

муниципальное бюджетное общеобразовательное учреждение
«Балезинская средняя школа № 3»
МБОУ «Балезинская средняя школа № 3»

ПРИНЯТО
На заседании педагогического совета
Протокол № 1 от 28.08.2024г.

УТВЕРЖДАЮ
Директор
Максимова И.В.

Приказ № 158-ОД от 30.08.24 г



Дополнительная образовательная программа
Технической направленности
«Программирование на языке Python»

Для обучающихся 16-18 лет класс

Срок реализации 1 год

Составитель: Максимова И.В.

учитель математики, информатики

п. Балезино, 2024

Пояснительная записка

Дополнительная образовательная программа «Программирование на языке Python» составлена на основе курса Д.П. Кириенко «Основы языка программирования Python», М.: Бином, 2014 г., Примерной программы внеурочной деятельности начального и основного образования.

Программа предназначена для изучения программирования в 10 – 11 классах средней школы на базовом уровне. Дополнительная общеобразовательная программа «Программирование на языке Python» имеет техническую направленность; по функциональному предназначению – учебно-познавательной; по форме организации – групповая, индивидуально-групповая; по времени реализации – одногодичной.

Информатика — это естественнонаучная дисциплина о закономерностях протекания информационных процессов в системах различной природы, а также о методах и средствах их автоматизации.

Многие положения, развиваемые информатикой, рассматриваются как основа создания и использования информационных и коммуникационных технологий — одного из наиболее значимых технологических достижений современной цивилизации. Вместе с математикой, физикой, химией, биологией изучение информатики закладывает основы естественнонаучного мировоззрения.

Информатика имеет большое и всевозрастающее число междисциплинарных связей, причем как на уровне понятийного аппарата, так и на уровне инструментария. Многие предметные знания и способы деятельности (включая использование средств ИКТ), освоенные обучающимися на базе информатики, находят применение как в рамках образовательного процесса при изучении других предметных областей, так и в иных жизненных ситуациях. Они становятся значимыми для формирования качеств личности, т. е. ориентированы на формирование метапредметных и личностных результатов. На протяжении всего периода становления школьной информатики в ней накапливался опыт формирования образовательных результатов, которые в настоящее время принято называть современными образовательными результатами.

Одной из основных черт нашего времени является всевозрастающая изменчивость окружающего мира. В этих условиях велика роль фундаментального образования, обеспечивающего профессиональную мобильность человека, готовность его к освоению новых технологий, в том числе информационных. Необходимость подготовки личности к быстро наступающим переменам в обществе требует развития разнообразных форм мышления, формирования у учащихся умений организации собственной учебной деятельности, их ориентации на деятельностную жизненную позицию.

Автоматизированные и компьютерные производства, новые информационные технологии, занявшие устойчивые позиции на современных предприятиях и организациях, предъявляют высокие требования к ИКТ–компетентности выпускников. Каждая сфера деятельности человека: медицина, проектирование зданий, машин, образование, — не обходится без применения компьютерных технологий в современном мире. Соответственно, навыки программирования пользуются высоким спросом.

В рамках предлагаемой программы «Программирование на языке Python» 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. Исключительно велика роль программирования для формирования мышления школьников, приёмов умственных действий, умения строить модели, самостоятельного нахождения и составления алгоритмов решения задач, умения чётко и лаконично реализовывать этапы решения задач. Использование этих возможностей для формирования общеинтеллектуальных и общеучебных умений школьников активизирует процесс индивидуально-личностного становления учащихся. Общепедагогическая направленность занятий – гармонизация индивидуальных и социальных аспектов обучения по отношению к информационным технологиям. Умение составлять алгоритмы решения и навыки программирования являются элементами информационной компетенции — одной из ключевых компетенций современной школы. Умение находить решение, составлять алгоритм решения и реализовать его с помощью языков программирования — необходимое условие подготовки современных школьников.

Актуальность программы

Программа посвящена одному из бурно развивающихся и популярных в настоящее время языков программирования – Python. Python имеет богатую стандартную библиотеку и большое количество модулей расширения практически для всех нужд отрасли информационных технологий. Синтаксис языка понятен и не вызывает значительных затруднений в изучении. Язык Python может быть использован при решении заданий в формате ОГЭ, ЕГЭ в части программирования.

