

**ООО «Алгоритмика-Пермь»**

УТВЕРЖДАЮ

Директор ООО «Алгоритмика-Пермь».



*А.В. Скокова*  
Скокова А.В.

Дополнительная общеобразовательная программа –

Дополнительная общеразвивающая программа

***«Курс программирования Python Про»***

***2 год обучения***

Направленность: техническая

Уровень программы: ознакомительный

Возраст учащихся: 14–17 лет

Срок реализации: 1 учебный год (64 академических часа)

Авторы-составители:

Скокова А.В.

г. Пермь, 2024

## Пояснительная записка

### 1.1. Направленность и уровень Программы

Информационные технологии — одна из наиболее быстро развивающихся сфер экономики. Ежегодно появляется множество новых вакансий в области разработки десктопных и мобильных приложений, создания сайтов, исследований в сфере искусственного интеллекта и анализа данных, и многих других. Статистика показывает, что востребованность специалистов в перечисленных сферах будет расти в последующие годы.

В таких условиях школьники заинтересованы в языке, который востребован в профессиональной среде и подходит для разработки современных программных продуктов. Таким языком является Python, интерес к которому растёт не только в России, но и за рубежом. Учителя, использующие Python в работе, отмечают такие достоинства языка, как сравнительная краткость программ и отсутствие необходимости использовать команды, не относящиеся к реализации алгоритмов. Таким образом, данный язык подходит и для изучения основ программирования, и для проектных работ, и для дальнейшей предпрофессиональной подготовки.

Еще одной важной задачей в подготовке IT-специалистов является развитие таких soft skills, как умение работать в команде, планирование, креативность, критическое мышление, навыки проведения презентаций и публичных выступлений.

Курс программирования Python Про позволяет учащимся познакомиться с профессиональным языком программирования Python и получить опыт решения задач, относящихся к разным рынкам и сферам IT, начать формировать портфолио разработчика и развивать перечисленные выше soft skills.

*Направленность* дополнительной образовательной программы — техническая. Она заключается в развитии технического и творческого мышления у учащихся среднего и старшего школьного возраста через разработку алгоритмов в цифровой среде, взаимодействие с приложениями, сервисами и инструментами с использованием разных типов платформ и интерфейсов. А также в развитии у учащихся умения составлять план деятельности, стремления к познавательным активностям, исследовательских, прикладных способностей, формированию навыков сотрудничества и коллективной работы в цифровой среде.

Развитие навыков программирования является уникальным способом формирования интереса учащихся к техническим направлениям деятельности, а также совершенствования логического, алгоритмического, критического мышления и навыков проектной деятельности.

Программа составлена с учетом задач национального проекта «Цифровая экономика» (федерального проекта «Кадры для цифровой экономики») и национального проекта «Образование» (федерального проекта «Молодые профессионалы»): создание условий для формирования у учащихся основ цифровых компетенций необходимых для дальнейшего участия в разработке и поддержке инфраструктур передачи, обработки и хранения данных, а также цифровых продуктов для граждан, бизнеса и власти; разработке, внедрении цифровых технологий и платформенных решений.

## **1.2. Актуальность**

В рамках курса учащиеся освоят основные инструменты программирования на языке Python. За время курса ребята создадут несколько собственных проектов и попробуют себя в роли наиболее востребованных IT-профессионалов — от разработчиков игр до специалистов по машинному обучению.

Особенность курса проявляется в том, что он направлен на предпрофессиональную подготовку и профориентацию в IT: каждый модуль представляет из себя одну из сфер деятельности, в которой может быть занят будущий программист. Результатом каждого такого модуля базового (теоретического) блока является программа, самостоятельно написанная учеником, в которой используются все инструменты, изученные в модуле. В проектном блоке в каждом модуле учащиеся знакомятся с функционалом одной из профессиональных библиотек, предназначенных для решения определенного круга задач. В программу обучения включены библиотека для разработки 3D игр, для работы с веб-страницами и их связи с базами данных, а также библиотеки для создания моделей машинного обучения и разработки мобильных приложений. Использование компьютерных технологий в работе с детьми среднего и старшего школьного возраста является стремительно развивающейся методикой в образовании во всем мире. С ее помощью можно более эффективно решать образовательные задачи, которые будут способствовать качественному улучшению обучения ребенка в школе.

### **Актуальность программы заключается в:**

- формировании основных навыков работы с профессиональными библиотеками на языке Python;
- формировании навыков работы с базами данных с помощью библиотеки Panda3D;
- формировании и развитии навыков работы с информацией в условиях модернизации образования;
- формирование навыков в области анализа данных и машинного обучения при помощи библиотек Pandas и Sklearn;
- формировании навыков работы с графикой и разработки пользовательских интерфейсов;
- развитие навыков планирования и использования итерационного подхода в рамках работы над проектами;
- формировании навыков предоставления и получения обратной связи;
- формировании навыков управления проектами;
- развитию цифровой и функциональной грамотности;
- развитию алгоритмического и креативного мышления, творчества через создание собственных проектов с использованием цифровых ресурсов;
- расширении кругозора школьника.

