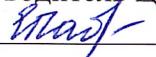


Муниципальное общеобразовательное учреждение
«Средняя общеобразовательная школа №14 города Пугачева»

<p>«Согласовано» Педагог дополнительного образования МОУ «СОШ №14 города Пугачёва имени П.А. Столыпина», руководитель Центра «Точка роста»  Е. Ю. Павлова «27» августа 2024 года</p>	<p>«Утверждено» Директор МОУ «СОШ №14 города Пугачёва имени П.А. Столыпина»  И.В. Саленко Приказ № 222 «30» августа 2024 года</p> 
---	--

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА**
технической направленности
**«Программирование на языке Python», реализуемая в
Центре образования естественно-научной и технологической
направленностей «Точка роста»**

Возраст учащихся: 13-18 лет
Срок реализации программы: 9 месяцев

Автор (составитель) программы:
Маркина Оксана Александровна, педагог
дополнительного образования

Рассмотрено на заседании педагогического совета
Протокол № 11 от 27.08.2024 года

г. Пугачев, 2024 г.

АННОТАЦИЯ

Дополнительная общеразвивающая общеобразовательная программа «Программирование на языке Python» ориентирована на развитие у учащихся технического мышления, составлена для организации внеурочной деятельности учащихся основной школы и ориентирована на обучающихся, проявляющих интересы и склонности в области информатики. Данный курс «Программирование на языке Python» рассчитан на 136 часов. Основные компоненты программы:

- информативный блок (интерактивные лекции, презентации, видеоролики);
- блок практических заданий в среде программирования, выполнение которых предполагает наработку умений и навыков в освоении изученных технологий;
- блок творческих проектов, тематика которых определяется по желанию учащихся;
- диагностический блок (тесты, опросы, выставки роботов).

Возраст учащихся – 13-18 лет.

Срок реализации – 9 месяцев.

РАЗДЕЛ I

КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ПРОГРАММЫ ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеразвивающая общеобразовательная программа «Практическая информатика» является программой технической направленности, так как ориентирована на развитие способностей учащихся в сфере информационных технологий и инженерно-технического творчества, формирование навыков научно-исследовательской и конструкторской деятельности. Освоение данной программы способствует развитию интереса детей к технике как объекту творчества, активизирует стремление к самостоятельному познанию и техническому конструированию, способствует осознанному выбору профессии в области техники и технологий. Ключевой особенностью курса является его направленность на формирование у учащихся навыков поиска собственного решения поставленной задачи, составления алгоритма решения и его реализации с помощью средств программирования.

Актуальность программы обусловлена широким распространением информационно-коммуникационных технологий в обществе и необходимостью обеспечивать связанную с этим инфраструктуру специалистами. Всё большее значение приобретает умение человека грамотно обращаться с компьютером, причём на уровне начинающего программиста.

Изучение основных принципов программирования невозможно без регулярной практики написания программ на каком-либо языке программирования. Для обучения в рамках программы выбран язык Python, который является достаточно эффективным и доступным инструментом достижения задач в области создания программных продуктов.

Синтаксис языка достаточно прост и интуитивно понятен, что понижает порог вхождения и позволяет сосредоточиться на алгоритмических аспектах программирования.

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.

Педагогическая целесообразность заключается в том, что данная программа позволит выявить заинтересованных обучающихся, проявивших интерес к знаниям, оказать им помощь в формировании устойчивого

интереса – 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. Особая роль отводится широко представленной в курсе системе рефлексивных заданий. Освоение рефлексии направлено на осознание учащимися того важного обстоятельства, что наряду с разрабатываемыми ими продуктами в виде программ на компьютере рождается основополагающий образовательный продукт: освоенный инструментарий. Именно этот образовательный продукт станет базой для творческого самовыражения учащихся в форме различных программ.

Адресат программы. Дополнительная образовательная общеразвивающая программа разработана для учащихся в возрасте от 13 до 18 лет.

Практическая значимость заключается в том, что обучающиеся получают теоретические знания и практические навыки, которые позволят профессионально использовать компьютер, обобщение и передачу различной информации, ознакомятся языком программирования Python.

Цель программы: понять значение алгоритмизации как метода познания окружающего мира, принципы структурной алгоритмизации; овладеть базовыми понятиями теории алгоритмов.

Задачи программы:

Обучающие задачи программы:

- познакомить с понятиями алгоритма, вычислимой функции, языка программирования;
- научить составлять и читать блок-схемы.

