



ИРМО

ОШ»

С.А.Шуткин

20 25

**Дополнительная общеразвивающая программа
«Программирование на языке Python»**

с использованием оборудования центра «Точка роста»

2025-2026 учебный год

Структура программы

Пояснительная записка	3
Учебный план	6
Календарный учебный график.....	6
Учебно-тематическое планирование.....	7
Содержание разделов программы	14
Оценочные материалы	15
Методические материалы.....	16

Пояснительная записка

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. Как считают многие, один из самых используемых языков программирования в мире.

Содержание обучения, представленное в программе курса «Программирование на языке Python», позволяет вести обучение школьников в режиме актуального познания. Практическая направленность курса на создание внешних образовательных продуктов блок-схем, алгоритмов, программ — способствует выявлению фактов, которые невозможно объяснить на основе имеющихся у школьников знаний. Возникающие при этом познавательные переживания обуславливают сознательное отношение к изучению основных теоретических положений информатики.

В рамках предлагаемого курса изучение основ программирования на языке Python это не столько средство подготовки к будущей профессиональной деятельности, сколько формирование новых общеинтеллектуальных умений и навыков: разделение задачи на этапы решения, построение алгоритма и др.

Направленность программы «Программирование на языке Python» - техническая.

Актуальность программы состоит в том, что активизация познавательного процесса позволяет учащимся более полно выражать свой творческий потенциал и реализовывать собственные идеи в изучаемой области знаний, создаёт предпосылки по применению информационных компетенций в других учебных курсах, а также способствует возникновению дальнейшей мотивации, направленной на освоение профессий, связанных с разработкой программного обеспечения.

С развитием современных информационных технологий сегодня любой учащийся под руководством опытного педагога может с лёгкостью научиться программировать. Компьютеры и компьютерные системы – неотъемлемая часть жизни нашего общества. Научившись программировать, мы можем быть не только пользователями информационных технологий, но и активными их создателями.

Новизна и отличительные особенности программы

Новизна Программы заключается в том, что Python дает более широкие возможности в области программирования, чем Pascal, который входит в школьный курс информатики. На языке Python можно легко и быстро создавать простые компьютерные игры, трёхмерные модели и программировать роботов. Этот язык быстрее и легче усваивается, чем

Pascal. Многие мировые компании такие, как Intel, Cisco, Hewlett-Packard, используют этот язык при реализации своих проектов. Крупнейшие интернет-ресурсы такие, как Google, YouTube, также разработаны с помощью языка программирования Python.

Адресат программы

Программа рассчитана на обучение детей 13-17 лет.

Сроки реализации программы «Программирование на языке Python»

1 год.

Количество часов на каждый год: 162 часа.

Форма обучения – очная.

Формы и режим занятий

В каждой группе проводится 3 занятия в неделю по 1,5 учебных часа. Перерыв между занятиями составляет 5 минут. Наполняемость групп: 14-15 человек.

Формы организации обучения детей на занятии: групповая, фронтальная и индивидуальная. Виды проведения занятий: теоретическое и практическое занятие.

Цель программы: ознакомить обучающихся с объектно-ориентированным языком программирования Python, и обучить практическим навыкам программирования для решения типовых задач математики и информатики, а также при разработке простейших игр.

Задачи:

Образовательные:

- обучить языку программирования Python и созданию программ на его основе;
- научить создавать прикладное программное обеспечение;
- расширять кругозор обучающихся в области программирования;
- научить дизайнерскому оформлению созданного ПО.

Развивающие:

- развитие умения творчески подходить к решению задачи;
- развивать память и внимание, познавательную и творческую активность;
- развивать творческие способности, эстетическое и эргономическое восприятие объектов труда;
- развивать логическое мышление.

Воспитательные:

- прививать интерес к активному творческому самовыражению, культуре труда;
- воспитывать упорство в достижении желаемого результата;
- воспитывать чувство взаимопомощи, доверия, коллективизма.