**Новизна программы** заключается в технической направленности обучения, которое проявляется в знакомстве с современными направлениями программирования, что способствует развитию информационной культуры и самоопределению учащихся относительно построения своей будущей карьеры. В отличие от других курсов программирования Python Pro предоставляет учащимся возможность с одной стороны освоить все профессиональные библиотеки используемые в Python, используя их непосредственно в разработке приложений, сайтов, при анализе данных и т.д.; с другой стороны — получить представление о работе программиста, изучая разнообразные современные технологии и библиотеки для работы с ними; с третьей стороны — выбирать наиболее близкие ученикам области для реализации полученных знаний. Проекты, выполняемые на уроках, охватывают разные области интересов, в том числе школьные предметы. Также ученики развивают умение самопрезентации и презентации своего проекта или идеи как в группе, так и перед учителями и родителями.

Авторское воплощение замысла курса заключается в том, что ученики используют инструменты профессиональной разработки для реализации собственных проектов и формирования портфолио разработчика. Для этого сначала ученик получает базовые знания о синтаксисе языка и алгоритмах, затем учится проектной деятельности, после чего реализует проекты в области своих увлечений. При этом базовая часть также подразумевает возможность выразить свои интересы. В базовой части проекты выполняются преимущественно индивидуально, чтобы учащиеся сначала научились работать над ними, затем уже — в группе.

Использование компьютерных технологий в работе с детьми среднего школьного возраста является стремительно развивающейся методикой в образовании во всем мире. С ее помощью можно более эффективно решать образовательные задачи, которые будут способствовать качественному улучшению обучения ребенка в школе.

### **1.3. Цель и задачи Программы**

Цель курса — способствовать формированию у учащихся умения использовать язык программирования Python в качестве инструмента для реализации идей в области своих личных или профессиональных интересов.

Достижение поставленной цели складывается из выполнения следующих **задач**:

*Обучающие:*

1. Изучить основы разработки сайтов и мобильных приложений.
2. Изучить основы анализа данных и применения алгоритмов машинного обучения.
3. Изучить основы разработки 3D-игр.
4. Сформировать навык использования итерационного подхода при решении различных задач.
5. Сформировать навык тестирования и оптимизации созданных проектов.

*Развивающие:*

1. Формировать и развивать логическое, алгоритмическое и критическое мышление.
2. Развивать навык публичного выступления и презентации.

3. Развитие креативных способностей и умений учиться.
4. Развивать навык планирования деятельности и основ тайм-менеджмента.
5. Развивать функциональную грамотность.
6. Развивать мотивацию успеха, готовность к действиям в новых условиях и нестандартных ситуациях.
7. Развивать коммуникативные способности и умение работать в команде.
8. Расширять кругозор, развивать память, внимание, творческое воображение, абстрактно-логические и алгоритмические виды мышления и типы памяти, основные мыслительные операции и свойства внимания.
9. Совершенствовать диалогическую речь учащихся: уметь слушать собеседника, понимать вопросы, смысл используемых терминов, уметь задавать вопросы, отвечать на них.

*Воспитательные:*

1. Воспитывать у учащихся потребность в сотрудничестве, взаимодействии со сверстниками, умение подчинять свои интересы определенным правилам, развивать эмоциональный интеллект.
2. Формировать умение давать качественную обратную связь и реагировать на нее.
3. Формировать информационную культуру: умение целенаправленно использовать современные технические средства и процессы для получения, обработки и передачи информации.
4. Повышение компетентности учащихся в области планирования карьеры.

Данные задачи не только способствуют формированию предметных и метапредметных компетенций, но и формируют мировоззрение школьника, раскрывают значение soft skills, учат грамотному использованию компьютера и готовят к жизни в цифровом обществе.

#### **1.4. Формы и режим занятий**

Занятия проходят 1 раз в неделю и длятся 2 академических часа. Дети занимаются в классе с учителем, самостоятельно работая за компьютером (непрерывно — не более 30 минут), занятия разделены перерывом. Основные формы работы — индивидуальная, фронтальная, в т.ч. через организацию групповых дискуссий.

*Специфика форм обучения:*

- Обучение от общего к частному.
- Поощрение вопросов и свободных высказываний по теме.
- Уважение и внимание к каждому ученику.
- Создание мотивационной среды обучения.
- Создание условий для дискуссий и развития мышления учеников при достижении учебных целей вместо простого одностороннего объяснения темы преподавателем.

*Учебное занятие состоит из следующих блоков:*

1. Повторение и актуализация знаний.
2. Изучение нового материала.
3. Практика на платформе.
4. Рефлексия: подведение промежуточных итогов.
5. Разминка.
6. Изучение нового материала.
7. Практика на платформе и в среде программирования с расширением учебной платформы Алгоритмики.
8. Рефлексия: подведение итогов занятия.

