

ТЕЙКОВСКОГО МУНИЦИПАЛЬНОГО РАЙОНА  
«НЕРЛЬСКАЯ СРЕДНЯЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ШКОЛА»

155030, Ивановская область, Тейковский район, деревня Харино, дом 37, строение 1,  
тел/факс (49343) 49441

**РАССМОТРЕНО**

НАМО математического  
цикла

от « 29 » 08. 2022 г

Руководитель МО

Ершова Е.Е. /Ершова Е.Е. /

**СОГЛАСОВАНО**

Заместитель директора по УВР

Макарова И.Н. /Макарова И.Н./

« 30 » 08. 2022 г

**УТВЕРЖДЕНО**

Директор школы

Сальникова Г.В. /Сальникова Г.В./

Приказ № 58

« 31 » 08. 2022 г.



**РАБОЧАЯ ПРОГРАММА  
КУРСА ВНЕУРОЧНОЙ ДЕЯТЕЛЬНОСТИ  
«Программирование на Python»**

Уровень образования- 5-9 класс

Количество часов – 34 часа

Срок реализации – 1 год

Составитель (ли) Макарова Ирина Николаевна

*Д.Харино, 2022г*

## 1.. Содержание программы курса внеурочной деятельности «Программирование на Python»

### Тема 1. Введение в Python.(2 часа)

История языков программирования. Язык программирования Python .Общие сведения о языке Python, преимущества языка, сферы применения. Структура программы в языке Python. . Режимы работы Python. Использование командной строки интерпретатора. Выбор редактора. Работа со средой программирования. Сохранение, открытие и запуск программ.Создание и запуск простых программ

### Тема 2.Переменные и выражения. Операции. Организация ввода и вывода данных.(6 часов)

Типы данных. Преобразование типов. Переменные. Оператор присваивания. Имена переменных и ключевые слова. Выражения. Операции. Пример скрипта, использующего ввод и вывод данных. Задачи на элементарные действия с числами. Решение задач на элементарные действия с числами.

Арифметические операции. Арифметические выражения. Целочисленное деление и остаток от целочисленного деления. Целые числа и числа с плавающей запятой. Выражения. Вычисление выражений. Синтаксические ошибки. Сохранение значений в переменных. Строковые значения. Конкатенация строк.. Функция print(). Функция input().Создание программы «Привет, мир!».

### Тема 3.Понятие алгоритм. Типы алгоритмов. Линейный алгоритм и алгоритм с условием. Условный оператор.(10 часов)

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. Синтаксис языка Python. Преобразование типов данных, простые вычисления. Числовые типы данных: int, float.Строковый тип данных.Модуль math.Использование скобок. Логические выражения и операторы. Алгоритмические конструкции. Генерирование случайных чисел. Условная конструкция if. Программирование линейных алгоритмов. Конструкции if...else. Использование elif. Составные условия. Программирование разветвляющихся алгоритмов.

### Тема 4. Управляющие инструкции – циклы. (16 часов)

Понятие цикла. Тело цикла. Условия выполнения тела цикла. Генерация случайных чисел. Циклическая инструкция while. Циклическая инструкция for. Управление выполнением циклических инструкций. Команды break и continue. Вложенные циклы. Бесконечный цикл. Встроенные функции. Строки в Python: ввод и вывод. Операции со строками. Кортежи. Списки. Списочные методы. Словари. Использование словарей. Функции преобразования типов переменных.

Решение задач различного типа с помощью циклических алгоритмов.

*Создание игры «Волк, коза и капуста» с использованием циклического алгоритма.*

Создание игры «Анаграммы»

Создание игры «Виселица»

## 2. Планируемые образовательные результаты курса внеурочной деятельности «Программирование на Python»

### Личностные результаты:

- сформированность мировоззрения, соответствующего современному уровню

развития науки и техники;

- осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов

#### **Метапредметные результаты:**

- владение навыками познавательной, учебно-исследовательской и проектной деятельности, навыками разрешения проблем;
- способность и готовность к самостоятельному поиску методов решения практических задач, применению различных методов познания

#### **Предметные результаты**

После изучения курса учащиеся должны:

