

Муниципальное бюджетное общеобразовательное учреждение  
«Средняя школа № 4 имени А.М. Горького»,  
структурное подразделение – «Городская физико-математическая школа»

«Рассмотрено»

Методист ФМПШ

Е.А. Лутцева

« 30 » 08 2021 г.

«Согласовано»

Руководитель ФМПШ

О.А. Зимоглядова

« 30 » 08 2021 г.

«Утверждаю»

Директор МБОУ СШ № 4

Л.А. Матвеева

« 30 » 08 2021 г.

Приказ № 79/2021



ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА  
ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ ДЕТЕЙ

«Программирование Python»

Адресат программы: обучающиеся 8-го и 9-ого классов  
Срок реализации программы: 2021-2022 учебный год  
Автор-составитель программы: Ким Алина Александровна

г. Петропавловск-Камчатский  
2021-2022 учебный год

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Характерной чертой развития общества на протяжении последних десятилетий является его все более расширяющаяся информатизация. Отражением и следствием этой тенденции явилась потребность в подготовке подрастающего поколения к вступлению в информатизированное общество, любая профессиональная деятельность в котором, будет связана с информатикой и информационными технологиями. Умение представлять информацию в виде, удобном для восприятия и использования другими людьми, – одно из условий социальной компетентности ученика. Это добавляет новую цель в образовании – формирование уровня информационной культуры, соответствующего требованиям информационного общества. Наиболее полно реализовать поставленную цель, призвана образовательная область «информатика».

Информатика обязана знакомить человека не только с навыками работы на компьютере и существующими информационными технологиями, но и с основами алгоритмики, реализуемыми в форме моделирования объектов и процессов в различных средах программирования. Часто в школах этому не уделяется должного внимания, так как именно эта часть предмета требует высокой квалификации учителя как математика, алгоритмиста и программиста.

Данный курс направлен на удовлетворение образовательных потребностей тех учащихся, которые хотели бы более подробно ознакомиться с основами программирования, попробовать себя в реализации индивидуальных и общих проектов.

Курс является вводным в основы программирования на языке высокого уровня Python. Для написания программ предполагается использовать среду программирования IDLE.

### Цели курса

Целью курса является приобретение учащимися базового набора знаний, умений и навыков по программированию. Особое внимание уделяется развитию алгоритмического стиля мышления учащихся. Под способностью алгоритмически мыслить понимается умение решать задачи различного происхождения, требующие составления плана действий для достижения желаемого результата.

### Задачи курса

#### *Обучающие:*

1. познакомить учащихся с основными принципами хранения данных на компьютере;
2. познакомить с типовыми алгоритмами и правилами их записи: ввод-вывод данных, использование условий, использование циклов;
3. научить учащихся составлять и записывать алгоритмы в различных формах представления;
4. познакомить с языком программирования Python и средой программирования IDLE;
5. научить писать программы на языке Python на основе составляемых алгоритмов;
6. научить осуществлять отладку и тестирование программ.

#### *Развивающие:*

1. профессиональное самоопределение учащихся;
2. расширение знаний и навыков в области компьютерных технологий.

#### *Воспитательные:*

1. повышение общекультурного уровня учащихся;

2. вооружение учащихся правильным методологическим подходом к познавательной и практической деятельности;
3. Привитие навыков сознательного и рационального использования компьютера в своей учебной, а затем и профессиональной деятельности;
4. Воспитание у учащихся стремления к овладению техникой исследования;
5. Воспитание трудолюбия, инициативности и настойчивости в преодолении трудностей.

**Возраст детей:** 14-15 лет (учащиеся 8-9 классов).

**Сроки реализации курса:** 1 учебный год.

### **Формы занятий**

Основной формой проведения занятий являются комбинированные занятия, состоящие из теоретической и практической частей, причём большее количество времени занимает практическая часть.

В *теоретической* части рассматриваются алгоритмические конструкции и основные понятия языка программирования Python.

