

**МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Министерство образования Калининградской области**

**Управление образования администрации МО**

**«Зеленоградский муниципальный округ Калининградской области»**

**МАОУ "Гимназия "Вектор" г. Зеленоградска"**

ПРИНЯТО

на педагогическом совете  
Протокол № 5 от  
31.05.2023 г.

УТВЕРЖДЕНО

ИО Директора МАОУ «Гимназия  
«Вектор» г. Зеленоградска»



Пеленс К.А.

Приказ № от 313-ОД от 31.05.2023 г.

**РАБОЧАЯ ПРОГРАММА ВНЕУРОЧНОЙ ДЕЯТЕЛЬНОСТИ  
«ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON»**

8класс

Разработчик:  
Коптева Светлана Ивановна  
учитель информатики.

г. Зеленоградск  
2023-2024

### **Пояснительная записка.**

Курс внеурочной деятельности «Основы программирования на языке Python» представляет собой вводный курс по программированию, дающий представление о базовых понятиях структурного программирования (данных, операциях, переменных, ветвлениях в программе, циклах и функциях, массивах, файлах), о реализации с помощью.

Изучая программирование на Python, учащиеся прочнее усваивают основы алгоритмизации, приобщаются к алгоритмической культуре, познают азы профессии программиста.

Основа курса – личностная, практическая и продуктивная направленность занятий. Одна из целей обучения информатике – предоставить ученикам возможность личностного самоопределения и самореализации по отношению к стремительно развивающимся информационным технологиям и ресурсам.

#### **Цели курса:**

- Познакомить учащихся с ролью программного обеспечения и его видами.
- Сформировать целостное представление об организации данных для эффективной алгоритмической обработки.
- Развитие логического мышления.
- Реализация математических способностей учащихся в ходе составления программ на языке программирования.

#### **Задачи курса:**

##### ***Обучающие:***

- Познакомить учащихся с основными алгоритмическими конструкциями и правилами их записи, с основными способами организации данных.
- Научить учащихся 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 учащихся разрабатывать алгоритмы и реализовывать их на языке программирования Python.
- Научить учащихся осуществлять отладку и тестирование программы.

##### ***Развивающие:***

- формировать новый тип мышления – операционный, который направлен на выбор оптимальных решений;
- предоставление возможности узнать новое в области компьютерного программирования;
- формирование представления о роли компьютерного программирования в развитии общества, изменении содержания и характера деятельности человека.

##### ***Воспитательные:***

- повышение общекультурного уровня учащихся;
- вооружение учащихся 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;
- формирование эмоционально-ценностного отношения к миру, к себе;
- воспитание у учащихся стремления к овладению техникой исследования;
- воспитание трудолюбия, инициативности и настойчивости в преодолении трудностей.

### **Общая характеристика курса**

Курс включает в себя практическое освоение языка программирования, знакомство учащихся с ролью программного обеспечения и его видами; нацелен на формирование целостного представления об организации данных для эффективной алгоритмической обработки; на развитие логического мышления и реализацию математических способностей учащихся в ходе составления программ на языке программирования.

Основа курса – личностная, практическая и продуктивная направленность занятий. Одна из целей обучения информатике – предоставить ученикам возможность личностного самоопределения и самореализации по отношению к стремительно развивающимся информационным технологиям и ресурсам.

#### **Описание места в учебном плане**

Курс внеурочной деятельности «Основы программирование на языке Python» является предметом по выбору для учащихся 8 класса средней школы.

Курс рассчитан на 34 часа, которые проводятся в течение учебного года по 1 часу в неделю. Концентрированное изучение курса позволяет учащимся более полно выявить свои способности в изучаемой области знаний, создать предпосылки по применению освоенных умений в других учебных курсах, подготовить себя к осознанному выбору профессий, предусматривающих программирование.

**Формы занятий:** Основными, характерными при реализации данной программы формами являются комбинированные занятия. Занятия состоят из теоретической и практической частей, причём большее количество времени занимает практическая часть.

В теоретической части рассматриваются основные понятия языка программирования Python, основные алгоритмические конструкции. В практической части предлагаются практические работы, направленные на отработку основных алгоритмических конструкций, на развитие логического мышления, на реализацию математических способностей учащихся в ходе составления программ. Практическая часть предполагает использование школьного компьютерного класса.

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;
- практические занятия.

#### **Режим занятий:**

Занятия проводятся: 1 раз в неделю по 1 часу (итого 1 час в неделю, 34 часа в год).

