

**государственное бюджетное общеобразовательное учреждение Самарской области многопрофильный лицей № 16
города Жигулевска городского округа Жигулевск Самарской области**

«Утверждено»

к использованию

Директор

ГБОУ лицея № 16 г.Жигулевска

приказ от 30 » августа 2020г. № 162-од

«Согласовано»

Заместитель директора по учебно-
воспитательной работе

ГБОУ лицея № 16 г.Жигулевска

28 августа 2020г.

«Рассмотрено»

на заседании МО учителей математики и
информатики

протокол № 1 от 27 августа 2020г.

**Рабочая программа
курса внеурочной деятельности «Программирование в Python»
для 7 классов**

Сроки реализации программы – 1 год

Разработчик программы: Копылова Елена Павловна

Год разработки программы – 2020

Пояснительная записка

Рабочая программа курса внеурочной деятельности «Программирование в Python» составлена на основе учебной программы «Программирование на Python и C++» для школьников, автор К.Ю. Поляков, 2019г.

Программа разработана с использованием новых образовательных компетенций в рамках регионального проекта «Современная школа» (в форме центров образования цифрового и гуманитарного профилей «Точка роста»).

Цель изучения данного курса: Способствовать формированию творческой личности, обладающей информационными компетенциями, владеющей базовыми понятиями теории алгоритмов, умеющей разрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке программирования Python.

Программа отвечает требованиям направления региональной политики в сфере образования – развитие научно-технического творчества детей школьного возраста.

В рамках предлагаемого курса «Программирования 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. Особую роль программирование служит для формирования мыслительных и психических процессов учащихся (внимание, память, логика), освоения приёмов умственных действий, самостоятельного нахождения и составления алгоритмов решения задач, умения строить модели, чётко и лаконично реализовывать этапы решения задач. Использование этих возможностей для формирования предметных и метапредметных результатов учащихся особенно важно, т.к. именно они активизирует процесс индивидуально-личностного становления учащихся.

Основные задачи курса:

1. Образовательные:

- познакомить с понятиями алгоритма, вычислимой функции, языка программирования;
- научить составлять и читать блок-схемы;
- сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Python;
- объяснить основные конструкции языка программирования Python, позволяющие работать с простыми и составными типами данных (строками, списками, кортежами, словарями, множествами);
- научить применять функции при написании программ на языке программирования Python;
- научить отлаживать и тестировать программы, делать выводы о работе этих программ.

2. Воспитательные:

- воспитать уважительное отношение к преподавателям и сверстникам, культуру поведения во время занятий и совместной продуктивной деятельности;
- сформировать культуру занятий, направленную на воспитание личностных и социальных качеств;

3. Развивающие:

- развить познавательные процессы (внимание, восприятие, логическое мышление, память),
- развить креативность,
- развить способности к самореализации.

Формы 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 педагогом, в переходе от репродуктивного вида работ к самостоятельным, поисково-исследовательским видам деятельности. Поэтому основная методическая установка в данном курсе – обучение учащихся навыкам самостоятельной творческой деятельности.

Теория преподносится в форме беседы, эвристической беседы, презентации, обзора и т.п.

Практические занятия проходят в форме тестирования, практикума, совместной продуктивной деятельности, моделирования и защиты проектов, проблемном изложении материала, с помощью которого дети сами решают познавательные задачи.

Организация занятий.

Занятия проводятся 1 раз в неделю по 1 учебному часу (34 часа).

Используемый учебно–методический комплект:

- «Программирование на Python и C++» для школьников, К.Ю. Поляков, 2019г.
- Сайт / интерактивный сборник задач для практики программирования [Электронный ресурс] – Режим доступа: <http://pythontutor.ru/>, свободный.
- Лутц, Марк Python. Справочник. – М.: Вильямс, 2015.

Перечень средств ИКТ, необходимых для реализации программы

Материально-техническое обеспечение:

- персональный компьютер (на каждого участника);
- мультимедийный проектор;
- видеоматериалы разной тематики по программе;
- оргтехника;
- выход в сеть Internet;

Программное обеспечение:

- Операционная система Windows 7 Профессиональная или выше;
- Интерпретатор Python версии 3.7 и выше;
- Wing 101;
- Любой браузер для интернет серфинга.

Место курса в плане внеурочной деятельности

Класс	количество		
	учебных недель	часов в год	часов в неделю
7«А», 7«Б»	34	34	1

Содержание курса 7 класса

№ раздела, темы	Раздел, тема	Итого
1	Введение. Знакомство с Python	3
1.1	Знакомство с WING 101 для Python	1
1.2	Вычисления и переменные	1
1.3	Первая программа на Python	1
2	Построение программы на языке Python	3
2.1	Строки и списки	1
2.2	Синтаксис языка Python	1
2.3	Операторы Python	1
3	Инструкция if - elif – else. Выбор подходящего варианта. Ветвление	3
3.1	Инструкция if - elif – else	1
3.2	Проверка истинности if - elif – else	2
4	Цикл в языке программирования Python	10
4.1	Цикл for	4
4.2	Цикл while	4
4.3	Операторы break и continue	2
5	Строки	4
5.1	Составной тип данных - строка	1
5.2	Доступ по индексу. Длина строки и отрицательные индексы	1
5.3	Преобразование типов. Применение цикла для обхода строки	1
5.4	Срезы строк	1
6	Функции в программировании	8
6.1	Параметры и аргументы функций	2
6.2	Локальные и глобальные переменные	2
6.3	Процедуры. Рекурсия	4
7	Файлы. Работа с файлами	4
7.1	Работа с файлами	2
7.3	Документирование кода в Python	2
	Итого:	34

Раздел 1. Введение. Знакомство с Python (3 часа)

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

Практическая работа 1.1. Установка программы Python Практическая работа 1.2. Режимы работы с 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.

