

Муниципальное бюджетное общеобразовательное учреждение  
«Основная общеобразовательная школа № 9»

Рассмотрена на заседании педсовета  
«16» мая 2023 года  
Протокол № 5  
от «16» мая 2023 г.

Утверждаю  
Директор МБОУ ООШ №9  
Н.В. Кузьмина  
Приказ № 44-08  
от «16» мая 2023 г.



Дополнительная общеобразовательная  
общеразвивающая программа  
«Программирование Python»

Направленность: техническая  
Возраст обучающихся: 7-9 класс  
Срок реализации: 1 год

Разработчик:  
Береснева Светлана Павловна,  
учитель информатики, 1 КК

с. Рудянское  
2023 год

## Пояснительная записка

В настоящее время прослеживается острый кризис квалифицированных кадров в различных сферах рынка труда. Работодатели, хоть и обращают внимание на область предметных знаний, но все же отдают предпочтения универсальным навыкам. Подобная позиция связана с постоянно изменяющимися условиями труда, технологическим прогрессом. Подобные универсальные навыки (способность работать в команде, многозадачность, творческий подход, критическое мышление, продуктивное мышление, нацеленность на результат) вырабатываются в процессе профессионального становления.

Характерной чертой развития общества на протяжении последних десятилетий является его все более расширяющаяся информатизация. Отражением и следствием этой тенденции явилась потребность в подготовке подрастающего поколения к вступлению в информационное общество, любая профессиональная деятельность в котором будет связана с информатикой и информационными технологиями. Изучение данного курса имеет важное значение для развития мышления подростков. В современной психологии отмечается значительное влияние изучения информатики и использования компьютеров в обучении на развитие у школьников теоретического, творческого мышления, а также формирование нового типа мышления, так называемого операционного мышления, направленного на выбор оптимальных решений; открывает новые возможности для овладения такими современными методами научного познания, как формализация, моделирование, компьютерный эксперимент и т.д. Алгоритмическое мышление является необходимой частью научного взгляда на мир. В то же время оно включает и некоторые общие мыслительные навыки, полезные и в более широком контексте.

Направленность дополнительной образовательной программы — техническая. Она заключается в развитии технического и творческого мышления у обучающихся среднего школьного возраста через разработку алгоритмов в цифровой среде, взаимодействие с приложениями, сервисами и инструментами вне зависимости от платформы или интерфейса, формирование навыка определения различных видов ошибок (логическая, синтаксическая, ошибка среды и ошибка взаимодействия) и принципах их исправления в процессе разработки с помощью процедур отладки. Также развитие у детей умения составлять план деятельности, стремления к познавательным активностям, исследовательских, прикладных способностей, формирование навыков сотрудничества и коллективной работы в цифровой среде.

Развитие навыков программирования является уникальным способом формирования интереса учащихся к техническим направлениям деятельности, а также совершенствования алгоритмического, креативного 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; разработка, внедрение цифровых технологий и платформенных решений.

Python – это язык программирования общего назначения, распространяемый с открытыми исходными текстами. Он оптимизирован для создания качественного программного обеспечения. Язык Python используется сотнями тысяч разработчиков по всему миру в таких областях, как создание веб-сценариев, системное программирование, создание пользовательских интерфейсов, настройка программных продуктов под пользователя, численное программирование и в других.

Программа «Программирование на Python» представляет собой углубленный курс по программированию, дающий представление о понятиях структурного программирования (данных, операциях, переменных, ветвлениях в программе, циклах и

функциях, массивах, файлах). Программа направлена на развитие логического и пространственного мышления обучающегося, способствует раскрытию творческого потенциала личности, приобретению практических умений и навыков в области компьютерных технологий, способствует интеллектуальному развитию ребенка.

Наш выбор Python для преподавания призван помочь обучающимся сделать первые шаги по одному из современных и перспективных путей развития IT-индустрии.

Курс служит средством внутрипрофильной специализации в области новых информационных технологий, что способствует созданию дополнительных условий для проявления индивидуальных образовательных интересов учащихся, их дальнейшей профессиональной ориентации.

Данная программа относится к углубленному уровню, так как направлена на овладение знаниями и развитие навыков, она дает возможность ребенку погрузиться в атмосферу дополнительного образования, попробовать себя в новом лично значимом виде деятельности. Родителям дает возможность разобраться с логикой дополнительного образования и наметить линию индивидуального развития своего ребенка.

Программа базируется на решении кейсов и проектной деятельности. Для повышения качества образования данные кейсы основываются на реальных ситуациях и проблемах потенциальных работодателей. Это дает возможность ранней профессиональной ориентации обучающихся.

