

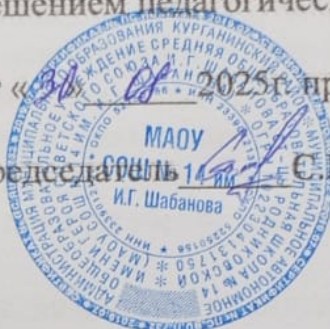
Муниципальное автономное общеобразовательное учреждение
средняя общеобразовательная школа №14 имени Героя Советского Союза
И.Г. Шабанова ст-цы Родниковской

УТВЕРЖДЕНО

решением педагогического совета

от «24» августа 2025г. протокол №1

председатель С.В.Солянов



РАБОЧАЯ ПРОГРАММА

внеурочной деятельности

«Питон. Программирование»

Уровень: 9-11 классы

Количество часов: 34

Учитель: Колтышева Н.М.

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.

2025-26 учебный год

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Актуальность и направленность программы.

Направленность программы внеурочного занятия «Программирование на Python» — 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.

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. Программа «Программирование на Python» направлена на устранение данного пробела.

Отличительные особенности программы и новизна. Отличительной особенностью программы является его направленность на формирование у учащихся навыков поиска собственного решения поставленной практической задачи, представленную в виде адаптированного кейса из реальной жизни, составления алгоритма решения и его реализации с помощью средств программирования. Для этого, учащиеся проходят через следующие этапы: определяют и составляют последовательность действий, ведущую к цели, и записывают ее с помощью формального языка. Полученные знания учащиеся применяют при создании собственных проектов, которые защищаются перед другими учениками, педагогами и родителями.

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.

Адресат программы. Программа ориентирована на дополнительное образование обучающихся 15-17 лет, интересующихся программированием.

Срок и этапы реализации программы. Программа рассчитана на 1 год обучения - 34 часа.

Основной формой обучения являются групповые занятия. В основе образовательного процесса лежит проектный подход.

Режим занятий: 1 раз по 1 час в неделю.

Основная форма работы теоретической части – лекционные занятия в группах до 15 человек. Практические задания планируется выполнять индивидуально, в парах и в малых группах. Занятия проводятся в виде бесед, семинаров, лекций: для наглядности изучаемого материала используется различный мультимедийный материал – презентации, видеоролики.

Цель программы

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

Достижение поставленной цели складывается из выполнения следующих задач:

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.

В результате освоения дисциплины обучающийся должен:

уметь:

~ работать в среде программирования;
~ реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
~ настраивать рабочую среду Python;

знать:

~ этапы решения задачи на компьютере;
~ типы данных;
~ базовые конструкции изучаемых языков программирования;
~ принципы структурного и модульного программирования;
~ принципы объектно-ориентированного программирования;
~ основы создания приложений в Python;
~ инструментальные средства Python.

Методы: кейс-метод, проектная деятельность.

Формы работы:

~ практическое занятие;
~ занятие – соревнование;
~ консультация;
~ самостоятельная работа.

Практические занятия составляют важную часть теоретической и профессиональной подготовки. Они направлены на формирование практических навыков и умений. Практические занятия создают оптимальные дидактические условия для деятельностного освоения обучающимися содержания и методологии изучаемой дисциплины, использование специального оборудования, технических средств. Ведущей дидактической целью практических занятий является формирование

практических умений – профессиональных (выполнять определённые действия, операции, необходимые в последующем в профессиональной деятельности).

Целью занятия-соревнования является создание условий для развития интеллекта и проявления способности к творчеству. Задачами такого занятия могут быть - повышение интереса к данному курсу в частности и познанию в общем, формирование навыка принимать необычные решения. Занятие-соревнование отличается от «традиционного» тем, что он учит выполнению работы по предмету, использованию теории на практике, коллективной деятельности, делать выводы.

Консультации являются одной из форм руководства самостоятельной работой обучающихся и оказания им помощи в освоении учебного материала. Они могут проводиться регулярно или внепланово по мере необходимости. Консультации помогают обучающемуся избрать правильные методы работы, ни в коей мере не снижая его ответственности за принятые решения. Часто применяются при работе обучающегося над итоговым проектом.

Самостоятельная работа обучающихся – это разнообразные виды деятельности обучающихся, осуществляемые под руководством, но без непосредственного участия педагога в специально отведенное для этого аудиторное или внеаудиторное время. Это особая форма обучения по заданиям педагога, выполнение которых требует активной мыслительной, поисково – исследовательской и аналитической деятельности. Методологическую основу самостоятельной работы обучающихся составляет деятельностный подход, когда цели обучения ориентированы на формирование умений решать типовые и нетиповые задачи, т. е. на реальные ситуации, где обучающимся надо проявить знание конкретной дисциплины, использовать внутрипредметные и межпредметные связи.

Виды учебной деятельности:

- ~ анализ проблемных учебных ситуаций;
- ~ построение гипотезы на основе анализа имеющихся данных;
- ~ поиск необходимой информации;
- ~ выполнение практических работ;
- ~ подготовка выступлений и докладов с использованием разнообразных источников информации;
- ~ публичное выступление.

