технологии компьютерного зрения;
- Компьютер, используемый для запуска управляющей программы;
- Управляющая плата Arduino для сопряжения манипулятора и программы.

Разработка системы.

Перед началом разработки необходимо составить функциональную схему для выявления алгоритма, который будет лежать в основе будущего программного обеспечения.

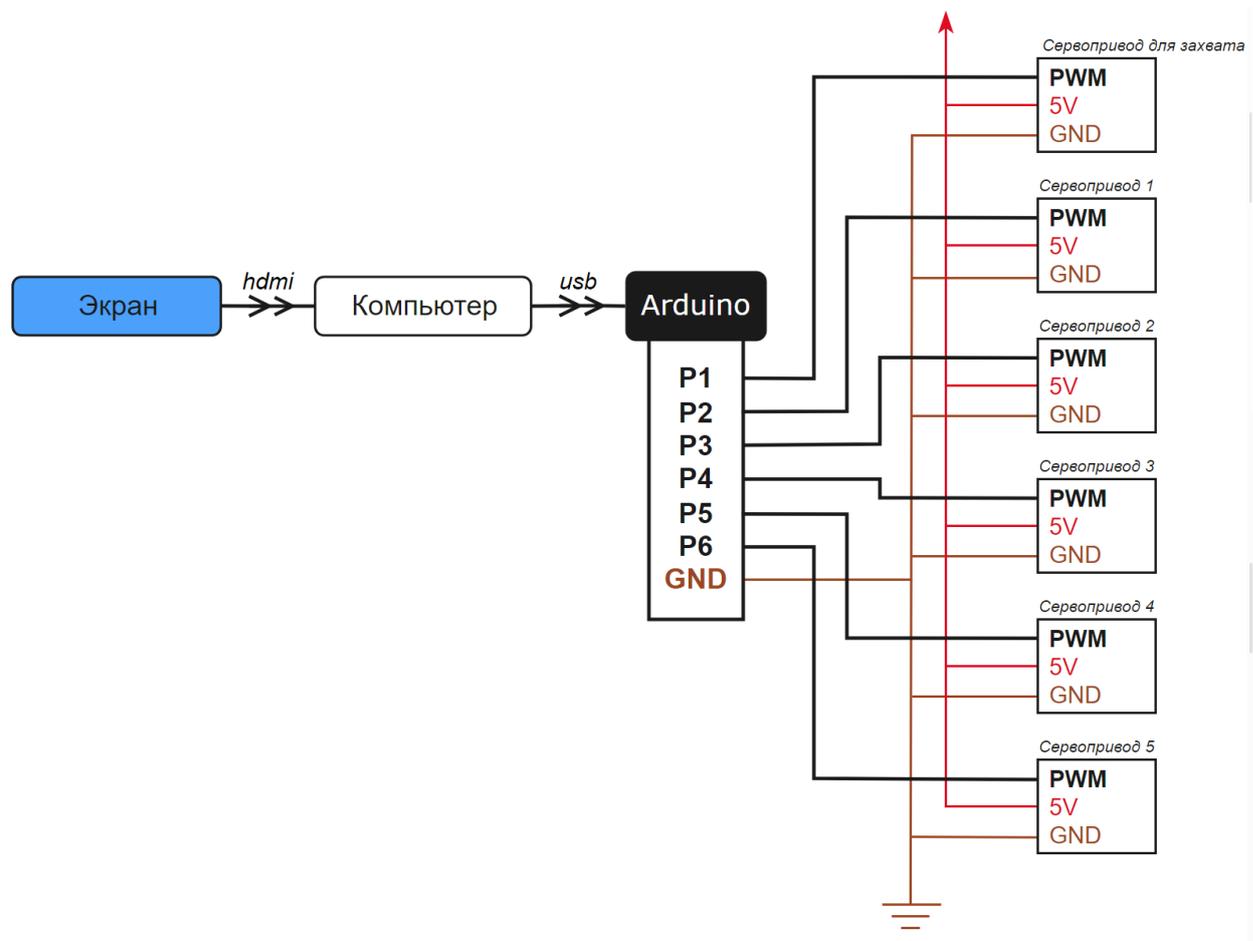


*Рисунок 1 Функциональная схема*

На схеме представлен будущий принцип работы манипулятора. После запуска программы на микроконтроллере и на компьютере выполняется цикл загрузки детали в станок. Цикл выполняется до тех пор, пока не останется необработанных заготовок.