По форме организации содержания и процесса педагогической деятельности программа «Программирование на Python» интегрированная, так как объединяет в целое отдельные образовательные области на основе единства математики и информатики. Главной целью является приобретение обучающимся теоретических знаний и практических навыков и компетенций, достаточных для дальнейшего успешного обучения в системе непрерывного образования: школа – допобразование – ВУЗ – предприятие. Сущностью программы можно определить как развитие мышления и формирование мировоззрения в условиях преподавания интегрированного курса информатики, математики, программирования обучающихся среднего и старшего звена. Изучение всех этих предметов должно быть не целью, а средством познания мира, давать возможность обучающимся проникать в сущность изучаемых проблем. В теории интеграции в данном случае вкладывается понятие взаимосвязи, взаимопроникновения математики и информатики, что предполагает качественное изменение в параметрах нового объекта. В школьном курсе информатики вопросы программирования рассматриваются лишь в ознакомительном плане и на это выделяется недостаточное количество часов, как следствие – формальное восприятие обучающимися основ современного программирования. Образовательная программа «Программирование на Python» направлена на устранение данного пробела.

**Целью курса** – способствовать формированию алгоритмического и критического мышления, навыков проектной деятельности через использование языка программирования Python как инструмента для реализации идей в области своих интересов.

#### **Задачи программы:**

##### Обучающие:

- 1.Познакомить с базовыми понятиями основ программирования.
- 2.Познакомить с особенностями синтаксиса языка программирования Python.
- 3.Познакомить с принципами объектно-ориентированного программирования.
- 4.Сформировать навык правильного оформления кода.
- 5.Сформировать навык использования итеративного подхода при решении различных задач.
6. Формировать навык решения задач и разработки проектов с помощью Python.
7. Формировать навык разработки игр с помощью языка программирования Python.
8. Формировать навык тестирования и оптимизации созданных проектов.

##### Развивающие:

1. Формировать и развивать логическое, алгоритмическое, критическое, креативное и пространственное мышление.
2. Формировать навык публичного выступления и презентации.
3. Формировать навык планирования деятельности и основ тайм-менеджмента.
4. Формировать и расширять словарный запас на английском языке.
5. Формировать функциональную грамотность.
6. Расширять кругозор, развивать память, внимание, творческое воображение, абстрактно-логические и наглядно-образные виды мышления и типы памяти, основные мыслительные операции и свойства внимания.
7. Совершенствовать диалогическую речь учащихся: уметь слушать собеседника, понимать вопросы, смысл знаний, 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.

**Направленность:** техническая.

**Возраст обучающихся:** учащиеся 7-9 классов.

**Режим занятий:** на реализацию программы отводится 1 академический час в неделю (всего 34 часа в год).

**Срок реализации:** 1 год.

**Формы работы.** Основной формой обучения являются групповые занятия. В основе образовательного процесса лежит проектный подход. Основная форма работы теоретической части – лекционные занятия в группах. Практические задания планируется выполнять индивидуально, в парах и в малых группах. Занятия проводятся в виде бесед, семинаров, лекций, консультаций, занятий-соревнований: для наглядности изучаемого материала используется различный мультимедийный материал – презентации, видеоролики.

**Материально-техническое обеспечение реализации дополнительной программы осуществляется с помощью оборудования «Точка Роста»: компьютерное оборудование (ноутбук, мультимедийный проектор, многофункциональное устройство).** Примерный перечень характеристик формируется с учетом положений КТРУ, СП 2.4.3648-20 "Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи". При формировании примерных характеристик также возможно использование положений приказа Министерства просвещения Российской Федерации, Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации от 08.09.2021 № 634/925 «Об утверждении стандарта оснащения государственных и муниципальных общеобразовательных организаций, осуществляющих образовательную деятельность в субъектах Российской Федерации, на территории которых проводится эксперимент по внедрению цифровой образовательной среды, компьютерным, мультимедийным, презентационным оборудованием и программным обеспечением» (Зарегистрирован 16.12.2021 № 66360).

**Планируемые результаты освоения курса.**

**Личностные:** умение работать в команде: работа в общем ритме, эффективное распределение задач и другие; формирование высокого познавательного интереса учащихся; формирование критического мышления; проявление технического мышления, познавательной деятельности, творческой инициативы, самостоятельности.

**Метапредметные:** умение ориентироваться в информационном пространстве, продуктивно использовать техническую литературу для поиска сложных решений; способность творчески решать технические задачи; готовность и способность применения теоретических знаний по физике, информатике для решения задач в реальном мире;

способность правильно организовывать рабочее место и время для достижения поставленных целей; Знание основ ТРИЗ, навыки публичного выступления и презентации результатов, навык генерации идей.

