

Муниципальное общеобразовательное учреждение
средняя общеобразовательная школа № 3 города Хвалынска
Саратовской области

Принята на заседании
педагогического совета
от «29» августа 20 23 г.
Протокол № 1
от 29 августа 2023

Утверждаю



Дополнительная общеобразовательная
общеразвивающая программа
технической направленности
«Программирование на языке Python»

Возраст обучающихся: 15-17 лет
Срок реализации – 1 год
Составитель программы – педагог
дополнительного образования
Захаров А.В.

Хвалынец 2023г

Комплекс основных характеристик дополнительной общеобразовательной общеразвивающей программы

Пояснительная записка

Программа «Программирование на языке «Python» направлена на подготовку творческой, технически грамотной, гармонично развитой личности, обладающей логическим мышлением, способной анализировать и решать задачи в команде в области информационных технологий, решать ситуационные кейсовые задания, основанные на групповых проектах. Занятия по данному курсу рассчитаны на общенаучную подготовку обучающихся, развитие их мышления, логики, математических способностей, исследовательских навыков. Программа направлена на изучение основ программирования на языке Python. В рамках реализации программы обучающиеся смогут познакомиться с физическими, техническими и математическими понятиями. Приобретённые знания будут применимы в творческих проектах.

Направленность программы: техническая.

Актуальность

Актуальность программы состоит в том, что она направлена на:

1. формирование целостного мировоззрения, соответствующего современному уровню развития науки и общественной практики за счет развития представлений об информации как важнейшем стратегическом ресурсе развития личности, государства, общества; понимания роли информационных процессов в современном мире;
2. совершенствование общеучебных и общекультурных навыков работы с информацией в процессе систематизации и обобщения имеющихся и получения новых знаний, умений и способов деятельности в области информатики и ИКТ; развитию навыков самостоятельной учебной деятельности школьников (учебного проектирования, моделирования, исследовательской деятельности и т.д.);
3. воспитание ответственного и избирательного отношения к информации с учётом правовых и этических аспектов ее распространения, воспитанию стремления к продолжению образования и созидательной деятельности с применением средств ИКТ.

Отличительная особенность:

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, Стандартов второго поколения в форме кружка в 9-10 классах.

Данная программа отличается от других программ, реализующих технологии на различных языках тем что, изучение основ программирования на языке Python — это не столько средство подготовки к будущей профессиональной деятельности, а сколько формирование новых общеинтеллектуальных умений и навыков: разделение задачи на этапы решения, построение алгоритма и др.

Педагогическая целесообразность:

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. 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 подростков.

Адресат программы: программа рассчитана на обучающихся 15-17 лет.

Возраст и возрастные особенности.

Обучающиеся 15-17 лет, участвующие в реализации программы, это уже подростки. На смену конкретному приходит логическое мышление. Это проявляется в критицизме и требовании доказательств. Для подростков характерно новое отношение к учению. Подросток стремится к самообразованию, причем часто становится равнодушным к оценке. Порой наблюдается расхождение между интеллектуальными возможностями и успехами в учебе: возможности высокие, а успехи низкие. Работая со старшеклассниками, проявившими интерес к робототехнике незадолго до окончания школы, приходится особенно бережно и тщательно относиться к их времени: создавать индивидуальные задания, больше внимания уделять самостоятельной работе. При работе используются различные приемы групповой деятельности в группах для обучения элементам кооперации, внесения в собственную деятельность самооценки, взаимно оценки, умение работать с технической литературой и выделять главное.

Программа разработана с учетом возрастных особенностей детей, их интересов и так, чтобы занятия были максимально интересными и познавательными.

Объем программы: 38 часов.

Срок реализации: 1 год.

Режим занятий: 1 час в неделю.

Наполняемость группы: 12-15 человек.

Цель: формирование интереса обучающихся к изучению профессий, связанных с основами программирования через освоение языка Python.

Задачи:

Обучающие:

1. Освоить основные алгоритмические конструкции.
2. Обучить основам алгоритмизации и программирования.
3. Освоить первоначальные навыки программирования на языке программирования высокого уровня Python.
4. Приобщить к проектно-творческой деятельности.