Раздел 2. Построение программы на языке Python (3 часа)

Теория: Типы данных. Преобразование типов. Переменные. Оператор присваивания. Имена переменных и ключевые слова.

Выражения. Операции. Порядок выполнения операций. Математические функции.

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.

Практическая работа 2.1. Переменные. Выражения. Практическая работа 2.2. Задачи на элементарные действия с числами

Учащиеся должны знать / понимать:

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

Раздел 3. Инструкция if - elif – else. Выбор подходящего варианта. Ветвление (3 часа)

Теория: 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 3.1. Логические выражения Практическая работа 3.2. "Условный оператор" Практическая работа 3.3. Множественное ветвление

Тест № 1. "Условные операторы".

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.

Раздел 4. Цикл в языке программирования Python (10 часов)

Теория: Понятие цикла. Тело цикла. Условия выполнения тела цикла. Оператор цикла с условием. Оператор цикла while. Бесконечные циклы. 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 range. Функция random. Примеры решения задач с циклом.

Практическая работа 4.1. Решение задачи с циклом for. Практическая работа 4.2. Случайные числа

Практическая работа 4.3. Решение задач с циклом while. Тест № 2. Циклы

Учащиеся должны знать / понимать:

- циклы с условием и их виды;
- правила записи циклов 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.

Раздел 5. Строки (4 часа)

Теория: 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. Примеры решения задач со строками.

Практическая работа 6.1. Строки Практическая работа 6.2. Решение 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.

Раздел 5. Функции в программировании (8 часов)

Теория: Создание функций. Параметры и аргументы. Локальные и глобальные переменные. Примеры решения задач с использованием функций.

Рекурсивные функции. Вычисление факториала. Числа Фибоначчи. Практическая работа 5.1. Создание функций

Практическая работа 5.2. Локальные переменные Практическая работа 5.3. Решение задач с использованием функций

Практическая работа 5.4. Рекурсивные функции

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.

Раздел 6. Файлы. Работа с файлами (4 часа)

Теория

Учащиеся должны знать / понимать:

- Что такое файл;
- Какие команды для открытия файла, чтения данных из файла, записи в файл, закрытия файла.

Учащиеся должны уметь:

- уметь читать текстовые файлы в заданном формате
- уметь записывать текстовые файлы в заданном формате

Прогнозируемые личностные и метапредметные результаты освоения курса

1. Коммуникативные универсальные учебные действия: формировать умение слушать и понимать других; формировать и отрабатывать умение согласованно работать в группах и коллективе; формировать умение строить речевое высказывание в соответствии с поставленными задачами.
2. Познавательные универсальные учебные действия: формировать умение извлекать информацию из текста и иллюстрации; формировать умения на основе анализа рисунка-схемы делать выводы.
3. Регулятивные универсальные учебные действия: формировать умение оценивать учебные действия в соответствии с поставленной задачей; формировать умение составлять план действия на уроке с помощью учителя; формировать умение мобильно перестраивать свою работу в соответствии с полученными данными.
4. Личностные универсальные учебные действия: формировать учебную мотивацию, осознанность учения и личной ответственности, формировать эмоциональное отношение к учебной деятельности и общее представление о моральных нормах поведения.

Ожидаемые предметные результаты реализации программы

В рамках курса «Программирование на языке Python» учащиеся овладевают следующими знаниями, умениями и способами деятельности:

- умеют составлять алгоритмы для решения задач;
- умеют реализовывать алгоритмы на компьютере в виде программ, написанных на языке Python;
- владеют основными навыками программирования на языке Python;
- умеют отлаживать и тестировать программы, написанные на языке Python.

Контроль и оценивание предметных результатов

Предметом диагностики и контроля в курсе «Основы программирования на языке Python» являются внешние образовательные продукты учащихся (созданные блок-схемы, программы), а также их внутренние личностные качества (освоенные способы деятельности, знания, умения), которые относятся к целям и задачам курса.

Качество внешней образовательной продукции желательно оценивать по следующим параметрам:

- алгоритм должен быть оптимальным по скорости выполнения и максимально простым в реализации на языке программирования;
- программа должна выполнять поставленные задачи;
- по степени «читаемости кода» (должны быть соблюдены отступы, обязательное наличие комментариев к коду программы и т. д.).

Проверка достигаемых учащимися результатов производится в следующих формах:

- текущий рефлексивный самоанализ, контроль и самооценка учащимися выполняемых заданий;
- текущая диагностика и оценка педагогом деятельности учащихся;
- итоговая оценка деятельности и образовательной продукции ученика в соответствии с его индивидуальными и возрастными особенностями;
- Промежуточный контроль проводится в конце изучения некоторых тем. При изучении каждой темы проводятся практические работы.