В *практической* части предлагаются практические работы на составление алгоритмов и написание программ на языке программирования Python. Одна часть работ выполняется учащимися в рабочих тетрадях, а другая – с помощью персонального компьютера в среде программирования IDLE. Практическая часть предполагает использование школьного компьютерного класса.

**Технологии и формы обучения:** теоретические и практические занятия.

**Форма обучения:** очная с возможностью применения дистанционных технологий (видео-, конференцсвязь, социальные сети и мессенджеры).

### **Межпредметные связи**

Знания, полученные при изучении курса «Программирование на языке Python», учащиеся могут использовать при создании программного обеспечения по определенной тематике, для решения задач из различных областей знаний – математики, физики, химии, биологии и др. Знания и умения, приобретенные в результате освоения данного курса, являются фундаментом для дальнейшего развития в области программирования.

### **Планируемые результаты**

**Учащиеся должны знать:**

- алгоритмические конструкции, исполнители;
- формы записи алгоритмов;
- основы программирования на одном из языков программирования;
- общую структуру и синтаксис языка программирования Python;
- правила определения типа переменной и ее описания;
- правила записи операторов языка, порядок выполнения операций, стандартные и пользовательские функции, записи и выполнения арифметических и логических функций;

**Учащиеся должны уметь:**

- правильно записывать символы и ключевые слова языка программирования;

- записывать числовые и логические константы, задавать и описывать переменные;
- составлять алгоритм и программу для решения простых задач;
- понимать листинг, корректировать программу;
- выводить результаты на дисплей или записывать в файл;
- осуществлять отладку и тестирование программы.

### **Контроль уровня достижений учащихся**

Текущий контроль учебных достижений проводится на каждом практическом занятии. Учащиеся совместно с преподавателем выполняют задания на составление алгоритмов и написание программ. Качество выполнения заданий оценивается по следующим характеристикам:

- умение правильно разбить задачу на более мелкие подзадачи;
- умение составить рабочий алгоритм для поставленной задачи;
- знание основных конструкций языка программирования Python;
- умение подбирать типы переменных и констант;
- умение найти более эффективный способ решения задачи;
- умение проведения отладки и тестирования программы.

## Содержание дополнительной образовательной программы «Программирование Python»

- 1. Системы счисления (2 часа).**  
Понятие систем счисления (СС). Позиционные и непозиционные СС. Двоичная СС. Перевод из двоичной СС в десятичную и обратно. Арифметические действия в двоичной СС.
- 2. Алгоритмы (6 часов).**  
Понятие алгоритма. Исполнитель. Свойства алгоритма. Формы представления алгоритма. Основные алгоритмические конструкции. Блок-схемы.
- 3. Линейные программы (8 часов).**  
Язык программирования Python, основные понятия. Среда программирования IDLE. Структура программы. Переменные и константы. Ввод/вывод данных. Выполнение математических операций. Оператор присваивания. Некоторые математические функции.
- 4. Ветвления (10 часов).**  
Логический тип данных. Основные логические операции. Условный оператор if с простым и сложным условием. Вложение условных операторов.
- 5. Циклы (10 часов).**  
Виды циклических конструкций. Цикл с предусловием, цикл с постусловием, цикл с параметром. Простые и вложенные циклы.
- 6. Массивы (10 часов).**  
Понятие массива. Размерность массива. Объявление и заполнение массива. Одномерные и двумерные массивы. Поиск элемента массива. Сортировка массива. Нахождение суммы, произведения, среднего значения.
- 7. Строки (4 часа).**  
Символьный и строковый типы данных. Таблица символов ASCII. Чтение/запись строк. Операции над строками. Процедуры и функции для работы со строками. Процедуры преобразования типов.
- 8. Работа с файлами (4 часа).**  
Понятие файла. Типизированные и нетипизированные файлы. Файловая переменная. Чтение и запись файла. Операции и функции для работы с файлами.
- 9. Индивидуальное задание (6 часов).**  
Анализ поставленной задачи. Разработка алгоритма. Проектирование общей структуры программы. Написание кода на языке программирования Python. Тестирование программы.