Развивающие:

1. Развить творческое воображение, алгоритмическое мышление учащихся.
2. Развить навыки планирования проекта, умения работать в группе.
3. Развить навыки ориентации в информационных потоках окружающего мира и применения точной и понятной инструкции для решения учебных задач и в повседневной жизни.

Воспитательные:

1. Воспитать интерес к информационной и коммуникационной сфере человеческой деятельности.
2. Воспитать потребность соблюдать этические и правовые нормы работы с информацией.
3. Воспитать бережное отношение к техническим устройствам.

Планируемые результаты:

Предметные результаты:

1. Освоят основные алгоритмические конструкции;
2. Узнают основы алгоритмизации и программирования;
3. Запомнят первоначальные навыки программирования на языке программирования высокого уровня Python;
4. Приобщатся к проектно-творческой деятельности;

Метапредметные результаты:

1. Развито творческое воображение, алгоритмического мышления учащихся.
2. Развит навык планирования проекта, умения работать в группе.
3. Развиты навыки ориентации в информационных потоках окружающего мира и применения точной и понятной инструкции для решения учебных задач и в повседневной жизни.

Личностные результаты:

1. Воспитан интерес к информационной и коммуникационной сфере человеческой деятельности.
2. Воспитано потребность соблюдать этические и правовые нормы работы с информацией.
3. Воспитано бережное отношение к техническим устройствам.

Содержание программы Учебный план

	Наименование раздела	Количество часов			Формы аттестации/контроля
		теория	практика	всего	
1	Знакомство с языком Python	1	2	3	Практическая работа
2	Переменные и выражения	1	3	4	Практическая работа
3	Условные операторы	4	4	8	Практическая работа
4	Циклы	5	5	10	Практическая работа
5	Функции	1	3	4	Практическая работа

6	Строки	2	3	5	Практическая работа
7	Итоговое тестирование по курсу	0	2	2	Практическая работа
8	РЕЗЕРВ УЧЕБНОГО ВРЕМЕНИ	1	1	2	
	ИТОГО	15	23	38	

Содержание учебного плана

Тема 1. Знакомство с языком Python

Теория. Общие сведения о языке Python.

Практика. Установка Python на компьютер. Режимы работы Python. Что такое программа. Первая программа. Структура программ на языке Python. Комментарии.

Тема 2. Переменные и выражения

Теория. Типы данных. Преобразование типов. Переменные. Оператор присваивания. Имена переменных и ключевые слова. Выражения. Операции. Порядок выполнения операций. Математические функции. Композиция.

Практика. 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.

Тема 3. Датчики и обратная связь.

Теория. Логический тип данных. Логические выражения и операторы.

Практика. Сложные условные выражения (логические операции and, or, not). Условный оператор. Примеры решения задач с условным оператором. Множественное ветвление. Реализация ветвления в языке Python.

Тема 4. Циклы

Теория. Понятие цикла. Тело цикла. Условия выполнения тела цикла.

Практика. 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.

Тема 5. Функции

Теория. Создание функций. Параметры и аргументы. Локальные и глобальные переменные. Поток выполнения. Функции, возвращающие результат. Анонимные функции, инструкция lambda.

Практика. Примеры решения задач с использованием функций. Рекурсивные функции. Вычисление факториала. Числа Фибоначчи.

Тема 6. Строки

Теория. Составной тип данных - строка. Доступ по индексу. Длина строки и отрицательные индексы. Преобразование типов. Применение цикла для обхода строки. Срезы строк. Сравнение строк. Оператор in. Модуль string. Операторы для всех типов последовательностей (строки, списки, кортежи).

Практика. Примеры решения задач со строками.

Тема 7. Итоговое тестирование по курсу.

Форма аттестации планируемых результатов:

Способы определения результативности реализации программы:

- педагогическое наблюдение на каждом занятии;
- активность обучающихся на занятиях;
- итоговое выполнение учащимися диагностических заданий;
- итоговое тестирование.

Во время проведения курса предполагается текущий, промежуточный и итоговый контроль.

Текущий контроль проводится на каждом занятии с целью выявления правильности применения теоретических знаний на практике. Текущий контроль может быть реализован посредством следующих форм: наблюдение, индивидуальные беседы, тестирование, творческие работы, проблемные (ситуативные) задачи, практические работы, контрольные вопросы и т. д .

Комплекс организационно- педагогических условий

Методическое обеспечение.