Планируемые результаты обучения:

Личностные:

- владение навыками алгоритмического мышления и понимание необходимости формального описания алгоритмов;
- владение стандартными приёмами написания программы для решения стандартной задачи с использованием основных конструкций программирования и отладки таких программ на языке Python;
- владение навыками и опытом разработки программ на Python, включая тестирование и отладку программ;
- владение элементарными навыками формализации прикладной задачи

Метапредметные:

- умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;
- умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата, понимая, что в программировании длинная программа – не значит лучшая программа;
- умение критически оценивать правильность решения учебно-исследовательской задачи;
- умение корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями;
- владение основами самоконтроля, принятия решений;
- способность и готовность к самостоятельному поиску методов решения практических задач, применению различных методов познания.

Предметные:

- владение навыками алгоритмического мышления и понимание необходимости формального описания алгоритмов;
- владение стандартными приёмами написания программы для решения стандартной задачи с использованием основных конструкций программирования и отладки таких программ на языке Python;
- владение навыками и опытом разработки программ на Python, включая тестирование и отладку программ;
- владение элементарными навыками формализации прикладной задачи.

Формы и методы контроля результативности обучения:

- практические работы (разработка и отладка программ на языке программирования Python)

Учебный план

№	Название раздела	Количество часов			Форма промежуточной аттестации
		Всего	Теория	Практика	
1	Введение в Python	25,5	7,5	18	Практическая работа
2	Алгоритмические конструкции. Рекурсия	33	6,5	26,5	Практическая работа
3	Множества. Словари	25,5	6	19,5	Практическая работа
4	Символьные строки	33	7,5	25,5	Практическая работа
5	Матрицы	42	10	32	Практическая работа
6	Итоговое занятие	3	-	3	Практическая работа
ИТОГО		162	37,5	124,5	

Календарный учебный график

Учебный год по дополнительной общеразвивающей программе «Программирование на языке Python» начинается с 1 сентября и заканчивается 30 мая.

Реализация программы начинается с 1 сентября. Продолжительность занятия – 45 минут. Занятия проводятся 3 раза в неделю.

Оптимальное количество учащихся в группе для успешного освоения программы 14-15 человек.

Количество часов по месяцам								
сентябрь	октябрь	ноябрь	декабрь	январь	февраль	март	апрель	май
18	22,5	18	18	13,5	18	18	21	15

Учебно-тематическое планирование

№ п/п	Наименование разделов	Количество часов			Форма промежуточной аттестации
		всего	теория	практика	
1	Введение в Python	25,5	7,5	18	
	Язык Python. Среда программирования	1,5	1	0,5	Беседа + Практическая работа
	Оператор ввода – input	1,5	0,5	1	Беседа + Практическая работа
	Оператор вывода – print	1,5	0,5	1	Беседа + Практическая работа
	Типы данных	1,5	0,5	1	Беседа + Практическая работа
	Тип данных int (целочисленный)	1,5	0,5	1	Беседа + Практическая работа
	Операции над типом int (целочисленное деление, остаток от деления)	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Тип данных float (вещественный)	1,5	0,5	1	Беседа + Практическая работа
	Операции над типом float	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Ветвление. Полное ветвление	1,5	0,5	1	Беседа + Практическая работа
	Ветвление. Неполное ветвление	1,5	0,5	1	Беседа + Практическая работа
	Знакомство с конструкцией if...elif...else	1,5	0,5	1	Беседа + Практическая работа

	Логические операнды	1,5	0,5	1	Беседа + Практическая работа
	Решение задач с помощью and, or	1,5	0,5	1	Беседа + Практическая работа
	Вложенное ветвление	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
2	Алгоритмические конструкции. Рекурсия	33	6,5	26,5	
	Знакомство с понятием «цикл»	1,5	0,5	1	Беседа + Практическая работа
	Цикл for, итерация цикла	1,5	0,5	1	Беседа + Практическая работа
	Решение задач с помощью for	1,5		1,5	Практическая работа
	Выход из цикла for с помощью break	1,5	0,5	1	Беседа + Практическая работа
	Конструкция for...else	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Цикл с условием	1,5	0,5	1	Беседа + Практическая работа
	Решение задач на цикл с условием	1,5		1,5	Практическая работа
	Цикл while	1,5	0,5	1	Беседа + Практическая работа
	Решение задач с помощью while	1,5		1,5	Практическая работа
	Выход из цикла while с помощью break	1,5	0,5	1	Беседа + Практическая работа
	Понятие вложенного	1,5	0,5	1	Беседа +