*Занятие проектного типа состоит из следующих блоков:*

1. Повторение.
2. Постановка проблемы.
3. Планирование проектов.
4. Прототипирование.
5. Подведение промежуточных итогов.
6. Разминка.
7. Тестирование и доработка решения.
8. Подготовка презентации и презентация.
9. Рефлексия.

*Ученики могут давать обратную связь по проектам друг друга, для чего используются этапы повторения, подведения итогов и рефлексии, также предусмотрены занятия с возможностью презентовать собственную работу.*

## **1.5. Срок реализации Программы**

**Содержание курса** программирования Python Pro для обучающихся 14–17 лет рассчитано на обучение в течение 8 месяцев (32 занятия — 64 академических часа).

## 1.6. Планируемые результаты

Достижение цели и задач образовательной программы предполагает получение следующих результатов:

Планируемые результаты	Способ достижения	Критерий достижения образовательного результата
<b>Предметные навыки</b>		
Следование стандарту при оформлении кода.	Изучение теоретического материала совместно с учителем, выполнение заданий на платформе.	Ученик оформляет код в соответствии с общепринятыми стандартами и использует комментарии.
Знакомство с профессиональными инструментами языка программирования Python.	Изучение теоретического материала совместно с учителем, выполнение заданий на платформе с постепенным увеличением сложности.	Ученик самостоятельно пишет программы, используя язык программирования Python.
Развитие умений применять Python для решения задач практического и междисциплинарного характера.	Выполнение заданий на платформе с постепенным увеличением сложности.	Ученик самостоятельно пишет программы на языке Python, решающие задачи практического характера.
Использование базы данных для обеспечения клиент-серверного взаимодействия.	Изучение теоретического материала совместно с учителем, выполнение заданий на платформе с постепенным увеличением сложности.	Ученик самостоятельно настраивает на сайте работу с данными.

Умение готовить и использовать данные для создания моделей машинного обучения	Изучение теоретического материала совместно с учителем, выполнение заданий на платформе с постепенным увеличением сложности.	Ученик для выполнения проекта самостоятельно отбирает, обрабатывает и готовит данные, на основе которых затем создает алгоритм для машинного обучения.
Умение тестировать программу и находить в ней ошибки.	Выполнение заданий на платформе с постепенным увеличением сложности.	Ученик самостоятельно находит ошибки в своем и чужом коде.
Умение использовать чужой код при разработке: анализировать и изменять.	Выполнение заданий на платформе.	Ученик самостоятельно вносит изменения в готовый код и использует фрагмент готового кода в процессе реализации собственной программы.
Использование итерационного подхода в разработке программ.	Выполнение заданий на платформе.	Ученик пишет программы, разделяя задачу на отдельные части.

<b>Личностные навыки</b>		
Освоение социальной роли обучающегося и формирование личностного смысла учения.	Демонстрация связи между способностью выполнить интересную задачу и наличием/отсутствием соответствующих знаний.	За отведенное время ученик пытается не только выполнить базовые уровни, но и приступить к бонусным.

<p>Развитие навыков сотрудничества со взрослыми и сверстниками и умения находить выходы из спорных ситуаций.</p>	<p>Использование программирования как способа показать преимущества работы в команде.</p>	<p>Ученик не боится просить помощь и сам пытается помогать одноклассникам и учителю.</p>
<p>Овладение начальными навыками адаптации в динамично изменяющемся и развивающемся мире.</p>	<p>Обсуждение научно-популярных материалов, демонстрирующих необходимость не просто овладения предметными навыками, но и развития умения учиться.</p>	<p>При работе над проектом ученик сначала пытается найти информацию в Интернете или теоретической справке на платформе и только при неудаче задает вопрос учителю.</p>
<p>Умение аргументировать свою позицию.</p>	<p>Выступление во время защиты проектов, ответы на вопросы.</p>	<p>Ученик обосновывает подход к решению проблемы, используя цепочки аргументов. Помнит, как использовались логические связи между ними. Отвечает на критику аргументами, а не эмоциями.</p>
<p>Умение решать задачи, с которыми раньше не сталкивался.</p>	<p>Ситуации успеха при решении незнакомых задач: вместо того, чтобы дать пример, затем повторить, сначала даем задачу и через подсказки помогаем дойти до решения.</p>	<p>Ученик, сталкиваясь с новой задачей, ищет решение: задает вопросы, проясняющие структуру задачи; находит знакомые подзадачи и аналогии; разделяет составные части задачи. При возникновении проблемы предлагает способы ее решения или понимает, что текущих знаний для решения проблемы не хватает и предлагает способы эти знания получить.</p>

<p>Понимание того, что ошибки — это инструмент обучения.</p>	<p>Часть заданий подразумевают, что надо попробовать несколько методов, один из которых заработает, единственно верный вариант не дается заранее.</p>	<p>Ученик пробует свои идеи, не ожидая разрешения. При обнаружении ошибок старается исправлять их самостоятельно.</p>
--	---	---