В рамках реализации программы предусматриваются следующие методы организации учебно-познавательной деятельности, позволяющие повысить эффективность обучения по курсу:

1. Объяснительно — иллюстративный (беседа, объяснение, инструктаж, демонстрация, работа с пошаговыми технологическими карточками и др.);
2. Метод проблемного изложения (учитель представляет проблему, предлагает ее решение при активном обсуждении и участии обучающихся в решении);
3. Эвристический (метод творческого моделирования деятельности).
4. Метод проектов.

Практический опыт многолетней работы показывает, что в выборе методов обучения, в организации учебного процесса необходимо учитывать специфику данной программы, и для успешной ее реализации необходимо соблюдение следующих условий.

Условия реализации программы

Кадровое обеспечение:

Педагог, работающий по данной программе, должен иметь высшее образование.

Обеспечение программы предусматривает наличие следующих методических видов продукции:

- электронные учебники;
- видео ролики;
- мультимедийные интерактивные домашние работы, выдаваемые обучающимся на каждом занятии.

Материально-техническое обеспечение

Кабинет оснащённый компьютерами с доступом к сети интернет, проектором.

Перечень используемого оборудования и материалов: рабочее место для работы с компьютером; компьютер с ОС Windows и выходом в Интернет; рабочая тетрадь ученика.

Оценочные материалы:

Тема 1. Знакомство с языком Python

Учащиеся должны знать / понимать: понятие программы; структура программы на Python; режимы работы с Python.

Учащиеся должны уметь: выполнить установку программы; выполнить простейшую программу в интерактивной среде; написать комментарии в программе.

Тема 2. Переменные и выражения

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.

Тема 3. Датчики и обратная связь.

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.

Тема 4. Циклы

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; при-меры 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;

Тема 5. Функции

Учащиеся должны знать / понимать: понятие функции; способы описания функции; принципы структурного программирования; понятие локальных переменных подпрограмм; понятие формальных и фак-тических параметров подпрограмм; способ передачи параметров.

Учащиеся должны уметь: создавать и использовать функции; использовать механизм параметров для передачи значений.

Тема 6. Строки

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.

Примеры оценочных материалов

Контрольная работа для проверки полученных навыков по темам «Условный оператор `if`», «Циклы в языке Python»

1. Найти корни квадратного уравнения, заданного своими коэффициентами
2. Вычислить значение функции $y = x^3 + 5x$ для заданного с клавиатуры значения аргумента x .
3. Проверить, есть ли во введённом числе одинаковые цифры, например 121.
4. Напечатать столбиком все целые числа от 20 до 35 и вычислить корни этих чисел. 7. Найти все трёхзначные числа, которые при увеличении на 3 делятся на 5 .
5. Даны натуральные числа от 20 до 50 . Напечатать те из них, которые делятся на 3, но не делятся на 5 .

Контрольная работа для проверки полученных навыков по темам

«Списки в языке Python»

1. Поменять местами самый большой и самый маленький элементы списка .
2. Даны два списка . Получить третий список, включая в него только те элементы, которые встречаются в исходных списках только 1 раз .
3. Сформировать возрастающий список из чётных чисел данного списка .
4. Даны два списка . Удалить все элементы первого списка из второго, остальные элементы второго списка отсортировать .
5. Создать список из случайных чисел. Найти номер его последнего локального максимума (локальный максимум — это элемент, который больше любого из своих соседей) .
6. Создать список из случайных чисел. Найти второй по величине максимум.

Список литературы и электронных ресурсов педагога.

1. Домашняя страница Python www.python.org . Справочные материалы, официальная документация.
2. Сайт проекта Интуит: Национальный открытый университет, курс «Введение в программирование на Python», <http://www.intuit.ru/studies/courses/12179/1172/info> .
3. Сайт проекта Интуит: Национальный открытый университет. Курс «Язык программирования Python» <http://www.intuit.ru/studies/courses/49/49/info> .
4. Python. Подробный справочник Дэвида М. Бизли — книга со справочной информацией о языке Python и модулях стандартной библиотеки.

Список литературы и электронных ресурсов обучающихся

1. Практический Python 3 для начинающих <https://pythonworld.ru/samouchitelpython>
2. Python для начинающих 2021 – уроки, задачи и тесты <https://pythonru.com/uroki/python-dlja-nachinajushhih>