**Предметные:** знание основ и принципов программирования; знание и понимание основных алгоритмических конструкций; знание основ и овладение практическими базисными знаниями программирования в Python; знание основ и овладение практическими базисными навыками разработки игр в PyGame.

## УЧЕБНЫЙ ПЛАН

№ п/п	Название раздела темы	Количество часов			Формы аттестации/контроля
		всего	теория	практика	
1	Введение в Python	2	1	1	Тест
2	Типы данных и операции.	6	3	3	Тест
3	Инструкции и синтаксис	7	3	4	Практическая работа
4	Функции и модули в программировании	9	4	5	Практическая работа
5	Сложные типы данных	7	2	5	Тест
6	Написание и отладка программ. Самостоятельная работа	3	1	2	Защита проекта
		34	14	20	

### Содержание учебного плана

#### Раздел 1. Введение в Python

1.3. Инструкции и структура программы. Техника безопасности. История создания. Области применения и перспективы. Инструкции и структура программы. Основы Алгоритмизации. Достоинства языка.

1.4. Установка Python. Ввод и вывод данных. Установка Python. Доступ к документации. Основы ввода и вывода данных. Первая программа на языке Python. Основы Алгоритмизации.

Практическая работа: Решение задач для развития логики и понимания работы алгоритма. Установка программы Python. Написание первой программы вывод на экран.

#### Раздел 2. Типы данных и операции

с. Переменные. Операторы. Переменные. Оператор присваивания. Имена переменных и ключевые слова. Выражения. Операции. Порядок выполнения операций. Математические функции. Композиция. Ввод и вывод. Ввод данных с клавиатуры. Вывод данных на экран. Практическая работа: Решение задач на элементарные действия с числами. Создание программы простейший калькулятор.

d. Типы данных. Базовые функции для работы с различными типами данных. Типы данных. Преобразование типов. Различия типов данных. Базовые функции (abs, round, int, math).

Практическая работа: Решение математических задач с использованием функции import math.

#### Раздел 3. Инструкции и синтаксис

3.3. Условные операторы if, elif, else. Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or,

not). Условный оператор. Альтернативное выполнение. Примеры решения задач с условным оператором. Множественное ветвление. Реализация ветвления в языке Python.

Практическая работа: Практическое закрепление знаний по условным операторам. Создание программ.

3.4. Циклы while, for. Обработка исключений. Случайные числа. Понятие цикла. Тело цикла. Условия выполнения тела цикла. Оператор цикла с условием. Оператор цикла while. Бесконечные циклы. Альтернативная ветка цикла while. Обновление переменной. Краткая форма записи обновления. Примеры использования циклов. Оператор цикла с параметром for. Операторы управления циклом. Пример задачи с использованием цикла for. Вложенные циклы. Циклы в циклах. Случайные числа. Функция randint. Функция random. Примеры решения задач с циклом.

Практическая работа: Числа Фибоначчи. Решение задачи с циклом for. Создание игры угадай число. Повторение.

#### **Раздел 4. Функции и модули в программировании**

4.3. Встроенные и пользовательские функции. Создание функций. Параметры и аргументы. Локальные и глобальные переменные. Поток выполнения. Функции, возвращающие результат. Анонимные функции, инструкция lambda. Примеры решения задач с использованием функций. Рекурсивные функции. Вычисление факториала.

Практическая работа: Создание игры русская рулетка. Создание всех ранее созданных программ с использованием функций.

4.4. Строки - последовательности символов. Составной тип данных - строка. Доступ по индексу. Длина строки и отрицательные индексы. Преобразование типов. Применение цикла для обхода строки. Срезы строк. Строки нельзя изменить. Сравнение строк. Оператор in. Модуль string. Операторы для всех типов последовательностей (строки, списки, кортежи). Примеры решения задач со строками.

Практическая работа: Написание программ. Работа со строками.

#### **Раздел 5. Сложные типы данных**

5.1. Списки. Срезы списков. Списки. Тип список (list). Индексы. Обход списка. Проверка вхождения в список. Добавление в список. Суммирование или изменение списка. Операторы для списков. Срезы списков. Удаление списка. Клонирование списков. Списочные параметры. Функция range. Списки: примеры решения задач.

Практическая работа: Написание программ. Работа со списками.

5.2. Матрицы Матрицы. Вложенные списки. Матрицы. Строки и списки. Генераторы списков в Python. Установка Python. Доступ к документации. Ввод и вывод данных. Первая программа на языке Python.

Практическая работа: Создание программ. Практическое закрепление полученных знаний.