	цикла				Практическая работа
	Решение задач на все типы циклов	1,5		1,5	Практическая работа
	Решение задач на все типы циклов	1,5		1,5	Практическая работа
	Понятие функции – def	1,5	0,5	1	Беседа + Практическая работа
	Описание переменных в функции, локальные и глобальные переменные	1,5	0,5	1	Беседа + Практическая работа
	Решение задач на вызов функции	1,5		1,5	Практическая работа
	Понятие рекурсии в python	1,5	0,5	1	Беседа + Практическая работа
	Понятие рекурсии в python	1,5	0,5	1	Беседа + Практическая работа
	Решение типовых задач на рекурсию	1,5		1,5	Практическая работа
	Решение задач на замену цикла for рекурсией	1,5	0,5	1	Беседа + Практическая работа
	Решение задач на замену рекурсии циклом for	1,5		1,5	Практическая работа
3	Множества. Словари	25,5	6	19,5	
	Понятие «Множества» в python	1,5	0,5	1	Беседа + Практическая работа
	Ввод множества с клавиатуры	1,5	0,5	1	Беседа + Практическая работа
	Запись в множества с помощью if	1,5	0,5	1	Беседа + Практическая работа
	Проход по значениям множества с помощью for	1,5	0,5	1	Беседа + Практическая работа
	Решение задач по теме	1,5		1,5	Практическая

	«Множества»				работа
	Понятие «Словари» в python	1,5	0,5	1	Беседа + Практическая работа
	Ввод с клавиатуры в словари	1,5	0,5	1	Беседа + Практическая работа
	Понятие значения в словарях	1,5	0,5	1	Беседа + Практическая работа
	Понятие значения ключа в словарях	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Запись в множества через условия	1,5	0,5	1	Беседа + Практическая работа
	Добавления элементов в множество	1,5	0,5	1	Беседа + Практическая работа
	Знакомство с методами множества	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Проход по ключам-значениям в множестве	1,5	0,5	1	Беседа + Практическая работа
	Решение задач по множествам	1,5		1,5	Практическая работа
	Решение задач по множествам	1,5		1,5	Практическая работа
4	Символьные строки	33	7,5	25,5	
	Ввод с клавиатуры	1,5	0,5	1	Беседа + Практическая работа
	Операции над строками	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Срезы в строках	1,5	0,5	1	Беседа + Практическая

					работа
	Проход по элементам в строках с помощью for	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Создание пустой строки	1,5	0,5	1	Беседа + Практическая работа
	Добавление элементов в новую строку через условие	1,5	0,5	1	Беседа + Практическая работа
	Вывод элементов строки в одну строку	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Методы строк. Проверка на числа	1,5	0,5	1	Беседа + Практическая работа
	Методы строк. Проверка на буквы	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Методы строк. Проверка на заглавные символы	1,5	0,5	1	Беседа + Практическая работа
	Методы строк. Поиск по значению	1,5	0,5	1	Беседа + Практическая работа
	Методы строк. Метод count	1,5	0,5	1	Беседа + Практическая работа
	Методы строк. Метод reversed	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Методы строк. Перевод из строки в число	1,5	0,5	1	Беседа + Практическая работа

	Методы строк. Перевод из типа int в тип str	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
5	Матрицы	42	10	32	
	Понятие массива в программировании	1,5	0,5	1	Беседа + Практическая работа
	Понятие list (список) в Python	1,5	0,5	1	Беседа + Практическая работа
	Ввод списка с клавиатуры	1,5	0,5	1	Беседа + Практическая работа
	Вывод списка	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Понятие индекса и значение массива (списка)	1,5	0,5	1	Беседа + Практическая работа
	Методы списка. Add	1,5	0,5	1	Беседа + Практическая работа
	Методы списка. Pop	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Методы списка. Copy	1,5	0,5	1	Беседа + Практическая работа
	Методы списка. Clear	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Методы списка. Reverse	1,5	0,5	1	Беседа + Практическая работа
	Методы списка. Index	1,5	0,5	1	Беседа +

					Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Понятие файла в программирование	1,5	0,5	1	Беседа + Практическая работа
	Открытие файла в python	1,5	0,5	1	Беседа + Практическая работа
	Запись в файл в python	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Прохождение по элементам списка	1,5	0,5	1	Беседа + Практическая работа
	Понятие «Список в списке» (двумерный массив)	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
	Перевод элемента списка в тип данных int	1,5	0,5	1	Беседа + Практическая работа
	Перевод из типа данных str в list	1,5	0,5	1	Беседа + Практическая работа
	Перевод из типа данных list в str	1,5	0,5	1	Беседа + Практическая работа
	Знакомство с методом join	1,5	0,5	1	Беседа + Практическая работа
	Решение задач	1,5		1,5	Практическая работа
6	Итоговое занятие	3	-	3	
	Решение задач	3		3	Беседа + Практическая работа
Итого за учебный год		162	37,5	124,5	

Содержание разделов программы

Раздел 1. Введение в Python (25,5 ч.)

Язык программирования Python. Достоинства и недостатки. Области применения. Интерактивный режим работы программы. Ввод и вывод числовой информации. 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. Первая программа

Раздел 2. Алгоритмические конструкции. Рекурсия (33 ч.)

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.

Раздел 3. Множества. Словари (25,5 ч.)

Введение в словари. Тип словарь (dict). Словарные операции. Словарные методы. Множества в языке Python. Словари со смешанными значениями. Кодирование и декодирование текста. Множества. Множественный тип данных. Описание множеств. Операции, допустимые над множествами: объединение, пересечение, разность, включение. Оператор определения принадлежности элемента множеству.

Раздел 4. Символьные строки (33 ч.)

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.

Раздел 4. Матрицы (42 ч.)

Матрицы. Вложенные списки. Матрицы. Строки и списки. Генераторы списков в Python. Вычисление сложности алгоритма. Многомерные списки. 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.

Раздел 5. Итоговое занятие (3 ч.)

Выполнение итоговой работы (разработка технической документации, составление программы для предложенной задачи).

Оценочные материалы

Оценочные материалы представляют собой систематизированные материалы наблюдений за процессом овладения знаниями, умениями, навыками, компетенциями, предусмотренными образовательной программой; материалы для проведения текущего контроля успеваемости и промежуточной аттестации учащихся.

Входящий контроль осуществляется при комплектовании группы в начале учебного года. Цель – определить исходный уровень знаний учащихся, определить формы и методы работы с учащимися.

Форма контроля: тестирование.

Текущий контроль осуществляется после изучения отдельных тем, раздела программы. В практической деятельности результативность оценивается качеством выполнения практических работ, поиску и отбору необходимого материала, умению работать с различными источниками информации. Анализируются положительные и отрицательные стороны работы, корректируются недостатки. Контроль знаний осуществляется с помощью заданий педагога (решение практических задач средствами языка программирования); взаимоконтроля, самоконтроля и др. Они активизируют, стимулируют работу учащихся, позволяют более полно проявлять полученные знания, умения, навыки.

Промежуточный контроль осуществляется в конце I полугодия учебного года.

Форма контроля: тест, решение практических задач средствами языка программирования.

Итоговый контроль осуществляется в конце учебного года.

Форма контроля: защита творческого проекта.

Методические материалы

В процессе обучения в тесной взаимосвязи реализуются такие *методы обучения*:

- словесные;
- наглядные;
- практические;
- проблемно-поисковые;
- индуктивные.

Материально-технические условия реализации программы:

- реализация программы предполагает наличие учебных кабинетов с автоматизированными рабочими местами для учащихся.

Оборудование учебного кабинета:

- комплект мебели для учащихся;
- комплект мебели для преподавателя.

Технические средства обучения:

- ноутбуки с предустановленным программным обеспечением (Интерпретатор Python версии 3.7 и выше; IDE JetBrainsPyCharm);
- доступ к сети Интернет;
- мультимедийный проектор.