Для функционирования системы управления, необходимо оснащение манипулятора технологией компьютерного зрения, а также разработка программного обеспечения, позволяющая роботу выполнять действия, заданные алгоритмом. Для настройки параметров нужно разработать программное обеспечение для встраиваемой панели человеко-машинного интерфейса.

Далее необходимо было составить структурную схему подключения для того, чтобы выявить из каких функциональных частей состоит манипулятор[4].



*Рисунок 2. Схема подключения*

Для сопряжения программы и манипулятора мы используем управляющую плату Arduino. К ней подключены 6 сервоприводов, которые нужны для осуществления изменений положений механизмов[5]. Сервоприводы, получая входной сигнал, будут поворачивать выходной вал в заданное положение. Робот имеет 5 осей вращения, для которых нужно задавать определенный угол[6].

Чтобы создать программное обеспечение потребовалось изучить необходимые функции языков программирования Arduino C и Python[7].

После объединения разработанных компонентов системы была произведена их программная настройка и апробирование.

**Заключение.**

Созданная в ходе работы системы была не идеальна и поэтому после разработки необходимого программного обеспечения и создания самой системы управления робота-манипулятора продолжилась работа по возможной оптимизации программного кода и улучшению интерфейса. Для лучшего быстрого действия программы возможно использование профайлера cProfile. С помощью Miro и Figma можно улучшить структуру интерфейса и его внешний вид.

**Выводы**

В ходе проделанной работы нам удалось разработать структуру системы управления, создать программное обеспечение для контроллера манипулятора, разработать программное обеспечение для встраиваемой панели человеко-машинного интерфейса, а также апробировать созданную систему и сформировать план по её улучшению.

Мы считаем, что создание подобных систем управления роботами-манипуляторами позволит заменить человеческий труд, свести количество возможных ошибок на производстве практически к минимуму, обеспечить более высокое качество выпускаемой продукции.

## СПИСОК ЛИТЕРАТУРЫ

1. Хасцаев Б. Д. и др. / Национальная ассоциация ученых 61 (2020) 17-21
2. Типовое решение по загрузке станков роботом-манипулятором
3. Doosan A0912S [Электронный ресурс] — режим доступа
4. <https://3dtool.ru/stati/robotizirovannaya-zagruzka-stankov-kollaborativny-manipulyatorom-a0912s/#4> (дата обращения: 11.03.2023)
5. Автоматический загрузчик деталей для токарного станка. [Электронный ресурс] — режим доступа <https://www.haascnc.com/ru/machines/lathes/auto-parts-loader.html> (дата обращения: 14.03.2023)
6. Структурная и функциональная электрические схемы. [Электронный ресурс] — режим доступа [https://studref.com/438950/tehnika/strukturnaya\\_funktsionalnaya\\_elektricheskie\\_shemy](https://studref.com/438950/tehnika/strukturnaya_funktsionalnaya_elektricheskie_shemy) (дата обращения: 02.03.2023)
7. Servo Motor Basics with Arduino. [Электронный ресурс] — режим доступа <https://docs.arduino.cc/learn/electronics/servo-motors> (дата обращения: 02.03.2023)
8. Basic servo control [Электронный ресурс] — режим доступа <https://docs.arduino.cc/tutorials/generic/basic-servo-control> (дата обращения: 02.03.2023)
9. tkinter — Python interface to Tcl/Tk [Электронный ресурс] — режим доступа <https://docs.python.org/3/library/tkinter.html> (дата обращения: 02.03.2023)

## РЕШЕНИЕ УРАВНЕНИЙ

*Шутов Артём*

*Государственное Бюджетное Нетиповое Общеобразовательное  
учреждение Лицей №84 им. В.А. Власова  
г. Новокузнецк*

Руководитель: Пылаева Анастасия Юрьевна, учитель информатики

Целью моего проекта стало ознакомление с языком java script и адаптации сайта под разные разрешения экранов

Программирование можно увидеть повсеместно и сейчас всё завязано на интернете. Почти каждый посещает различные сайты. Знание языков программирования может открыть широкий выбор профессии в будущем. Java Script является языком программирования, который позволяет взаимодействовать пользователю с сайтом.

Задачами было поставлено изучить JavaScript, научиться адаптировать сайты с помощью CSS и научиться работать в программе Figma.

Если рассмотреть аналоги моего проекта, то на мой взгляд их нет. Существует приложение на мобильные устройства «Photomath», однако его нужно устанавливать. Мой сайт проигрывает данному приложению тем, что на данный момент он умеет решать только квадратные уравнения, системы с тремя переменными (матричным способом) и находить уравнение параболы по трём точкам. В дальнейшем иной потенциал сайта будет увеличиваться.