5.3. Кортежи Кортежи. Присваивание кортежей. Кортежи как возвращаемые значения.

Практическая работа: Создание программ. Практическое закрепление полученных знаний.

5.4. Словари Введение в словари. Тип словарь (dict). Словарные операции. Словарные методы.

Практическая работа: Создание программ. Практическое закрепление полученных знаний.

5.5. Множества в языке Python Множества в языке Python. Множества. Множественный тип данных. Описание множеств. Операции, допустимые над множествами: объединение, пересечение, разность, включение. Оператор определения принадлежности элемента множеству.

Практическая работа: Создание программ. Практическое закрепление полученных знаний.

**Раздел 6. Написание и отладка программ. Самостоятельная работа.**  
***Практическая работа:*** Стиль программирования. Отладка программ.  
**Проектная работа по курсу «Программирование на языке Python».**

**Тематическое планирование**

Класс 7-9 класс

Количество часов (годовых / недельных) 34/1

<b>№ п/п</b>	<b>Раздел/Тема урока</b>	<b>Кол-во часов</b>	<b>Дата</b>
1	Инструкции.	1	
2	Структура программы.	1	
3	Установка Python.	1	
4	Ввод и вывод данных.	1	
5	Переменные.	1	
6	Операторы.	1	
7	Типы данных.	1	
8	Преобразование типов.	1	
9	Различия типов данных.	1	
10	Базовые функции.	1	
11	<b>Базовые функции и (abs)*</b>	1	
12	<b>Базовые функции и (round)*</b>	1	
13	<b>Базовые функции и (int)*</b>	1	
14	<b>Базовые функции и (math)*</b>	1	
15	<b>Условный оператор if*</b>	1	
16	<b>Альтернативный условный оператор elif, else*</b>	1	
17	<b>Циклы while. Цикл for.*</b>	1	
18	<b>Обработка включений.*</b>	1	
19	Функция random. Случайные числа.	1	
20	<b>Решение задач.*</b>	1	
21	Встроенные пользовательские функции.	1	
22	<b>Создание функций.*</b>	1	
23	Функции, возвращающие результат.	1	
24	<b>Строки. Последовательность символов.*</b>	1	
25	<b>Доступ по индексу. Отрицательные индексы.*</b>	1	
26	<b>Применение цикла для обхода строки.*</b>	1	
27	<b>Списки. Срезы списков.*</b>	1	
28	<b>Матрицы.*</b>	1	
29	<b>Словари.*</b>	1	
30	<b>Множества в языке Python.*</b>	1	
31	<b>Написание и отладка программ.*</b>	1	
32	<b>Написание и отладка программ.*</b>	1	
33	<b>Самостоятельная работа.*</b>	1	
34	Самостоятельная работа.	1	

\* Шрифтом выделены уроки, проводимые с оборудованием центра «Точка роста»

## **СПИСОК ЛИТЕРАТУРЫ**

### **Список литературы, используемой педагогом**

#### **Основная:**

1. Клейнберг Дж. Алгоритмы: разработка и применение. СПб: Питер, 2016. - 800 с.
2. Златопольский Д. М. Основы программирования на языке Python. – М.: ДМК Пресс, 2017. – 284 с.: ил.



3. Бхаргава А. Грокаем алгоритмы. Иллюстрированное пособие для программистов и любопытствующих. - СПб.: Питер, 2017. - 288 с.

**Дополнительная:**

1. Луц М. Изучаем Python. СПб: Симво-плюс, 2011. - 1280 с.
2. Паронджанов В.Д. Учись писать, читать и понимать алгоритмы. М: Ямб, 2012. – 520 с.
3. Златопольский Д.М. Сборник задач по программированию. СПб: БХВПетербург, 2011. - 295 с.
4. ООП на Python: концепции, принципы и примеры реализации [Электронный ресурс]. – Режим доступа: <https://proglib.io/p/python-oop/>

**Список рекомендуемой литературы для обучающихся**

**Основная:**

1. Свейгар.Эл. Учим python, делая крутые игры М: Эксмо, 2018. – 416 с.
2. Семакин И.Г. Основы алгоритмизации и программирования. М.: Академия, 2016. - 304 с.

**Дополнительная:**

1. Луц М. Изучаем Python. СПб: Симво-плюс, 2011. - 1280 с.
2. Уроки по Python для начинающих [Электронный ресурс]. – Режим доступа: <https://pythonru.com/uroki/vvedenie-uroki-po-python-dlja-nachinajushhih>
3. Алгоритмизация. Программирование Python 3 [Электронный ресурс]. – Режим доступа: <https://stepik.org/course/7215/prom>