Новизна и отличительные особенности программы состоят в том, что реализуется возможность обучению навыкам работы в группе, создания коллективных проектов, чего практически невозможно достичь при изучении традиционных языков Бейсик и Паскаль. Возможность увидеть результаты своего труда в Интернет также стимулирует интерес детей получить представление об алгоритмах и исполнителях, основных алгоритмических конструкциях языков программирования.

Отличительной особенностью данной дополнительной образовательной программы от уже существующих образовательных программ является тот факт, что, программа разработана как самостоятельная дисциплина, являющаяся образовательным компонентом общего среднего образования. Вместе с тем она пронизывает содержание многих других предметов и, следовательно, становится дисциплиной обобщающего, методологического плана. Образовательная программа «Программирование на языке Python» является расширением предмета «Основы программирования».

Предлагаемая программа «Программирование на языке Python» предназначена для организации внеурочной деятельности по трем взаимосвязанным направлениям развития личности: общеинтеллектуальное, общекультурное и социальное.

Цель: получить навыки программирования на языке Python.

В ходе реализации программы дополнительного образования решаются следующие **задачи:**

Воспитательные:

- воспитывать ответственность за результаты учебного труда, понимание его значимости, соблюдение техники безопасности, санитарно-гигиенических условий труда;
- влиять на профессиональное самоопределение;
- воспитание воли, умения преодолевать трудности, познавательной активности и самостоятельности, настойчивости;

- воспитание интереса к предмету, к учению.

Обучающие:

- изучить структуру программы, основные операторы языка Python;
- получить навыки работы в среде программирования.

Развивающие:

- развивать учебно-познавательные компетенции в процессе тренировки навыков, решения задач, касающихся написания программ различными методами;
- развивать компетенции самоорганизации в процессе выработки и тренировки наиболее эффективной стратегии выполнения тестовых заданий.

Планируемые результаты изучения учебного предмета

Личностные результаты

- сформированность мировоззрения, соответствующего современному уровню развития науки и техники;
- готовность и способность к образованию, в том числе самообразованию, на протяжении всей жизни; сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности;
- навыки сотрудничества со сверстниками, детьми младшего возраста, взрослыми в образовательной, учебно-исследовательской, проектной и других видах деятельности;
- эстетическое отношение к миру, включая эстетику научного и технического творчества;
- осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов; отношение к профессиональной деятельности как возможности участия в решении личных, общественных, государственных, общенациональных проблем.

Метапредметные результаты

- 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 результаты

- 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.

Состав учащихся: программа адресована детям (подросткам, мальчикам и девочкам) с 15-17 лет без специальной подготовки.

Условия набора детей в коллектив: принимаются все желающие (не имеющие медицинских противопоказаний).

Наполняемость в группах составляет: от 10 человек.

Сроки реализации образовательной программы: 2 года.

Режим занятий: занятия проводятся: 2 раза в неделю по 1 часу (итого 68 часов в год).

Особенности организации учебного процесса: занятия проводятся в группах учащихся одного возраста, являющихся основным составом объединения, а также индивидуально. Состав группы – постоянный. Система работы программы включает в себя теоретические и практические занятия, ориентирована на большой объем практических творческих работ с использованием компьютера. Все образовательные модули предусматривают не только усвоение теоретических знаний, но и формирование деятельностно-практического опыта. Освоение материала в основном происходит в процессе практической творческой деятельности.

Формы занятий. Важным условием реализации вышеназванных задач является метод учебно-исследовательских проектов, основанный на исследовательской деятельности обучающихся.

При проведении занятий традиционно используются три формы работы:

- демонстрационная, когда обучающиеся слушают объяснения педагога и наблюдают за демонстрационным экраном или экранами компьютеров на ученических рабочих местах;
- фронтальная, когда обучающиеся синхронно работают под управлением педагога;
- самостоятельная, когда обучающиеся выполняют индивидуальные задания в течение части занятия или нескольких занятий.

Технологии и формы обучения:

- теоретические занятия;
- учебно-исследовательские проекты;
- свободное творчество.

Методы обучения

Отбор методов обучения обусловлен необходимостью формировать информационную и коммуникативную компетентности учащихся, реализовывать личностно-ориентированное обучение, направлять их на самостоятельное решение разнообразных проблем, развивать исследовательские и творческие способности. Решение данных задач кроется в организации деятельностного подхода к обучению, в проблемном изложении материала учителем, в переходе от репродуктивного вида работ к самостоятельным, поисково-исследовательским видам деятельности. Поэтому основная методическая установка в данной программе — обучение учащихся навыкам самостоятельной творческой деятельности.