<p align="center"><b>Метапредметные навыки</b></p>		
<p>Развитие и формирование навыка умения учиться.</p>	<p>Создание благоприятных условий для участия в диалоге, в коллективном обсуждении. Строится продуктивное взаимопонимание со сверстниками и взрослыми в процессе коллективной деятельности.</p>	<p>Ученик легко общается, не боится просить помощь или оказать ее другим. Ученик способен кооперироваться с другими, чтобы достичь цели.</p>
<p>Умение презентовать свою работу.</p>	<p>Презентация индивидуальных проектов. Учитель и другие ученики дают обратную связь. Учитель также дает советы, каким образом это лучше делать.</p>	<p>Ученик во время презентации своих проектов пользуется вниманием аудитории.</p>
<p>Развитие критического и алгоритмического мышления учеников.</p>	<p>Выполнение логических операций: сравнение, анализ, синтез, обобщение, установление аналогий на уроке. Анализ любой получаемой информации, составление плана действий для решения задачи.</p>	<p>Ученик строит логическую цепь рассуждений. Управляет своей деятельностью.</p>
<p>Развитие творческих способностей учеников.</p>	<p>Планирование и создание проекта по сценарию. Использование инструментов цифрового</p>	<p>Ученик придумывает, обсуждает, планирует и реализует свой проект.</p>

	сервиса для воплощения своего проекта.	
Умение управлять своим временем: устанавливать приоритеты, умеет отказываться от неважных задач.	Выполнение заданий, направленных на получения опыта ведения проектов.	Ученик умеет планировать свое время, определять приоритет задач, благодаря чему успевает сделать проект в срок и доволен своим результатом.
Умение совместно с другими распределять задания в группе.	Выполнение групповых проектов.	Ученик без помощи преподавателя договаривается с другими участниками команды, кто что делает.
Умеет давать и получать обратную связь.	Презентация проектов.	Ученик в ходе комментирования работы другого избегает оценочных суждений, выделяет как сильные, так и слабые стороны.
Умеет планировать проекты.	Выполнение проектов: в начале выполнения проектов зарисовывается его план, учитель показывает, как это лучше делать.	Ученик раскладывает проект на составные элементы, затем выстраивает их в последовательность шагов.

## 1. Содержание Программы

### 2.1. Учебный (тематический) план

№ п/п	Наименование раздела/темы	Количество часов			Форма аттестации (контроля)
		всего	теория	практика	
<b>Модуль 1. Повторение</b>		<b>4</b>			
1	Повторение. Алгоритмы и структуры данных	2	1	1	Задания на платформе
2.	Повторение. Функции и ООП	2	1	1	Задания на платформе
<b>Модуль 2. Мобильные приложения</b>		<b>8</b>			
3.	Знакомство с библиотекой Kivy	2	1	1	Задания в среде программирования с расширением учебной платформы Алгоритмики
4.	Разработка MVP мобильного приложения	2	1	1	
5.	Создание user-friendly интерфейса. Анимация.	2	1	1	

6.	Тестирование приложений. Презентация продукта потенциальным пользователям.	2	1	1	Защита проекта
<b>Модуль 3. Анализ данных</b>		<b>12</b>			
7.	Основы работы с Pandas. Фильтрация данных	2	1	1	Задания в среде программирования с расширением учебной платформы Алгоритмики
8.	Методы группировки данных	2	1	1	
9.	Очистка данных. Feature engineering	2	1	1	
10.	Работа над индивидуальным проектом	2	1	1	
11.	Визуализация данных	2	1	1	
12.	Защита проектов	2	1	1	Защита проекта
<b>Модуль 4. Машинное обучение</b>		<b>8</b>			
13.	Алгоритмы машинного обучения. Классификация методом KNN	2	1	1	Задания в среде программирования с расширением учебной платформы Алгоритмики
14.	Стратегия и тактика обучения модели	2	1	1	
15.	Работа над индивидуальным проектом	2	1	1	
16.	Презентация проектов. Подведение итогов соревнования	2	1	1	Защита проекта
<b>Модуль 5. 3D игры</b>		<b>10</b>			
17.	Знакомство с Panda3D	2	1	1	Задания в среде программирования с расширением учебной платформы Алгоритмики
18.	Камера и карта	2	1	1	
19.	Редактор карт	2	1	1	
20.	Игровой режим	2	1	1	
21.	Работа над индивидуальным проектом	2	1	1	
22.	Итоги. Презентация игры	2	1	1	Защита проекта
<b>Модуль 6. Веб-разработка</b>		<b>16</b>			
23.	Основы веб-технологий. Введение в HTML	2	1	1	Задания в среде программирования с расширением учебной платформы Алгоритмики
24.	Введение в Agile и Scrum. CSS	2	1	1	
25.	Работа с базами данных. Введение в SQL	2	1	1	

26.	Работа со связанными таблицами. Создание скрипта базы данных для сайта	2	1	1	
27.	Взаимодействие сайта с несколькими пользователями. Сессии	2	1	1	
28.	Создание форм	2	1	1	
29.	Использование шаблонов	2	1	1	
30.	Презентация проектов	2	1	1	Защита проекта
<b>Модуль 7. Моя карьера в IT</b>		<b>4</b>			
31.	Варианты развития карьеры в IT	2	1	1	Задания в среде программирования с расширением учебной платформы Алгоритмики
32.	Выпускной: «Моя карьера в IT»	2	1	1	Защита проекта