- владеть навыками алгоритмического мышления и понимание необходимости формального описания алгоритмов;
- владеть стандартными приёмами написания программы для решения стандартной задачи с использованием основных конструкций программирования и отладки таких программ на языке Python;
- знать место языка Python среди языков программирования высокого уровня,
- знать особенности структуры программы, представленной на языке Python,
- иметь представление о модулях, входящих в состав среды Python,
- знать возможности и ограничения использования готовых модулей,
- иметь представление о величине, ее характеристиках,
- знать, что такое операция, операнд и их характеристики,
- знать принципиальные отличия величин, структурированных и не структурированных,
- иметь представление о таких структурах данных, как число, текст, кортеж, список, словарь,
- иметь представление о составе арифметического выражения;
- знать математические функции, входящие в Python, иметь представление о логических выражениях и входящих в них операндах, операциях и функциях,
- уметь записывать примеры арифметических и логических выражений всех атрибутов, которые могут в них входить,
- знать основные операторы языка Python, их синтаксис,
- иметь представление о процессе исполнения каждого из операторов,
- уметь разрабатывать программы обработки числовой и символьной информации,
- уметь разрабатывать программы (линейные, разветвляющиеся и с циклами),
- иметь представление о значении полноценных процедур и функций для структурно-ориентированного языка высокого уровня,
- знать правила описания функций в Python и построение вызова,
- знать принципиальные отличия между формальными, локальными и глобальными переменными.

### **3. Тематическое планирование**

№ п/п	Название раздела, темы	Количество часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
1	Введение в Python.	2	1	1	тест
2	Переменные и выражения. Операции. Организация ввода и вывода данных.	6	1	5	Выполнение практического задания

3.	Понятие алгоритм. Типы алгоритмов. Линейный алгоритм и алгоритм с условием. Условный оператор.	8	2	6	Выполнение практического задания
4.	Управляющие инструкции – циклы.	16	6	10	Выполнение практического задания Тест
5.	Создание итогового проекта	2		2	Защита проекта
Итого		34	10	24	

### 1.1. Промежуточный контроль

Задание 1

#### ВВОД И ВЫВОД ДАННЫХ

Программа запрашивает у пользователя имя и фамилию, после чего выводит приветственное сообщение.

На входе: две строковые переменные

На выходе: строка

Например:

```
>>>Ваше имя? Иван
```

```
>>>Ваша фамилия? Петров
```

```
Здравствуйте, Петров Иван!
```

Задание 2

#### РАЗВЕТВЛЯЮЩИЙСЯ АЛГОРИТМ

Программа находит минимальное значение из трёх.

На входе: три целых числа.

На выходе: одно целое число.

Пример:

```
>>> Введите первое число: 1
```

```
>>> Введите второе число: 2
```

```
>>> Введите третье число: 6
```

```
Минимальное число: 1
```

Задание 3

#### ЦИКЛИЧЕСКИЕ АЛГОРИТМЫ

Напишите программу, выводящую на экран все числа от А до В включительно.

Пояснение:  $A \leq B$ .

На входе: в первой строке вводится А, во второй вводится В.

На выходе: последовательность целых чисел, разделённых пробелом.

Например:

```
>>>Введите А: -3
```

```
>>>Введите В: 5
```

```
-3 -2 -1 0 1 2 3 4 5
```

### Тест «Язык программирования. Ветвления. Сложные условия. Циклы. Циклы с условиями. Функции»

1. PYTHON является:

Варианты ответов

- Машинно - ориентированным языком (низкого уровня)
- Языком высокого уровня
- Объектно - ориентированным языком

2. Область применения PYTHON:

Варианты ответов

- Робототехника и искусственный интеллект
- Обучение
- Интернет

3. Год разработки PYTHON:

Варианты ответов

- 1990
- 1991
- 1993

4. Чувствителен ли PYTHON к регистру (большая или маленькая буквы):

Варианты ответов

- Да
- Нет

5. Какие существуют типы переменных (выбрать несколько вариантов):

Варианты ответов

- float
- list
- num
- int
- bool
- integer

6. Переменная int:

Варианты ответов

- вещественная переменная
- символьная строка
- логическая переменная
- целая переменная

7. Переменная str:

Варианты ответов

- символьная строка
- логическая переменная
- целая переменная

8. Переменная float:

Варианты ответов

- целая переменная
- вещественная переменная
- логическая переменная

9. Каков будет результат выполнения `int("88")`:

Варианты ответов

- "88"
- 88
- 88.00

10. Каков будет результат выполнения `str(88)`:

Варианты ответов

- "88"
- 88
- 88.00

11. Имена переменных не могут включать:

Варианты ответов

- Русские буквы
- Латинские буквы
- Пробелы
- Скобки, знаки + = ! ? b др.

12. Какие имена являются правильными в PYTHON (выбрать несколько):

Варианты ответов

- N
- ABC
- sum
- 41And
- A+B
- \_mam

13. Что будет в результате выполнения команды:

```
a = 20
```

```
b = a + 5
```

```
a = b * 100
```

```
print(a)
```

Варианты ответов

- 25
- 2500
- 25000
- 1000

14. Что будет в результате следующего действия `print(2**20)`

Варианты ответов

- 104576
- 1048576
- 964
- 2

15. Что будет в результате выполнения следующего действия `print(23 % 2)`

Варианты ответов

- 11
- 1
- 0

16. Результатом вычисления `print(24 // 3)` будет число:

Варианты ответов

- 4
- 8
- 12

17. Что будет результатом выполнения алгоритма:

```
a = 5
```

```
b = 7
```

```
a = int(input())
```

```
b = int(input())
```

```
s = a + b
```

```
print(s)
```

Варианты ответов

- 57
- 12
- 35

18. Что будет результатом выполнения алгоритма:

```
a = 5
```

```
b = 7
```

```
a = input()
```

```
b = input()
```

```
s = a + b
```

```
print(s)
```

Варианты ответов

- 12
- 57
- 35

19. Что будет в результате выполнения следующего алгоритма:

Входные данные: -57

```
x = int(input())
```

```
if x > 0:
```

```
    print(x)
```

```
else:
```

```
    print(-x)
```

Варианты ответов

- -57
- 57
- 0
- -1

20. Что будет в результате выполнения программы:

Входные данные:

```
10
```

```
20
```

```
a = int(input())
```

```
b = int(input())
```

```
if a < b:
```

```
    print(a)
```

```
else:
```

```
    print(b)
```

Варианты ответов

- 10
- 20
- 30
- -10

21. Какой ряд чисел образуется после выполнения следующего алгоритма:

```
for i in range(1,10):
```

```
    print(i)
```

Варианты ответов

- 1 2 3 4 5 6 7 8 9 10



- 1 2 3 4 5 6 7 8 9
- 0

22. Какой ряд чисел образуется после выполнения алгоритма:  
for i in range(1,10+1):  
print(i)

Варианты ответов

- 1 2 3 4 5 6 7 8 9 10
- 1 2 3 4 5 6 7 8 9 10 11
- 1 4 9 16

23. Что выведет программа после выполнения данного алгоритма:

Входные данные: Иванов

```
print('Как Ваша фамилия?')
```

```
name = input()
```

```
print('Здравствуйте, '+ name + '!')
```

Варианты ответов

- Как Ваша фамилия? Здравствуйте, Иванов!
- Как Ваша фамилия? Здравствуйте, Иванов
- Как Ваша фамилия? Здравствуйте, Иванов !

24. Как обозначается логический оператор И, ИЛИ, НЕ в питоне:

Варианты ответов

- OR, NOT, IF
- AND, OR, NOT
- AND, OR, IF
- AND, ELSE, NOT

25. Что будет в результате выполнения следующего алгоритма программы:

Входные данные:

15

45

```
a = int(input())
```

```
b = int(input())
```

```
if a % 10 == 0 or b % 10 == 0:
```

```
print('YES')
```

```
else:
```

```
print('NO')
```

Варианты ответов

- YES
- NO

26. Как будет записано число 18 после выполнения следующего алгоритма:

```
x = float(input())
```

```
print(x)
```