Данный курс предусматривает классно-урочную систему обучения.

#### **Межпредметные связи**

Знания, полученные при изучении курса « Основы программирование на языке Python», учащиеся могут использовать при создании собственных программ по определенной тематике, для решения задач из различных областей знаний – математике, физике, химии, биологии и др. Знания и умения, приобретенные в результате освоения данного курса, являются фундаментом для дальнейшего мастерства в области программирования.

#### **Методическая часть**

Основной тип занятий – практикум. Большинство заданий курса выполняется с помощью персонального компьютера и необходимых программных средств.

Занятия включают лекционную и практическую часть. Практическая часть курса реализуется через классно-урочную систему. Важной составляющей каждого урока

является самостоятельная работа учащихся. Тема урока определяется приобретаемыми навыками. В каждом уроке материал излагается следующим образом: повторение основных понятий и методов работы с ними, разбор новой темы, основные приемы работы (самостоятельное выполнение заданий для получения основных навыков работы), упражнения для самостоятельного выполнения.

Теоретическая и практическая части курса изучаются параллельно, чтобы сразу же закреплять теоретические вопросы на практике.

В ходе обучения учащимся периодически предлагаются короткие (5-10 мин) контрольные работы на проверку освоения изученных способов действий. Проводятся краткие срезовые работы (тесты, творческая работа) по определению уровня знаний учеников по данной теме. Выполнение контрольных способствует быстрой мобилизации и переключению внимания на осмысливание материала изучаемой темы. Кроме того, такая деятельность ведет к закреплению знаний и служит регулярным индикатором успешности образовательного процесса.

Регулярное повторение способствует закреплению изученного материала. Возвращение к ранее изученным темам и использование их при изучении новых тем способствуют устранению весьма распространенного недостатка – формализма в знаниях учащихся – и формируют научное мировоззрение учеников.

### **Результаты освоения учебного курса**

В результате изучения курса получают дальнейшее развитие *личностные, регулятивные, коммуникативные и познавательные универсальные учебные действия, учебная (общая и предметная) и общепользовательская ИКТ-компетентность обучающихся.*

В основном формируются и получают развитие *метапредметные результаты*, такие как:

- умение самостоятельно планировать пути достижения целей, осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач;
- умение соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата;
- умение оценивать правильность выполнения учебной задачи, собственные возможности ее решения;
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач;
- владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;
- умение организовывать учебное сотрудничество совместную деятельность с учителем и сверстниками;
- формирование и развитие компетентности в области использования информационно коммуникационных технологий (далее ИКТ-компетентности).

Вместе с тем вносится существенный вклад в развитие *личностных результатов*, таких как:

- формирование ответственного отношения к учению;
- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками, детьми старшего и младшего возраста, взрослыми в процессе образовательной, творческой и других видов деятельности.

В части развития *предметных результатов* наибольшее влияние изучение курса оказывает:

- на формирование умений формализации и структурирования информации, умения выбирать способ представления данных в соответствии с поставленной задачей с использованием соответствующих программных средств обработки данных;
- формирование навыков и умений безопасного и целесообразного поведения при работе с компьютерными программами и в Интернете, умение соблюдать нормы информационной этики и права.
- умеют составлять и записывать алгоритмы с использованием соответствующих алгоритмических конструкций;
- умеют распознавать необходимость применения той или иной алгоритмической конструкции при решении задачи;
- умеют организовывать данные для эффективной алгоритмической обработки;
- умеют разрабатывать алгоритмы и реализовывать их на языке программирования Python;
- умеют осуществлять отладку и тестирование программы

### **Форма контроля за уровнем достижения учащихся**

Предметом диагностики и контроля являются составленные алгоритмы и программы на языке программирования Python к предложенным задачам.

Оценка имеет различные способы выражения – устные суждения педагога, письменные качественные характеристики, систематизированные по заданным параметрам аналитические данные, в том числе и рейтинги. Оценке подлежит в первую очередь уровень достижения учеников минимально необходимых результатов, обозначенных в целях и задачах курса.

Качество знаний и умений ученика оценивается следующими характеристиками:

- знание основных алгоритмических конструкций;
- умение составить и записать алгоритм с использованием соответствующей алгоритмической конструкции;
- умение найти более эффективный способ решения задачи;
- умение тестировать программу.