## 2.2. Содержание учебного (тематического) плана

### Модуль 1. Повторение

#### 1.1. Повторение. Алгоритмы и структуры данных

*Типы данных. Условный оператор, цикл. Структуры данных: списки, словари, кортежи. Функции и модули для реализации программ.*

#### 1.2. Повторение. Функции и ООП

*Функции и модули для реализации программ. Функции с переменным количеством аргументов.*

### Модуль 2. Мобильные приложения

#### 2.1. Знакомство с библиотекой Kivy

*Параметры виджетов в Kivy и их назначение. Способы работы с событиями в Kivy. Именованный аргумент. Распаковка словаря. Эмулирование работы приложения для различных устройств. Обработчик события. Переключение между экранами.*

*Создание приложения, состоящего из нескольких связанных между собой экранов.*

## 2.2. Разработка MVP мобильного приложения

*Понятие MVP. Распакованный список. Способы кодирования цвета в форматах RGBA и HEX. Назначение background\_color и clearcolor. Теги для стилизации встроенного в виджет текста*

## 2.3. Создание user-friendly интерфейса. Анимация

*Объект Clock в Kivy, его назначение и методы. Типы свойств NumericProperty, BooleanProperty и StringProperty. Визуализация в виде счётчика и бегунка. Анимация в Kivy. Принцип «Fail Fast».*

## 2.4. Тестирование приложений. Презентация продукта потенциальным пользователям.

*Понятие Тестирования, причины его проводить. Виды тестирования. Объект Ropir. Удаление виджетов. Метод previous(). Остановка работы приложения при помощи stopTouchApp. Роль MVP в тестировании гипотез и проверки жизнеспособности продукта.*

## Модуль 3. Анализ данных

### 3.1. Основы работы с Pandas. Фильтрация данных

*Питчинг, его роль и назначение в предпринимательской-среде. Основные характеристики структур данных Series и DataFrame. Назначение и синтаксис методов: info(), head(), tail(), describe(), min(), max(), mean(), median(). Синтаксис фильтрации данных по одному или нескольким условиям. Понятие фильтрации данных. Статистические показатели: среднее арифметическое, стандартное отклонение, медиана, квантили.*

### 3.2. Методы группировки данных

*Целевая аудитория и её роль в питчинге. Функции подсчёта количества уникальных вхождений value\_counts() и группировки: groupby(), pivot\_table(). Типы задач, которые могут быть решены с помощью группировки. Настройка параметров методов группировки в зависимости от задачи.*

*Получение значений конкретных показателей из DataFrame и Series, полученных с помощью группировки.*

### 3.3. Очистка данных. Feature engineering

*Достоверные источники данных и их влияние на качество анализа и презентации. Пустые значения и как они влияют на работу с DataFrame. Синтаксис функций и методов fillna(), isnull(), apply(). Очистка данных, ее влияние на работу с датасетом. Генерация новых признаков.*

### **3.4. Работа над индивидуальным проектом**

*Понятие гипотезы. Формулирование и проверка гипотезы. Улучшение качества анализа данных на этапе подготовки. Планирование индивидуального проекта с помощью mind-тар и чек-листа.*

### **3.5. Визуализация данных**

*Дашборд и его назначение в презентации. Диаграммы. Улучшение качества анализа данных с помощью графиков и диаграмм. Метод plot() библиотеки Pandas.*

### **3.6. Защита проектов**

*Построение графиков и диаграмм с помощью Seaborn и Plotly. Презентация анализа одного из проектов. Понятие питчинга и его роль в привлечении инвесторов. Структура питчинга.*

## **Модуль 4. Машинное обучение**

### **4.1. Алгоритмы машинного обучения. Классификация методом KNN**

*A/B тестирование и его роль в предпринимательской-среде. Машинное обучение. Классические задачи машинного обучения. Алгоритм KNN. Создание математической модели. Оценка точности модели.*

### **4.2. Стратегия и тактика обучения модели**

*Организация работы по созданию модели машинного обучения. Фиксирование гипотез. Подготовка данных для анализа. Создание и модели на основе алгоритма KNN, оценка ее точности. Тренировочный и тестовый набор данных для обучения модели. Закрепление понятия A/B тестирования.*

### **4.3. Работа над индивидуальным проектом**

*Знакомство с сайтом kaggle.com. Назначение файлов train.csv и test.csv, которые прилагаются к соревнованию. Преимущества участия в соревнованиях по машинному обучению. A/B тестирование.*

*Принятие участия в соревновании на сайте.*

#### **4.4. Презентация проектов. Подведение итогов соревнования**

*Продвинутые методы решения задачи классификации и виды задач машинного обучения. Повышение точности модели на этапе преобразования данных.*