Данный сайт рассчитан для тех, кто только начинает познавать решение уравнений. С помощью него можно сверить ответы или даже решения. Целевая аудитория рассчитана на учеников 6-11 классов.

JavaScript язык, который позволяет выполнять сложные вещи на web страницы – анимации, счёт, работа с базой данных. Он позволяет создать динамически обновляемый контент. Данный язык позволяет быстро интегрировать в код страницы любые теги, определять внешний вид элементов, а также их позиционирование.

Текстовым редактором, который я использовал является Visual Studio Code. Он быстрый, легкий и к тому же довольно мощный. Тем более его хвалят многие программисты. Однако от JetBrains есть альтернатива WebStorm, но он является платным и возобновлять подписку на территории России больше нельзя; поэтому он сразу отпал.

Для создания дизайна я использовал программу Figma. В ней есть возможность обработки как векторной, так и растровой графики; есть функция совместной работы; можно создавать всё что угодно – от логотипов до дизайнов приложений. Всего было сделано 3 макета под различные устройства. За основу брал Iphone 14, Ipad Pro 11”, MacBook Pro 16”.

После создания дизайна нужно создать директорию и нужные файлы. Главный файл я назвал index.html. По мимо этого создал три паки – 'styles', 'js files', 'pages'. Сразу создал все нужные страницы. При написании, когда следует проверить, чтобы файлы html имели кодировку UTF-8 для верного отображения.

Так как раньше я изучал Python, JS показался мне не таким сложным, но было тяжело понять его логику работы.

Следующим этапом было адаптивное устройство сайта под разные устройства. Пришлось набраться терпением и временем. Одним из главных инструментов является нажатие на сайте клавиши F12 и выбор разрешения экрана, под которое адаптируем. Адаптация осуществлялась с помощью атрибута @media

screen. При адаптации сайта лучше использовать расстояния в процентном соотношении, а не в пикселях, т.к. угадать с какого устройства будет смотреть пользователь и в какой ориентации невозможно.

В результате проделанной работы мы познакомились с языком программирования Java Script. С его помощью добавили вычислительные способности нашему сайту. Однако, он довольно непрост в изучении для тех, кто только начал изучать языки программирования. Во время работы проблемы возникали, но довольно быстро решались. Например, я не знал, как адаптировать сайт под планшеты, компьютеры и телефоны одновременно. Решилась проблема добавление двух «разных» страниц для каждого устройства.

Стоит отметить, что сайт идеально работает на всех устройствах, даже на телевизоре!

В целом, работа получилась интересна и поучительна. Во время неё я изучил новый язык программирования и научился делать адаптивные сайты. В дальнейшем, полученные знания я смогу использовать для создания более сложных проектов.

Ссылка на сайт: Решение квадратных уравнений (calculator-two-orgin.vercel.app).

## **ПРОТОТИП АГЕНТА ДЛЯ ИГРЫ 2048**

Мугтабаров Артур Русланович, Энгельке Виктория Алексеевна

*МАОУ «Молчановская СОШ 1»*

*10 класс с.Молчаново*

Руководитель: Осинцев Артем Викторович,

к.т.н., ассистент каф. "Телевидения и управления", Томский государственный университет систем управления и радиоэлектроники (ТУСУР)

### **Аннотация**

В работе представлены результаты проектирования программной модели агента для игры "2048", которая относится к математическим головоломкам. Разработан алгоритм и программный прототип управления игровым процессом. Выбран стек технологий и сформирован сценарий управления прототипом.

Ключевые слова: Искусственный интеллект, нейронная сеть, программирование, модель агента, машинное обучение.

### **Введение**

В текущее время активно развивается относительно новая область информатики «Машинное обучение» и искусственные нейронные сети. Они часто применяются в медицине (аппараты УЗИ, рентген), статистике (сбор и анализ данных), промышленности, образовании, игровой индустрии (создание игр). Также нейронные сети используются в разработке интеллектуальных систем видеонаблюдения, а также в различных приложениях, например, голосовые помощники и ассистенты [1]. С помощью искусственного интеллекта камеры способны распознавать людей, считывать номера машин, обрабатывать

поисковые запросы пользователей и предлагать товары или услуги, интересные именно данному пользователю. Искусственный интеллект упрощает жизнь людей и улучшает ее качество, что лишний раз подтверждает актуальность развития данной области.