Развивающие задачи программы:

- развитие абстрактного и логического мышления;
- развитие творческого подхода к решению различных задач.

Воспитательные задачи программы:

- воспитание самостоятельности;
- воспитание культуры взаимодействия с другими людьми в условиях открытого информационного общества.

Дополнительная общеразвивающая общеобразовательная программа «Программирование на языке Python» направлена на достижение целого комплекса образовательных результатов:

Предметные:

- ✓ познакомить с основными алгоритмическими конструкциями и правилами их записи, с основными способами организации данных.
- ✓ научить составлять и записывать алгоритмы с использованием соответствующих алгоритмических конструкций.
- ✓ научить распознавать необходимость применения той или иной алгоритмической конструкции при решении задач
- ✓ научить организовывать данные для эффективной алгоритмической обработки.
- ✓ научить разрабатывать алгоритмы и реализовывать их на языке программирования Python.
- ✓ научить осуществлять отладку и тестирование программы.

Личностные:

- ✓ повышение общекультурного уровня;
- ✓ вооружение правильным методологическим подходом к познавательной и практической деятельности;
- ✓ выделение и раскрытие роли информационных технологий и компьютеров в развитии современного общества;
- ✓ привитие навыков сознательного и рационального использования компьютера в своей учебной, а затем и профессиональной деятельности;
- ✓ воспитание стремления к овладению техникой исследования;
- ✓ воспитание трудолюбия, инициативности и настойчивости в преодолении трудностей.

Метапредметные:

- ✓ владение умениями организации собственной учебной деятельности, включающими: целеполагание как постановку учебной задачи на основе соотнесения того, что уже известно, и того, что требуется установить;
- ✓ планирование – определение последовательности промежуточных целей с учетом конечного результата, разбиение задачи на подзадачи, разработка последовательности и структуры действий, необходимых для достижения цели при помощи фиксированного набора средств;
- ✓ прогнозирование – предвосхищение результата.

Основные формы и методы организации занятий.

Формы проведения занятий: учебное занятие, выполнение и защита проекта, практические работы, лекции, выставки работ, презентации работ.

Методы проведения занятия: словесные, наглядные, практические, их сочетание. Каждое занятие по темам программы включает теоретическую часть и практическое выполнение задания.

Дополнительная общеразвивающая общеобразовательная программа «Программирование на языке Python» реализуется в центре образования естественно-научной и технологической направленностей «Точка роста».

Объем программы: Количество часов, необходимое для реализации программы – 136.

Срок освоения: 9 месяцев.

Режим занятий: 2 раза в неделю по 2 часа.

СОДЕРЖАНИЕ ПРОГРАММЫ
Учебный план

№ п/п	Наименование раздела, темы	Количество часов			Формы аттестации/контроля
		Всего	Теория	Практика	
1.	Раздел 1. Введение в Python	8	5	3	Тест
2.	Раздел 2. Типы данных и операции	22	11	11	Тест
3.	Раздел 3. Инструкции и синтаксис	18	9	9	Тест
4.	Раздел 4. Функции и модули в программировании	30	10	20	Тест
5.	Раздел 5. Сложные типы данных	28	10	18	Тест
	Раздел 6. Написание и отладка программ. Самостоятельная работа	30	4	26	Защита проекта
	Итого	136	49	87	

Содержание тем 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. 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».

КАЛЕНДАРНО-ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

№ п/п	Наименование темы занятия	Форма проведения занятия	Всего, часов	Дата	
				План	Факт
Раздел 1. Введение в Python – 8 ч.					
1	Техника безопасности. История создания. Области применения и перспективы. Инструкции и структура программы.	Беседа; презентация	1		
2	Основы Алгоритмизации. Достоинства языка. Установка программы Python.	Работа с компьютером	2		
3	Основы ввода и вывода данных. Первая программа на языке Python.	Работа с компьютером	2		
4	Решение задач для развития логики и понимания работы алгоритма.	Работа с компьютером	2		
5	Написание первой программы вывод на экран.	Работа с компьютером	1		
Раздел 2. Типы данных и операции – 22 ч.					
6	Переменные. Оператор присваивания. Имена переменных и ключевые слова. Выражения. Операции.	Работа с компьютером	1		
7	Порядок выполнения операций. Математические функции. Ввод и вывод данных на экран.	Работа с компьютером	2		
8	Решение задач на элементарные действия с числами.	Работа с компьютером	2		