### **Модуль 5. 3D игры**

#### **5.1. Знакомство с Panda3D**

*Понятие высококонкурентной среды и его влияние на игровую индустрию. Этапы получения объемного изображения. Элементы сцены в 3D графике. Создание приложений с помощью Panda3D. Создание и настройка основных элементов игры.*

#### **5.2. Камера и карта**

*Методы Panda3D для перемещения и поворота камеры. Принципы обработки событий и задач в Panda3D. События нажатия кнопок клавиатуры и мышки. Менеджер задач Panda3D. Ориентирование в 3D пространстве. Закрепление понятия высококонкурентная среда.*

*Создание 3d карты для игры.*

#### **5.3. Редактор карт**

*Определение столкновений трехмерных объектов. Библиотека pickle для чтения и записи объектов Python. Проектирование структуры приложения, связь между классами. Редактирование 3D карты.*

*Создание приложения из разных классов.*

#### **5.4. Игровой режим**

*Эффект гравитации, механика его создания. Перемещение по карте. Расчет положения объекта с учетом его скорости и ускорения.*

*Добавление игрового режима в проект, создание карт из блоков разных типов.*

#### **5.5. Работа над индивидуальным проектом.**

*Конкурентная среда. Принципы продвижения продукта в ней.*

*Работа над проектом.*

## **5.6. Итоги**

*Анимация персонажей, вывод текста на экран, skybox, работа со звуком. Принципы планирования и управления проектами. Trello.*

*Планирование дальнейшего развития проекта.*

## **Модуль 6. Веб-разработка**

### **6.1. Основы веб-технологий. Введение в HTML**

*Социальное предпринимательство и его назначение в бизнес-среде. Принципы организации Интернета и работы браузера. Значения понятий: клиент, сервер, IP-адрес, домен, протокол. Синтаксис HTML для создания веб-страниц. Теги абзаца, заголовков, списков, длинных цитат, выделения слов, ссылок, изображений. Особенности взаимодействия клиента и сервера для статического и динамического сайта. Структуру HTML-документа.*

*Создание веб-страниц с использованием синтаксиса HTML.*

### **6.2. Введение в Agile и Scrum. CSS**

*Синтаксис CSS. Подключение файлов стилей к веб-страницам. Каскадные таблицы стилей. Подключение .css и .html файлов к веб-серверу. Закрепление понятия социальное предпринимательство.*

*Оформление веб-страниц с помощью стилей.*

### **6.3. Работа с базами данных. Введение в SQL**

*Команды SQL для работы с однотабличной базой данных: CREATE, SELECT, INSERT, UPDATE, DELETE. Синтаксис методов модуля sqlite3 для подключения базы данных и выполнения запросов к ней. Соединение, курсор, их роль при работе с базой данных.*

*Совершение запросов на выборку, в том числе содержащих условия. Обработка результатов запросов с помощью Python.*

### **6.4. Работа со связанными таблицами. Создание скрипта базы данных для сайта.**

*Команды SQL для создания, заполнения и выполнения запросов к связанным таблицам: CREATE, PRIMARY KEY, FOREIGN KEY, DROP, INSERT, SELECT, FROM, WHERE, ORDER BY. Синтаксис метода cursor.executemany(). Связи «один к одному», «один ко многим», «многие ко многим».*

*Создание структуры базы данных со связанными таблицами, ее заполнение данными. Выполнение запросов к связанным таблицам.*

## **6.5. Взаимодействие сайта с несколькими пользователями. Сессии**

*Функция представления. Связь страниц сайта. Сессии. Словарь сессий для каждого клиента, секретный ключ.*

*Создание работающего веб-приложения с несколькими страницами, позволяющее просмотреть по очереди записи из БД.*

## **6.6. Создание форм**

*HTML-формы. Объект request, методы GET и POST. Получение и обработка данных из форм.*

*Создание работающего веб-приложения, которое позволяет пользователю выбрать викторину и выводит вопросы в соответствии с этим выбором.*

## **6.7. Использование шаблонов**

*Понятие шаблона. Путь к шаблонам и статическим файлам. Вызов функции обработки шаблона. Наполнение шаблона данными. Подключение CSS стилей.*

## **6.8. Презентация проектов**

*Локальная сеть. Подключение к веб-приложению через локальную сеть. Презентация проекта “Викторина”.*

## **Модуль 7. Моя карьера в IT**

### **7.1. Варианты развития карьеры в IT**

*Профессии в сфере IT. Онлайн-сообщества разработчиков и IT-предпринимателей. Онлайн-ресурсы для поиска работы и финансирования стартапов.*

### **7.2. Выпускной «Моя карьера в IT»**

*Формирование представления о вариантах развития собственной карьеры. Составление плана профессионального развития. Презентация «Моя карьера в IT: мечта → цель → план → действия».*

### **2.3 Формы аттестации и оценочные материалы**

Текущий контроль сформированности результатов освоения программы осуществляется с помощью нескольких инструментов на нескольких уровнях:

- **на каждом занятии:** опрос, выполнение заданий на платформе, реализация проектов, самоконтроль ученика либо взаимоконтроль учеников в парах или мини-группах, общее подведение итогов занятия;
- **в конце каждого модуля:** проведение презентации (по желанию) финальных проектов модуля и их оценка.