## Тематическое планирование

*Преподаватель: Ким А.А.*

№ занятия	Дата занятия	Тема	Кол-во часов
Тема 1		<b>Алгоритмы</b>	
1	16.09	Алгоритмы. Основные понятия, свойства алгоритмов. Разветвляющиеся алгоритмы.	2
2	23.09	Циклические алгоритмы. Построение блок-схем.	2
3	30.09	Решение задач по теме «Алгоритмы»	2
Тема 2		<b>Линейные программы</b>	
4	07.10	Основы языка Python.	2
5	14.10	Переменные и константы. Написание линейных программ	2
6	21.10	Написание линейных программ.	2
7	28.10	Использование некоторых математических функций.	2
Тема 3		<b>Ветвления</b>	
8	11.11	Логический тип данных. Условный оператор. Организация ветвлений в программе.	2
9	18.11	Вложение условных операторов.	2
10	25.11	Условный оператор со сложным условием.	2
11	02.12	Написание программ с использованием условных операторов.	2
12	09.12	Написание программ с использованием условных операторов.	2
Тема 4		<b>Циклы</b>	
13	16.12	Использование циклов в программе. Цикл с параметром.	2
14	23.12	Цикл с предусловием.	2
15	13.01	Цикл с постусловием.	2
16	20.01	Написание программ с использованием циклических операторов.	2
17	27.01	Вложение циклических операторов.	2
Тема 5		<b>Массивы</b>	
18	03.02	Одномерные массивы. Основные понятия. Заполнение массива и вывод на экран.	2
19	10.02	Написание программ с использованием одномерных массивов.	2
20	17.02	Двумерные массивы. Основные понятия. Заполнение массива и вывод на экран.	2

21	24.02	Написание программ с использованием двумерных массивов.	4
22	03.03	Реализация алгоритмов. Поиск максимального и минимального элементов в массиве. Сортировка массива. Поиск, суммы, произведения, среднего.	2
Тема 6		<b>Строки</b>	
23	10.03	Символьный и строковый типы данных. Основные операции.	2
24	17.03	Написание программ, реализующих обработку строк.	2
Тема 7		<b>Работа с файлами</b>	
25	24.03	Работа с файлами. Чтение и запись.	2
26	31.03	Написание программ. Чтение и запись информации в файлы.	2
Тема 8		<b>Индивидуальное задание</b>	
27	07.04	Написание программы по индивидуальному заданию преподавателя.	2
28	14.04	Написание программы по индивидуальному заданию преподавателя.	2
29	21.04	Написание программы по индивидуальному заданию преподавателя.	2
30	28.04	Написание программы по индивидуальному заданию преподавателя.	2
		Всего:	60

## Список литературы.

1. Гагарина Л.Г., Колдаев В.Д. Алгоритмы и структуры данных. – М.: Финансы и статистика, 2009. – 303 с.
2. Меньшиков Ф.В. Олимпиадные задачи по программированию. – СПб.: Питер, 2006.-315 с
3. Окулов С.М. Задачи по программированию. – М.: БИНОМ. Лаборатория знаний, 2014. –823 с.
4. Окулов С.М. Основы программирования. – М.: БИНОМ. Лаборатория знаний, 2012. – 336 с.
5. Семакин И. Г. Информатика. Базовый курс. Учебник для 9-го класса. – М.: БИНОМ. Лаборатория базовых знаний, 2005. – 371 с.
6. Банкрашков, А.В. Программирование для детей на языке Python / А.В. Банкрашков. - М.: АСТ, 2018. - 288 с.
7. Банкрашков, А.В. Программирование для детей на языке Python / А.В. Банкрашков. - М.: АСТ, 2018. - 288 с.
8. Лутц, М. Программирование на Python. Т. 2 / М. Лутц. - М.: Символ, 2016. - 992 с.
9. Лутц, М. Программирование на Python. Т. 1 / М. Лутц. - М.: Символ, 2016. - 992 с.