Содержание программы

Тема 1. Знакомство с языком Python

Общие сведения о языке Python. Установка Python на компьютер. Режимы работы Python. Что такое программа. Первая программа. Структура программы на языке Python. Комментарии.

Учащиеся должны знать/понимать:

- понятие программы;
- структуру программы на Python;
- режимы работы с Python.

Учащиеся должны уметь:

- выполнить установку программы;
- выполнить простейшую программу в интерактивной среде;
- написать комментарии в программе.

Виды деятельности: познавательная; практическая.

Формы деятельности: беседа, демонстрация мультимедийных презентаций, работа в творческих группах.

Тема 2. Переменные и выражения

Типы данных. Преобразование типов. 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. Пример скрипта, использующего ввод и вывод данных. Задачи на элементарные действия с числами. Решение задач на элементарные действия с числами.

Учащиеся должны знать/понимать:

- общую структуру программы;
- типы данных;
- целые, вещественные типы данных и операции над ними;
- оператор присваивания;
- операторы ввода-вывода.

Учащиеся должны уметь:

- пользоваться интерфейсом среды программирования Python;
- использовать команды редактора;
- организовывать ввод и вывод данных;
- записывать арифметические выражения.

Виды деятельности: познавательная; практическая.

Формы деятельности: беседа, демонстрация мультимедийных презентаций, работа в творческих группах.

Тема 3. Условные выражения

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.

Учащиеся должны знать/понимать:

- назначение условного оператора;
- способ записи условного оператора;
- логический тип данных;
- логические операторы or, and, not; Учащиеся должны уметь:
- использовать условный оператор;
- создавать сложные условия с помощью логических операторов.

Виды деятельности: познавательная; практическая.

Формы деятельности: беседа, групповая проблемная работа, демонстрация мультимедийных презентаций, подготовка проекта, работа в творческих группах.

Тема 4. Циклы

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.

Учащиеся должны знать/понимать:

- циклы с условием и их виды;
- правила записи циклов с условием;
- назначение и особенности использования цикла с параметром;
- формат записи цикла с параметром;
- примеры использования циклов различных типов.

Учащиеся должны уметь:

- определять вид цикла, наиболее удобный для решения поставленной задачи;
- использовать цикл с условием;
- определять целесообразность применения и использовать цикл с параметром для решения поставленной задачи;

Виды деятельности: познавательная; практическая.

Формы деятельности: беседа, групповая проблемная работа, демонстрация мультимедийных презентаций, подготовка проекта, работа в творческих группах.

Тема 5. 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. Числа Фибоначчи.

Учащиеся должны знать/понимать:

- понятие функции;
- способы описания функции;
- принципы структурного программирования;
- понятие локальных переменных подпрограмм;
- понятие формальных и фактических параметров подпрограмм;
- способ передачи параметров.

Учащиеся должны уметь:

- создавать и использовать функции;
- использовать механизм параметров для передачи значений.

Тема 6. 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.

Учащиеся должны знать/понимать:

- назначение строкового типа данных;
- операторы для работы со строками;
- процедуры и функции для работы со строками;
- операции со строками.

Учащиеся должны уметь:

- описывать строки;
- соединять строки;
- находить длину строки;
- вырезать часть строки;
- находить подстроку в строке;
- находить количество слов в строке.

Виды деятельности: познавательная; практическая.

Формы деятельности: беседа, групповая проблемная работа, демонстрация мультимедийных презентаций, подготовка проекта, работа в творческих группах.

Тема 7. Сложные типы данных

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.

Матрицы. Вложенные списки. Матрицы. Строки и списки. Генераторы списков в Python.

Кортежи. Присваивание кортежей. Кортежи как возвращаемые значения

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

Учащиеся должны знать/понимать:

- сложные типы данных;
- способ описания списка;
- способ доступа к элементам списка;
- способ описания кортежа;
- способ описания словаря;
- операции, выполняемые со списками, кортежами и словарями;
- понятие множества;
- способы описания множества;
- операторы работы с множествами.

Учащиеся должны уметь:

- описывать списки;
- вводить элементы списка;
- выводить элементы списка;
- выполнять поиск элемента в списке, поиск минимума и максимума, нахождение суммы элементов списка;
- использовать вложенные списки;
- приводить примеры использования вложенных списков (матриц);
- описывать множества;
- определять принадлежность элемента множеству;
- вводить элементы множества;
- выводить элементы множества.

