



# ПРОГРАММИРОВАНИЕ ЛИНЕЙНЫХ АЛГОРИТМОВ

## НАЧАЛА ПРОГРАММИРОВАНИЯ

**8 класс**

# Вывод данных

Вывод данных из оперативной памяти на экран монитора:

```
print(<выражение 1> ,< выражение 2> , ..., < выражение N>)
```

СПИСОК ВЫВОДА

**Выражения** - символьные, числовые, логические, в том числе переменные и константы

**Пример:**

```
s = 10
```

```
print ('s=',s)
```

**На экране:**

```
s= 10
```



**Информация в кавычках  
выводится на экран без  
изменений**

# Варианты организации вывода

Вариант организации вывода	Оператор вывода	Результат
По умолчанию	<code>print (1, 20, 300)</code>	1 20 300
Убрать разделители — пробелы	<code>print (1, 20, 300, sep="")</code>	120300
Добавить другой разделитель	<code>print (1, 20, 300, sep=", ")</code>	1, 20, 300

# Формат вывода

**Формат вывода** позволяет установить количество позиций на экране, занимаемых выводимой величиной.

d - вывод целого значения

f – вывод вещественного значения

Оператор вывода	Результат выполнения оператора
<pre>a = 4 print ("a=", "{:5d} {:5d}".format(a, a*a))</pre>	a=○○○○4○○16
<pre>a = 1/3 b = 1/9 print ("{:7.3f} {:7.3f}".format(a, b))</pre>	○○0.333○○0.111

# Ввод данных с клавиатуры

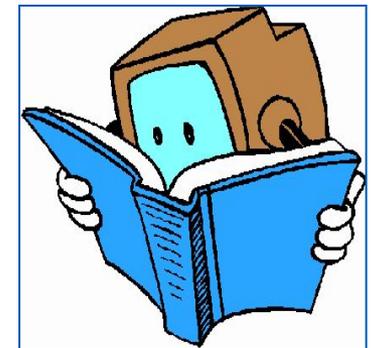
Ввод в оперативную память значений переменных :

```
<имя переменной> = input()
```

Переменная будет строкового типа (str)

Выполнение оператора **input**:

- 1) компьютер переходит в режим ожидания данных:
- 2) пользователь вводит данные с клавиатуры:  
- значение переменной
- 3) пользователь нажимает клавишу **Enter**.



# Ввод данных с клавиатуры

Ввод переменных целого типа:

```
<имя переменной> = int (input())
```

*Переменная из строковой сразу преобразуется в целую*

Ввод переменных вещественного типа (float):

```
<имя переменной> = float (input())
```

*Переменная из типа str сразу преобразуется в тип float*

Ввод переменных с приглашением:

```
n = int (input ("введи n:"))
```

# Числовые типы данных

Стандартные функции языка Python:

Функция	Назначение	Тип аргумента	Тип результата
<code>abs(x)</code>	Модуль $x$	int, float	Такой же, как у аргумента
<code>round(x)</code>	Округление вещественного $x$ до заданного количества знаков после запятой	float	int, float
<code>int(x)</code>	Преобразование вещественного или строкового $x$ к целому	str, float	int

Прочие математические функции собраны в модуле `math`, подключение модуля:

**`from math import *`**

<code>sqrt(x)</code>	Квадратный корень из $x$	int, float	float
----------------------	--------------------------	------------	-------

Датчик случайных чисел: **`from random import *`**

<code>randint(a,b)</code>	Целое случайное число в интервале $[a, b]$	int	int
<code>random ()</code>	Вещественное случайное число в интервале $[0, 1)$		float

# Целочисленный тип данных

Операции над целыми числами:

Операция	Обозначение	Тип результата
Сложение	+	int
Вычитание	-	int
Умножение	*	int
Возведение в степень	**	int
Получение целого частного	//	int
Получение целого остатка деления	%	int
Деление	/	float

# Операции // и %

Трёхзначное число можно представить в виде следующей суммы:  $x = a \cdot 100 + b \cdot 10 + c$ , где  $a, b, c$  - цифры числа.

Программа нахождения суммы цифр вводимого с клавиатуры целого трёхзначного числа:

```
print ('Нахождение суммы цифр трёхзначного числа');  
x = int (input ('Введите исходное число>>'))  
a = x // 100  
b = x % 100 // 10  
c = x % 10  
s = a + b + c  
print ('s= ', s)
```

# Вопросы и задания

## Для 8 «Ж» и 8 «И»

Запишите на языке Python программу, которая для произвольного трехзначного числа определяет:

а) сумму и произведение его цифр(разрядов);

## Для 8 «З»

Запишите на языке Python программу, которая в прямоугольнике со сторонами(a-случайное целое число и b- случайное вещественное число), находит его периметр, площадь и размер диагонали(если надо числа все округлить до 2х знаков после запятой)