Варианты ответов

- 18
- 18.0
- 18.00

27. Что будет после выполнения следующего листинга программы:  
for i in range(4)

```
print(i)
```

```
print(i ** 2)
```

Варианты ответов

- 0 0 1 1 3 3 4 4

- 0 0 1 1 2 4 3 4

- 0 0 1 1 2 3 3 9

28. Результатом выполнения алгоритма цикла while будет:

```
i = 1
```

```
while i <= 10:
```

```
print(i ** 2)
```

```
i = i + 1
```

Варианты ответов

- 1 2 4 8 12 14

- 1 2 16 24 32

- 1 2 4 16 25 36 49 64 81 10

## 1.2. Итоговый контроль:

### Критерии оценки проекта:

Оценка работ проводится с учетом критериев

—Актуальность проекта и практическая применимость (от 0 до 2 баллов). Критерий оценивает соответствие темы проекта прикладным проблемам, перспективы возможного практического использования результатов проекта.

—Новизна и сложность проекта (от 0 до 3 баллов). Критерий оценивает новизну выбранного в проекте подхода (расширенная постановка лабораторных работ; самостоятельно изученные задачи, алгоритмы и технологии), сложность используемых алгоритмов и т.п.

—Сложность программной разработки (от 0 до 6 баллов). Критерий оценивает сложность выполненной программной разработки (используемые технологии, программная архитектура, объем программного кода, использование внешних библиотек, уровень выполненного тестирования).

—Качество пользовательского интерфейса (от 0 до 2 баллов). Критерий оценивает дружелюбность пользовательского интерфейса (привлекательность, доступность, простота использования и т.п.) и наличие наглядной демонстрации принципов работы реализованных алгоритмов.

—Уровень проектного выступления (от 0 до 4 баллов). Критерий оценивает качество подготовленной проектной презентации и уровень выступления с представлением проекта (соблюдение регламента, понятность изложения основных результатов проекта, ответы на вопросы и т.п.).

## Календарно - тематическое планирование

№ п/п	Название раздела, темы	Количество часов	Дата		
			Всего	План	факт
1	Тема 1. Введение в Python.	2			
1.1.	История языков программирования. Язык программирования Python. Структура программы в языке Python. Создание и запуск простых программ	2			
2.	<b>Переменные и выражения. Операции. Организация ввода и вывода данных.</b>	6			
2.1	Типы данных и функции вывода. Определение переменной. Переменные и арифметические выражения.	2			
2.2.	Вычисления и переменные. Операторы print() и input() и их параметры.	1			
2.3	Арифметические операции. Целочисленное деление и остаток от целочисленного деления.	2			
2.4	Создание и отладка программы «Привет, мир!»	1			
3.	<b>Понятие алгоритм. Типы алгоритмов. Линейный алгоритм и алгоритм с условием. Условный оператор.</b>	8			
3.1.	Синтаксис языка Python. Логические выражения и операторы.	1			
3.2.	Генерирование случайных чисел. Условная конструкция if.	1			
3.3	Программирование линейных алгоритмов. Решение задач	2			
	Конструкции if...else. Использование elif. Составные условия.	2			
3.4	Программирование разветвляющихся алгоритмов. Решение задач	2			
4.	<b>Управляющие инструкции – циклы.</b>	16			
4.1.	Генерация случайных чисел. Цикл while.	1			
4.2	Решение задач с циклом while.	1			
4.3	Создание и отладка программы «Угадай число».	1			
4.4	Циклы for.	2			

4.5	Команды break и continue. Вложенные циклы. Бесконечный цикл.	1		
4.6	Решение задач различного типа с помощью циклических алгоритмов.	1		
4.7	Встроенные функции.	1		
4.8	<i>Создание игры «Волк, коза и капуста» с использованием циклического алгоритма.</i>	1		
4.9	Работа со строками: индексация, срезы.	1		
4.10	Кортежи	1		
4.11	Создание игры «Анаграммы»	1		
4.12	Списки. Списочные методы	1		
4.13.	Словари	1		
4.14.	Использование словарей	1		
4.15	Создание игры «Виселица»	1		
5.	<b>Создание итогового проекта</b>	2		
<b>Итого</b>		<b>34</b>		