Виды деятельности: познавательная; практическая.

Формы деятельности: беседа, групповая проблемная работа, демонстрация мультимедийных презентаций, подготовка проекта, работа в творческих группах.

Тема 8. Стиль программирования и отладка программ.

Стиль программирования. Отладка программ. Зачет по курсу «Программирование на языке Python»

Учащиеся должны знать/понимать:

- что такое стиль программирования;
- правила именования объектов;

- основные рекомендации при написании программ.

Учащиеся должны уметь:

- определять вид ошибок и находить ошибки в программе;
- выполнять тестирование и отладку программ.

Виды деятельности: познавательная; практическая.

Формы деятельности: беседа, групповая проблемная работа, самостоятельная работа, демонстрация мультимедийных презентаций, подготовка проекта, работа в творческих группах.

Учебно-тематический план

№	Наименование разделов или тем	Общее количество часов	В том числе	
			теоретических	практических
1	Знакомство с языком Python и его установка	4	2	2
2	Переменные и выражения	4	1	3
3	Условные выражения	8	2	6
4	Циклы	16	4	12
5	Функции	10		10
6	Строки – последовательности символов	6	1	5
7	Сложные типы данных	14	4	10
8	Стиль программирования и отладка программ	6		6
Итого часов		68	14	54

Календарно-тематическое планирование

Используемые сокращения: СР – самостоятельная работа, ПР – практическая работа, ЗП – защита проекта.

№ урока	Наименование тем	Количество часов	Деятельность на уроке
	Тема 1. Знакомство с языком Python	4	
1	Общие сведения о языке	2	ПР 1.1. «Установка программы Python»
2	Режимы работы	2	ПР 1.2. «Режимы работы с Python». Тест № 1. «Знакомство с языком Python»
	Тема 2. Переменные и выражения	4	
3	Переменные	1	ПР 2.1. «Работа со справочной системой». ПР 2.2. «Переменные»
4	Выражения	1	ПР 2.3. «Выражения»
5	Ввод и вывод	1	

6	Задачи на элементарные действия с числами	1	ПР 2.5. «Задачи на элементарные действия с числами». Тест № 2. «Выражения и операции»
---	---	---	--

	Тема 3. Условные выражения	8	
7	Логические выражения и операторы	1	ПР 3.1. «Логические выражения»
8	Условный оператор	1	ПР 3.2. «Условный оператор»
9	Множественное ветвление	1	ПР 3.3. «Множественное ветвление»
10	Реализация ветвления в языке Python.	3	ПР 3.4. «Условные операторы» СР № 1 по теме «Условные операторы»
11	Зачетная работа № 1. «Составление программ с ветвлением»	2	Тест № 3. «Условные операторы»
	Тема 4. Циклы	16	
12	Оператор цикла с условием	2	ПР 4.1. «Числа Фибоначчи»
13	Оператор цикла for	2	ПР 4.2. «Решение задачи с циклом for»
14	Вложенные циклы	4	ПР 4.3. «Реализация циклических алгоритмов»
15	Случайные числа	2	ПР 4.4. «Случайные числа»
16	Примеры решения задач с циклом	2	ПР 4.5. «Решение задач с циклом». СР № 2 «Составление программ с циклом». Тест № 4. «Циклы»
17	Творческая работа № 1. "Циклы"	4	
	Тема 5. Функции	10	
18	Создание функций	2	ПР 5.1. «Создание функций»
19	Локальные переменные	2	ПР 5.2. «Локальные переменные»
20	Примеры решения задач с использованием функций	2	ПР 5.3. «Решение задач с использованием функций». СР № 3 по теме «Функции»
21	Рекурсивные функции	4	ПР 5.4. «Рекурсивные функции». Тест № 5. «Функции»
	Тема 6. Строки - последовательности символов	6	
22	Строки	2	
23	Срезы строк	2	ПР 6.1. «Строки»
24	Примеры решения задач со строками	2	ПР 6.2. «Решение задач со строками» Тест № 6 «Строки»
	Тема 7. Сложные типы данных	14	
25	Списки	1	
26	Срезы списков	1	ПР 7.1. «Списки»
27	Списки: примеры решения задач	2	ПР 7.2. «Решение задач со списками»
28	Матрицы	4	Тест № 7. «Списки»
31	Множества в языке Python	6	

	Тема 8. Стил ь программирования и отладка программ	6	
32	Стил ь программирования	2	ПР 8.1. «Работа над проектом»
33	Отладка программ	2	ПР 8.2. «Отладка программ»
34	Зачет по курсу «Программирование на языке Python»	2	ЗП
	Всего	68	

Материально-техническое обеспечение

Реализация программы предполагает наличие учебных кабинетов: компьютерный класс.