Оборудование и программное обеспечение:

Компьютерное оборудование:

- ~ персональный компьютер – 10 шт.
- ~ моноблок – 1 шт.
- ~ маршрутизатор – 1 шт.
- ~ коммутатор – 1 шт.

Программное обеспечение:

- ~ ОС Windows

~ Python
~ Pyzo
~ Pygame
~ Текстовый редактор Блокнот
~ KompoZer
~ Microsoft Power point

Методическое обеспечение реализации программы

Методическое обеспечение программы включает приёмы и методы организации образовательного процесса, дидактические материалы, техническое оснащение занятий.

Методы и приёмы организации образовательного процесса при реализации программы:

Словесные методы: объяснение, беседа, комментированное чтение, рассказ.

Практические методы: работа с текстом, составление планов, работа над проектами, выполнение творческих заданий: составление кроссвордов, сочинение загадок, рассказов, выпуск бюллетеней, сборников или альбомов с творческими работами и проектами.

Игровые методы: фантазирование, театральная импровизация, живая наглядность.

Наглядные методы: показ видеоматериалов, посещение выставок, проведение экскурсий.

Виды дидактических материалов, используемые при реализации программы:

Для обеспечения наглядности и доступности изучаемого материала педагог использует наглядные пособия следующих видов:

- схематические или символические (оформленные стенды и планшеты, таблицы, схемы, рисунки, графики, плакаты, диаграммы, чертежи, шаблоны и т.п.);
- картинные (иллюстрации, слайды, фотоматериалы и др.);
- звуковые (аудиозаписи);
- смешанные (видеозаписи, учебные кинофильмы и т.д.);
- дидактические пособия (карточки, рабочие тетради, раздаточный материал, вопросы и задания для опроса, тесты, практические задания, упражнения и др.).
- компьютерные программы в электронном виде (компьютеры с программами, CD, флеш-носители);
- учебные пособия, журналы, книги, Интернет-ресурсы.

При реализации программы с целью повышения качества и эффективности процесса обучения применяются современные эффективные технологии обучения, ориентированные не на накопление знаний, а на организацию активной деятельности обучающихся:

- технологии проектной деятельности;
- компьютерные (информационные) технологии;

- технологии учебно-игровой деятельности (моделирование);
- технологии коммуникативно-диалоговой деятельности;
- модульные технологии;
- квест-технологии;
- технологии личностно-ориентированного обучения;
- кейс-технологии.

Информационные технологии используются в различных видах деятельности:

- при подготовке и проведении занятий;
- для создания авторских мультимедийных презентаций;
- в рамках индивидуальной и групповой проектной деятельности;
- для самостоятельной работы;
- для накопления демонстрационных материалов к занятиям (видеоматериалы, таблицы, презентации, карты);

Одним из основных методов является метод проектного обучения, так как он является неотъемлемой частью учебного процесса. Исходный лозунг основателей системы проектного обучения – «Все из жизни, все для жизни». Обучение строится на активной основе, через практическую деятельность ученика, ориентируясь на его личный интерес и практическую востребованность полученных знаний в дальнейшей жизни, обучающийся имеет возможность через проектную деятельность освоить получаемые знания. Проекты представляются в виде готовых программ, презентаций проектов, научных докладов, моделей, демонстрации видеофильма. Достоинствами проектной деятельности являются:

- Уметь работать в коллективе;
- Брать ответственность за выбор решения на себя;
- Разделять ответственность с другими;
- Предоставлять ребенку свободу выбора темы, методов работы;
- Понимание каждым обучающимся важности работы и др.

Содержание учебного плана

Раздел 1. Введение в Python

1.1. Инструкции и структура программы.

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

1.2. 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

а. 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.

б. 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

3.1. 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.

3.2. 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 randrange. Функция random. Примеры решения задач с циклом.

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

Раздел 4. Функции и модули в программировании

4.1. 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.

4.2. 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. Словари

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

Практическая работа: Создание программ. Практическое закрепление полученных знаний.

5.4. 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».

Согласовано

Руководитель Центра образования
Гуманитарного и цифрового профилей
«Точка роста»

_____ И.М. Казанкина

Тематическое планирование с указанием количества часов

№	Тема занятий	Кол-во часов	Дата		примечание
			план	факт	
1	Вводное занятие. Инструктаж правила безопасности	1			
2	Установка Python.	1			
3	Инструкции и структура программы.	1			
4	Переменные. Операторы.	1			
5	Типы данных. Преобразование типов. Различия типов данных.	1			
6	Базовые функции (abs, round, int, math)	1			
7	Условный оператор if. Альтернативный условный оператор elif, else.	2			
8	Циклы while и for. Обработка вложений.	2			
9	Функция random. Случайные числа. Встроенные и пользовательские функции. Создание функций. Функции, возвращающие результат.	3			
10	Строки, последовательность символов. Доступ по индексу. Длина строки и отрицательные индексы. Преобразование типов. Применение цикла для обхода строки. Срезы строк. Сравнение строк.	3			
11	Списки. Срезы списков. Решение задач со списками	2			
12	Матрицы	1			

13	Словари	1			
14	Множества в языке Python	1			
15	Написание и отладка программ. Самостоятельная работа	10			
32	Защита проекта	3			
	Итого	34			