Для контроля сформированности результатов освоения программы на платформе используются задания на программирование (как написание программ с нуля, так и исправление ошибок в предложенной программе) и следующие механики тестирования:

- классический тест (выбор одного или нескольких правильных ответов),
- заполнение пропусков,
- классификация,
- сопоставление,
- сортировка,
- ответ в свободной форме, в т. ч. прикрепление файла.

Задания на платформе представлены как с автопроверкой (все задания на тему базовых понятий проверяются автоматически системой), так и с возможностью свободного решения; в проектных заданиях имеются чек-листы для преподавателей и учеников. Преподаватель имеет доступ ко всем кодам учеников (как записанных на онлайн-платформе, так и в системе Visual Studio Code), помогает находить ошибки и проводить ревью кода (выборочная оценка кода учеников использование оптимальных конструкций и структур данных, и на соблюдение стандартов его написания). В учебном плане предусмотрены конференции и круглые столы, на которых возможна взаимопроверка, собирается обратная связь учеников, может проводиться анкетирование родителей.

### **3. Организационно-педагогические условия реализации Программы**

#### **3.1. Организационно-педагогические условия реализации Программы содержат:**

##### **материально-технические условия реализации Программы:**

- учебный кабинет с местами для 12 учеников;
- компьютеры для учеников (1 ученик — 1 компьютер) и компьютер для учителя (требования к компьютерам: ОС Windows версии 7, 8 или 10, либо Mac OS версии не ниже 10.10, либо Linux — Ubuntu Desktop не ниже 14.04 или Debian 7; оперативная память объемом от 4 Гб, рекомендуемый объем от 8Гб; процессор Intel серии Core I3, 5, 7 не младше третьего поколения,

или серий Pentium Silver/Gold, либо AMD серии Ryzen или старшие модели серий A10 и A12);

- стационарный доступ в Интернет с компьютеров учеников и учителя (рекомендуется от 30 Мбит/с);
- веб-браузер на основе Chromium (рекомендуется Google Chrome), обновленный до последней версии;
- проектор, подключенный к компьютеру для учителя, выводящий изображение на экран;
- распечатанные логины и пароли учеников для доступа на платформу (учитель имеет доступ к общему списку логинов и паролей своей группы в электронном виде);
- установленная на компьютерах среда разработки Visual Studio Code с дополнением от Алгоритмики, которое позволяет организовывать учебный процесс работы над проектами (инструкция по установке дается учителю вместе с другими учебными материалами);
- канцелярия: бумага А4, ручки/карандаши;
- доска или флипчарт, мел или маркер.

#### **учебно-методическое и информационное обеспечение Программы:**

- поурочные методические рекомендации к занятиям;
- тематические презентации;
- задания на платформе с автопроверкой/со свободным решением;
- бонусные задания на платформе;
- задачи со свободным решением для реализации в среде программирования Visual Studio Code, в которую интегрирована платформа Алгоритмики.

**Календарный учебный график**

№ п/п	Месяц	Число	Время проведения занятия	Форма занятия	Количество часов	Тема занятия	Место проведения	Форма контроля
1				Интерактивное занятие	2	Повторение. Алгоритмы и структуры данных		Задания на платформе
2				Интерактивное занятие	2	Повторение. Функции и ООП		Задания на платформе
3				Интерактивное занятие	2	Знакомство с библиотекой Kivy		Задания в среде программирования с расширением учебной платформы Алгоритмики
4				Интерактивное занятие	2	Разработка MVP мобильного приложения		Задания в среде программирования с расширением учебной платформы Алгоритмики
5				Интерактивное занятие	2	Создание user-friendly интерфейса. Анимация.		Задания в среде программирования с расширением учебной платформы Алгоритмики
6				Интерактивное занятие	2	Тестирование приложений. Презентация продукта потенциальным пользователям.		Защита проекта
7				Интерактивное занятие	2	Основы работы с Pandas. Фильтрация данных		Задания в среде программирования с расширением учебной платформы Алгоритмики
8				Интерактивное занятие	2	Методы группировки данных		Задания в среде программирования с расширением учебной платформы Алгоритмики
9				Интерактивное занятие	2	Очистка данных. Feature engineering		Задания в среде программирования с расширением учебной платформы Алгоритмики