### **Основное содержание.**

1. Введение. Знакомство с языком Питон. Структура программы. Окно программы. Познакомить учащихся с алфавитом, структурой программы. Рассмотреть окно среды программирования, интерфейс, меню, как пользоваться справкой, показать пример программы, ввод и вывод значений переменных. Рассмотреть вопросы техники безопасности и здоровьесберегающих технологий при работе за компьютером.

2. Операторы ввода-вывода. Запись программы. Структура программы, блоки. Познакомить учащихся с переменными величинами и их типами, рассмотреть операторы ввода-вывода и присваивания. Работа с готовой программой. Особенности линейного ввода. Особенности форматного вывода.

3. Составление первой программы (самостоятельно). Линейные алгоритмы и программы. Отладка. Математические функции на Питоне. Программы с их использованием.

4. Алгоритмы с ветвлением. Решение задачи на вычисление значений многочлена. Познакомить учащихся с алгоритмами ветвления и рассмотреть задачи полного и неполного ветвления.

5. Циклические алгоритмы и программы. Ввести представление о циклических алгоритмах. Рассмотреть их виды.

6. Массивы. Обработка массива. Задание массива. Ввести учащимся определение массива. Представить способы задания массива и вывода его членов на экран.

7. Программы на ввод и обработку элементов массива. Закрепить с учащимися понятие массива. Рассмотреть основные задачи на массивы.

8. Двумерные массивы. Дать учащимся представление о двумерных массивах. Рассмотреть их заполнение и вывод на экран.

9. Строковые данные. Особенности работы со строковыми данными. Строковые массивы.

10. Библиотечные модули. Стандартные модули. Способы подключения модулей, и работа с ними. Графический модуль.

11. Текстовые файлы. Ввод и вывод с использованием текстовых файлов.

#### Тематическое планирование.

		Кол-во часов	теория	практика
1	Введение. Знакомство с языком Питон.	1	1	0
2	Операторы ввода-вывода.	2	1	1
3	Линейные алгоритмы и программы	7	3	4
4	Алгоритмы с ветвлением	4	1	3
5	Циклические алгоритмы и программы	6	2	4
6	Массивы. Обработка массива.	2	1	1
7	Двумерные массивы	5	1	4
8	Строковые данные	2	1	1
9	Библиотечные модули. Графический модуль. Рисование в Python	3	1	2
10	Текстовые файлы	2	1	1
	<b>ИТОГО</b>	<b>34</b>	<b>13</b>	<b>21</b>

#### Календарно-тематическое планирование

№ п/п	Тема	Количество часов
1	Инструктаж по технике безопасности. Среда Python. Элементы интерфейса. Использование справки. Структура программы.	1
2	Операторы ввода-вывода. Запись программы. Первая программа в Python. Линейные алгоритмы и программы.	1
3	Синтаксис языка. Исполнение программы. Отладка. Сообщения об ошибках. Типичные ошибки.	1
4	Арифметические выражения. Формат результата вычислений. Примеры программ на вычисления.	1
5	Арифметические выражения. Формат результата вычислений. Примеры программ на вычисления.	1
6	Оператор присваивания. Операторы ввода и вывода. Форматный вывод. Линейный ввод.	1
7	Оператор присваивания. Операторы ввода и вывода. Форматный вывод. Линейный ввод.	1
8	Пошаговое выполнение программы	1
9	Данные логического типа. Запись условий. Составные условия и их запись.	1

10	Данные логического типа. Запись условий. Составные условия и их запись.	1
11	Алгоритмы с ветвлением. Условный оператор. Полное и неполное ветвление.	1
12	Алгоритмы с ветвлением. Условный оператор. Полное и неполное ветвление.	1
13	Сложное ветвление.	1
14	Сложное ветвление.	1
15	Циклические алгоритмы Цикл с предусловием (while)	1
16	Циклические алгоритмы Цикл с предусловием (while)	1
17	Циклические алгоритмы Цикл с предусловием (while)	1
18	Цикл с параметром (for).	1
19	Цикл с параметром (for).	1
20	Цикл с параметром (for).	1
21	Одномерные массивы. Описание массива, основные типы задач с массивами	1
22	Одномерные массивы	1
23	Двумерные массивы.	1
24	Двумерные массивы.	1
25	Двумерные массивы.	1
26	Вложенные циклы.	1
27	Строковые данные. Функции для работы со строковыми данными.	1
28	Строковые массивы.	1
29	Графический модуль. Рисование в Python	1
30	Графический модуль. Рисование в Python	1
31	Графический модуль. Рисование в Python	1
32	Текстовые файлы	1
33	Текстовые файлы	1
34	Обобщение	1