9	Создание программы простейший калькулятор.	Работа с компьютером	2		
10	Типы данных. Преобразование типов. Различия типов данных. Базовые функции (abs, round, int, math)	Работа с компьютером	2		
11	Решение математических задач с использованием функции import math.	Работа с компьютером	2		
Раздел 3. Инструкции и синтаксис -18 ч.					
12	Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or, not).	Работа с компьютером	2		
13	Условный оператор. Альтернативное выполнение. Примеры решения задач с условным оператором	Работа с компьютером	2		
14	Множественное ветвление. Реализация ветвления в языке Python.	Работа с компьютером	2		
15	Практическое закрепление знаний по условным операторам. Создание программ.	Работа с компьютером	2		
16	Понятие цикла. Тело цикла. Условия выполнения тела цикла. Оператор цикла с условием. Оператор цикла while. Бесконечные циклы. Альтернативная ветка цикла while.	Работа с компьютером	2		
17	Пример задачи с использованием цикла for. Вложенные циклы.	Работа с компьютером	2		

18	Циклы в циклах. Случайные числа. Функция randrange. Функция random. Примеры решения задач с циклом.	Работа с компьютером	2		
19	Числа Фибоначчи. Решение задачи с циклом for. Создание игры угадай число.	Работа с компьютером	2		
20	Решение задач на пройденный материал.	Работа с компьютером	2		
Раздел 4. Функции и модули в программировании – 30 ч.					
21	Создание функций. Параметры и аргументы. Локальные и глобальные переменные. Поток выполнения. Функции, возвращающие результат.	Работа с компьютером	3		
22	Анонимные функции, инструкция lambda. Рекурсивные функции. Вычисление факториала.	Работа с компьютером	3		
23	Примеры решения задач с использованием функций.	Работа с компьютером	4		
24	Составной тип данных - строка. Доступ по индексу. Длина строки и отрицательные индексы. Преобразование типов. Операторы для всех типов последовательностей (строки, списки, кортежи).	Работа с компьютером	3		
25	Применение цикла для обхода строки. Срезы строк. Строки нельзя изменить. Сравнение строк. Оператор in. Модуль string.	Работа с компьютером	3		

26	Примеры решения задач со строками.	Работа с компьютером	4		
27	Списки. Тип список (list). Индексы. Обход списка. Проверка вхождения в список. Добавление в список.	Работа с компьютером	2		
28	Суммирование или изменение списка. Операторы для списков. Срезы списков. Удаление списка. Клонирование списков. Списочные параметры.	Работа с компьютером	2		
29	Функция range. Списки: примеры решения задач.	Работа с компьютером	2		
30	Написание программ. Работа со списками.	Работа с компьютером	4		
Раздел 5. Сложные типы данных – 28 ч.					
31	Списки. Тип список (list). Индексы. Обход списка. Проверка вхождения в список. Добавление в список. Суммирование или изменение списка. Операторы для списков.	Работа с компьютером	2		
32	Срезы списков. Удаление списка. Клонирование списков. Списочные параметры. Функция range. Списки: примеры решения задач.	Работа с компьютером	2		
33	Написание программ. Работа со списками.	Работа с компьютером	2		
34	Матрицы. Вложенные списки. Матрицы. Строки и списки. Генераторы списков в Python.	Работа с компьютером	2		

35	Установка Python. Доступ к документации. Ввод и вывод данных. Первая программа на языке Python.	Работа с компьютером	2		
36	Создание программ. Практическое закрепление полученных знаний.	Работа с компьютером	2		
37	Кортежи. Присваивание кортежей. Кортежи как возвращаемые значения.	Работа с компьютером	2		
38	Создание программ. Практическое закрепление полученных знаний	Работа с компьютером	2		
39	Введение в словари. Тип словарь (dict). Словарные операции. Словарные методы.	Работа с компьютером	2		
40	Создание программ. Практическое закрепление полученных знаний.	Работа с компьютером	2		
41	Множества в языке Python. Множества. Множественный тип данных. Описание множеств.	Работа с компьютером	2		
42	Операции, допустимые над множествами: объединение, пересечение, разность, включение.	Работа с компьютером	2		
43	Оператор определения принадлежности элемента множеству.	Работа с компьютером	2		
44	Создание программ. Практическое закрепление полученных знаний.	Работа с компьютером	2		

Раздел 6. Написание и отладка программ. Самостоятельная работа – 30 ч.