10				Интерактивное занятие	2	Работа над индивидуальным проектом		Задания в среде программирования с расширением учебной платформы Алгоритмики
11				Интерактивное занятие	2	Визуализация данных		Задания в среде программирования с расширением учебной платформы Алгоритмики
12				Интерактивное занятие	2	Защита проектов		Защита проекта
13				Интерактивное занятие	2	Алгоритмы машинного обучения. Классификация методом KNN		Задания в среде программирования с расширением учебной платформы Алгоритмики
14				Интерактивное занятие	2	Стратегия и тактика обучения модели		Задания в среде программирования с расширением учебной платформы Алгоритмики
15				Интерактивное занятие	2	Работа над индивидуальным проектом		Задания в среде программирования с расширением учебной платформы Алгоритмики
16				Интерактивное занятие	2	Презентация проектов. Подведение итогов соревнования		Защита проекта
17				Интерактивное занятие	2	Знакомство с Panda3D		Задания в среде программирования с расширением учебной платформы Алгоритмики
18				Интерактивное занятие	2	Камера и карта		Задания в среде программирования с расширением учебной платформы Алгоритмики
19				Интерактивное занятие	2	Редактор карт		Задания в среде программирования с расширением учебной платформы Алгоритмики

20				Интерактивное занятие	2	Игровой режим		Задания в среде программирования с расширением учебной платформы Алгоритмики
21				Интерактивное занятие	2	Работа над индивидуальным проектом		Задания в среде программирования с расширением учебной платформы Алгоритмики
22				Интерактивное занятие	2	Итоги. Презентация игры		Защита проекта
23				Интерактивное занятие	2	Основы веб-технологий. Введение в HTML		Задания в среде программирования с расширением учебной платформы Алгоритмики
24				Интерактивное занятие	2	Введение в Agile и Scrum. CSS		Защита проекта
25				Интерактивное занятие	2	Работа с базами данных. Введение в SQL		Задания в среде программирования с расширением учебной платформы Алгоритмики
26				Интерактивное занятие	2	Работа со связанными таблицами. Создание скрипта базы данных для сайта		Задания в среде программирования с расширением учебной платформы Алгоритмики
27				Интерактивное занятие	2	Взаимодействие сайта с несколькими пользователями. Сессии		Задания в среде программирования с расширением учебной платформы Алгоритмики
28				Интерактивное занятие	2	Создание форм		Задания в среде программирования с расширением учебной платформы Алгоритмики
29				Интерактивное занятие	2	Использование шаблонов		Задания в среде программирования с расширением учебной платформы Алгоритмики

30				Интерактивное занятие	2	Презентация проектов		Защита проекта
31				Интерактивное занятие	2	Варианты развития карьеры в IT		Задания в среде программирования с расширением учебной платформы Алгоритмики
32				Интерактивное занятие	2	Выпускной: «Моя карьера в IT»		Защита проекта

### Список литературы

1. Кларин М.В. Инновационные модели обучения: Исследование мирового опыта. Монография. 2-е издание. — М.: ЛУч, 2018. — 640 с.
2. Лапчик М.П., Семакин И.Г., Хеннер Е.К., Самылкина Н.Н. Теория и методика обучения информатике. — М.: Академия, 2008. — 592 с.
3. Федеральный государственный образовательный стандарт основного общего образования (5-9 кл.) [Электронный ресурс] Режим доступа: [минобрнауки.рф/документы/543](http://минобрнауки.рф/документы/543).
4. Федеральный закон «Об образовании в РФ» [Электронный ресурс] Режим доступа: [consultant.ru/document/cons\\_doc\\_LAW\\_140174/](http://consultant.ru/document/cons_doc_LAW_140174/)
5. Актуальные проблемы методики обучения информатике в современной школе : материалы Международной научно-практической интернет-конференции, г. Москва, 24–26 апреля 2018 г. / под ред. Л. Л. Босовой, Н. К. Нателаури ; Московский педагогический государственный университет. Кафедра теории и методики обучения информатике [Электронное издание]. – Москва: МПГУ, 2018. – 222 с.
6. А.В. Хуторской. Компетентностный подход в обучении. Научно-методическое пособие. — М.: Издательство «Эйдос»; Издательство Института образования человека, 2013. — 73 с.
7. А.А. Вербицкий, О.Г. Ларионова: Личностный и компетентностный подходы в образовании: проблемы интеграции. Монография. — М.: Издательство «Логос», 2017. — 336 с.
8. К. Вольдерман, К. Стили, К. Квигли, М. Гудфеллоу, Д. Маккафферти, Дж. Вудкок. Программирование на Python: иллюстрированное руководство для детей. — М.: Манн, Иванов и Фербер, 2018. — 224 с.: илл.
9. Дж. Бриггс. Python для детей. Самоучитель по программированию. — М.: Манн, Иванов и Фербер, 2020.
10. Салман Хан. Весь мир - школа. Преобразованное образование/ The One World Schoolhouse: Education Reimagined. [текст]- Арт-транзит, Классика-XXI, 2015. — 176 с.
11. Самылкина Н.Н. Современные средства оценивания

результатов обучения. М.: БИНОМ, 2007. – 172 с.

12. М.Э. Абрамян Programming Taskbook Электронный задачник по программированию Версия 4.6 Ростов-на-Дону 2007.

13. Босова Л.Л. Как учат программированию в XXI веке: отечественный и зарубежный опыт обучения программированию в школе. // Информатика в школе. 2018. №6. С. 3-11.