В частности, выделяют ряд методов и алгоритмов в области разработки компьютерных игр. Так во многих играх, для улучшения игрового процесса используют сложные алгоритмы поведения NPC (Non-Player Character), неигровой персонаж в компьютерных и в настольных ролевых играх, которым управляет не игрок, а компьютер, что не может не привлекать внимание со стороны любителей компьютерных игр. Применение методов машинного обучения в тестировании и разработке игр имеет большую актуальность в наше время. Вот несколько причин, почему это так:

- Улучшение качества тестирования: МО может помочь автоматизировать процесс тестирования и обнаруживать более сложные ошибки, которые могут остаться незамеченными при ручном тестировании. Это увеличит качество тестирования и уменьшит количество ошибок в игре.
- Персонализированный геймплей: МО может помочь в создании персонализированного геймплея для каждого игрока. Алгоритмы МО могут анализировать данные об игроке, такие как его игровой стиль, предпочтения, уровень мастерства и т.д. Используя эту информацию, игра может настраиваться под каждого игрока, делая его более увлекательным и интересным.
- Улучшение искусственного интеллекта: МО может помочь улучшить искусственный интеллект в играх. Используя алгоритмы МО, разработчики могут обучать ИИ, как действовать в различных ситуациях и принимать более интеллектуальные решения. Это может улучшить игровой процесс и сделать игру более интересной для игроков.

На текущее время разработкой методов и алгоритмов искусственного интеллекта занимается множество научных коллективов по всему миру. Так, например, в их число входят как коммерческие компании OpenAI [2], Google [3], Microsoft [4].

Цель работы: разработка программной модели агента для достижения наилучшего результата в игре 2048.

Для решения поставленной цели были сформированы следующие задачи:

1. Провести анализ стратегий, обеспечивающих высокий результат в игре 2048.
2. Разработка прототипа программной модели агента, способной управлять игровым процессом.
3. Запуск и тестирование прототипа агента для игры 2048.

В процессе разработки прототипа был выполнен анализ открытых репозиторий игры 2048, опубликованных на платформе GitHub и выбран подходящий вариант реализации игры [5]. Выбранный проект обладает консольным

пользовательским интерфейсом, что упрощает реализацию игры, снижает требования и является гибким решением для реализации различных модификаций. В код игры был внесен ряд изменений. Структура проекта представлена на рисунке 1.



Рисунок 1. Структура прототипа проекта

В частности для эмуляции нажатия клавиш на клавиатуре. Для реализации этой задачи использовался модуль "keyboard" [6]. Что позволило "агенту" управлять игровым процессом без участия человека. Также мы автоматизировали подсчет среднего результата. Прототип программной реализации агента выполнял хаотичные (псевдослучайные) команд (вверх, вниз, влево вправо) за счет эмуляции нажатий клавиш клавиатуры. Что в конечном результате отражалось в перемещении блоков расположенных на игровом поле. При тестировании данного алгоритма выяснилось, что средний счет в игре – 680 очков, максимальный номинал плитки – 128 очков. На следующем этапе работы будет собрана статистика результатов 500 игр агента при выполнении хаотичного перемещения игровых блоков для оценки худшего результата не учитывающего номинал блоков и их расположение на игровом поле. Планируется разработка прототипа алгоритма для формирования стратегии управления игрой для анализа и выявления более эффективной стратегии.

Выводы и дальнейшие перспективы проекта.

Изучив принципы достижения высоких результатов в игре 2048, у нас появилось несколько идей создания алгоритмов. Разработан прототип программы агента, который будем взят за основу в последующих этапах выполнения работ по проекту.

## СПИСОК ЛИТЕРАТУРЫ

1. Волчек А. Технологии искусственного интеллекта: что умеют, где используются [Электронный ресурс] URL: <https://gb.ru/blog/tekhnologii-iskusstvennogo-intellekta> (дата обращения: 10.02.2023).
2. Официальный сайт OpenAI [Электронный ресурс] URL: <https://openai.com> (дата обращения: 10.02.2023).

3. Официальный сайт Google [Электронный ресурс] URL: <https://about.google> (дата обращения: 10.02.2023).
4. Официальная страница сайт Microsoft по искусственному интеллекту <https://www.microsoft.com/en-us/ai> (дата обращения: 10.02.2023).
5. Репозиторий проекта игры 2048 [Электронный ресурс] URL: <https://github.com/kubenet/py2048CLI> (дата обращения: 10.02.2023).
6. Страница модуля keyboard [Электронный ресурс] URL: <https://pypi.org/project/keyboard/> (дата обращения: 10.02.2023).