45	Решение типовых задач. Условный оператор	Работа с компьютером	4		
46	Решение типовых задач. Множественное ветвление	Работа с компьютером	4		
47	Решение типовых задач. Циклы	Работа с компьютером	4		
48	Решение типовых задач. Функции	Работа с компьютером	4		
49	Решение типовых задач. Строки	Работа с компьютером	4		
50	Стиль программирования. Отладка программ.	Работа с компьютером	4		
51	Проектная работа по курсу «Программирование на языке Python».	Работа с компьютером	6		

РАЗДЕЛ II КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ

МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

Программа дополнительного образования «Программирование на языке Python» разработана с использованием разнообразных методов и приемов обучения. Методы и формы, используемые педагогом, отражают его организующую, обучающую, контролирующую функции и обеспечивают учащимся возможность ознакомления, тренировки и применения учебного материала.

Программа предполагает использование следующих образовательных технологий: развивающего обучения, личностно ориентированного обучения, моделирующего обучения, здоровьесберегающих технологий и технологий ИКТ.

Формы проведения занятий:

- лекции;
- практические задания на компьютере;
- выполнение проектов;
- выставки работ.

Методические материалы:

- инструкции по ТБ;
- методические разработки занятий;
- презентации;
- демонстрационный материал;
- дидактический материал;
- инструкции к проектам.

УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

Материально-техническое обеспечение программы

Программа реализуется на базе центра образования естественно-научной и технологической направленности «Точка роста». Занятия проводятся в учебном кабинете, который оснащен необходимым оборудованием: классная доска, столы и стулья для учащихся и педагога, шкафы и стеллажи для хранения образовательных конструкторов и учебных материалов, ноутбуки для работы в среде программирования. Учебный кабинет оформлен в соответствии с профилем проводимых занятий и оборудован в соответствии с санитарными нормами.

Перечень технических средств обучения: ноутбуки, компьютеры, принтер, мультимедиапроектор.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

В процессе выполнения работы по изготовлению моделей используется текущий контроль. Педагог непрерывно отслеживает процесс работы учащихся, своевременно направляет учащихся на исправление неточностей в практической работе. Текущий контроль позволяет в случае необходимости вовремя произвести корректировку деятельности и не испортить изделие.

Формы текущего контроля: опрос, демонстрация изделий, презентация роботов.

В конце учебного года проводится защита итогового проекта.

Формы проведения итогового контроля: итоговый проект, выставка работ.

Планируемые результаты, в соответствии с целью программы, отслеживаются, фиксируются и демонстрируются в формах: готовая работа, журнал посещаемости, фото, выставки, демонстрация моделей.

Оценочные материалы: опрос, педагогическое наблюдение, творческая работа, выставка готовых работ.

ЛИТЕРАТУРА

- 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.
- 4) Луц М. Изучаем Python. СПб: Симво-плюс, 2011. - 1280 с.
- 5) Паронджанов В.Д. Учись писать, читать и понимать алгоритмы. М: Ямб, 2012. – 520 с.
- 6) Златопольский Д.М. Сборник задач по программированию. СПб: БХВ-Петербург, 2011. - 295 с.
- 7) ООП на Python: концепции, принципы и примеры реализации [Электронный ресурс]. – Режим доступа: <https://proglib.io/p/python-oop/>
- 8) Свейгар.Эл. Учим python, делая крутые игры М: Эксмо, 2018. – 416 с.
- 9) Семакин И.Г. Основы алгоритмизации и программирования. М.: Академия, 2016. - 304 с.
- 10) Уроки по Python для начинающих [Электронный ресурс]. – Режим доступа: <https://pythonru.com/uroki/vvedenie-uroki-po-python-dlja-nachinajushhih>
- 11) Алгоритмизация. Программирование Python 3 [Электронный ресурс]. – Режим доступа: <https://stepik.org/course/7215/promo>
- 12) <https://www.videosoftdev.com/ru/how-to-use-free-video-editor>
- 13) Python. Подробный справочник Дэвида М. Бизли — книга со справочной информацией о языке Python и модулях стандартной библиотеки.
- 14) Практический Python 3 для начинающих <https://pythonworld.ru/samouchitelpython>
- 15) Python для начинающих 2021 – уроки, задачи и тесты <https://pythonru.com/uroki/python-dlja-nachinajushhih>