Оборудование компьютерного класса:

- рабочие места по количеству обучающихся, оснащенные персональными компьютерами или ноутбуками с установленным лицензионным программным обеспечением;
- рабочее место преподавателя, оснащенное персональным компьютером или ноутбуком с установленным лицензионным программным обеспечением;
- комплект учебно-методической документации: рабочая программа кружка, раздаточный материал, задания,
- цифровые компоненты учебно-методических комплексов (презентации).

Технические средства обучения: демонстрационный комплекс, включающий в себя: интерактивную доску (или экран), мультимедиапроектор, персональный компьютер или ноутбук с установленным лицензионным программным обеспечением.

Обязательно наличие локальной сети и доступа к сети Интернет.

Требования к комплектации компьютерного класса

Наиболее рациональным с точки зрения организации деятельности детей в школе является установка в компьютерном классе 13–15 компьютеров (рабочих мест) для школьников и одного компьютера (рабочего места) для педагога.

Предполагается объединение компьютеров в локальную сеть с возможностью выхода в Интернет, что позволяет использовать сетевые цифровые образовательные ресурсы.

Минимальные требования к техническим характеристикам каждого компьютера следующие:

- процессор – не ниже *Celeron* с тактовой частотой 2 ГГц;
- оперативная память – не менее 256 Мб;
- жидкокристаллический монитор с диагональю не менее 15 дюймов;
- жёсткий диск – не менее 80 Гб;
- клавиатура;
- мышь;
- устройство для чтения компакт-дисков (желательно);
- аудио-карта и акустическая система (наушники или колонки).

Кроме того, в кабинете информатики должны быть:

- принтер на рабочем месте учителя;
- интерактивная система на рабочем месте учителя;
- сканер на рабочем месте учителя.

Требования к программному обеспечению компьютеров

На компьютерах, которые расположены в кабинете информатики, должна быть установлена операционная система *Windows* или *Linux*, а также необходимое программное обеспечение:

- текстовый редактор (*Блокнот* или *Gedit*) и текстовый процессор (*MS Word* или *OpenOffice Writer*);
- табличный процессор (*MS Excel* или *OpenOffice Calc*);
- средства для работы с базами данных (*MS Access* или *OpenOffice Base*);
- среда программирования Python 3+, интегрированная среда разработки.

Информационное обеспечение обучения

1. Домашняя страница Python www.python.org. Справочные материалы, официальная документация.
2. Дистанционная подготовка. Курс «Основы языка программирования Python: легко и просто» на python «Д.П. Кириенко. «Основы языка программирования Python: легко и просто» на языке Python (школа 179 г. Москва)» (<http://informatics.msk.ru/course/view.php?id=156>)
3. Сайт проекта Интуит: Национальный открытый университет, курс «Введение в программирование на Python», <http://www.intuit.ru/studies/courses/12179/1172/info>.
4. Сайт проекта Интуит: Национальный открытый университет. Курс «Язык программирования Python» <http://www.intuit.ru/studies/courses/49/49/info>.
5. Сайт проекта Open Book Project openbookproject.net содержит серию практических примеров на Python Криса Мейерса.
6. Python. Подробный справочник Дэвида М. Бизли — книга со справочной информацией о языке Python и модулях стандартной библиотеки.
7. Python. Справочник Марка Лутца. Справочник по наиболее часто используемым функциям и модулям.
8. Компьютерный практикум в электронном виде с комплектом электронных учебных средств, размещённый на сайте: <http://kpolyakov.spb.ru/school/probook.htm>
9. Материалы для подготовки к итоговой аттестации по информатике в форме ОГЭ, размещённые на сайте <http://kpolyakov.spb.ru/school/oge.htm>

Литература, рекомендованная для учащихся

1. К.Ю. Поляков, Е. А. Еремин. «Информатика», углубленный уровень, М.: Бином. Лаборатория Знаний, 2013.
2. М. Лутц «Изучаем Питон», Санкт-Петербург: Символ, 2011 г.
3. Задачник-практикум по информатике: Учебное пособие для средней школы/Под ред. И. Г. Семакина, Е. К. Хеннера. — М.: БИНОМ. Лаборатория