Программирование на языке Python

§ 54. Арифметические выражения
§ 55. Вычисления

© К.Ю. Поляков, Е.А. Ерёмин, 2014

Типы данных

- `int  # целое`
- `float  # вещественное`
- `bool  # логические значения`
- `str  # символьная строка`

```
a = 5
print ( type(a) )
a = 4.5
print ( type(a) )
a = True
print ( type(a) )
a = "Вася"
print ( type(a) )
```

```
<class 'int'>
<class 'float'>
<class 'bool'>
<class 'str'>
```

Арифметическое выражения

```
       3    1   2   4       5   6
a = (c + b**5*3 - 1) / 2 * d
```

Приоритет (*старшинство*):

1) скобки
2) возведение в степень **
3) умножение и деление
4) сложение и вычитание

$$a = \frac{c + b^5 \cdot 3 - 1}{2} \cdot d$$

```
a = (c + b*5*3 - 1) \
                / 2 * d
```

перенос на следующую строку

```
a = (c + b*5*3
                - 1) / 2 * d
```

перенос внутри скобок разрешён

Деление

Классическое деление:

```
a = 9; b = 6
x = 3 / 4     # = 0.75
x = a / b     # = 1.5
x = -3 / 4    # = -0.75
x = -a / b    # = -1.5
```

Целочисленное деление (округление «вниз»!):

```
a = 9; b = 6
x = 3 // 4     # = 0
x = a // b     # = 1
x = -3 // 4    # = -1
x = -a // b    # = -2
```

Остаток от деления

% – остаток от деления

```
d = 85
b = d // 10
a = d % 10
d = a % b
d = b % a
```

Для отрицательных чисел:

```
a = -7
b = a // 2   # -4
d = a % 2    # 1
```

! Как в математике!

остаток ≥ 0

-7 = (-4)*2 + 1

Сокращенная запись операций

```
a += b   # a = a + b
a -= b   # a = a - b
a *= b   # a = a * b
a /= b   # a = a / b
a //= b  # a = a // b
a %= b   # a = a % b
```

```
a += 1
```

увеличение на 1

Вещественные числа

Целая и дробная части числа разделяются **точкой**!

Форматы вывода:

```
x = 123.456
print( x )
print("{:10.2f}".format(x))
```

```
123.456
    123.46
```

всего знаков

в дробной части

```
print("{:10.2g}".format(x))
```

```
   1.2e+02
```

значащих цифр

$1{,}2 \cdot 10^2$

Стандартные функции

abs(x) – модуль числа

int(x) – преобразование к целому числу

round(x) – округление

```
x = abs( -1.6 )      # 1.6
x = int( -1.6 )      # -1
x = round( -1.6 )    # -2
```

bin(x) – в двоичную систему

oct(x) – в восьмеричную систему

hex(x) – в шестнадцатеричную систему

```
x = bin( 29 )   # '0b11101'
x = oct( 29 )   # '0o35'
x = hex( 29 )   # '0x1d'
```

Математические функции

```
import math
```

подключить математический модуль

```
math.pi        – число «пи»
math.sqrt(x)   – квадратный корень
math.sin(x)    – синус угла, заданного в радианах
math.cos(x)    – косинус угла, заданного в радианах
math.exp(x)    – экспонента e^x
math.ln(x)     – натуральный логарифм
math.floor(x)  – округление «вниз»
math.ceil(x)   – округление «вверх»
```

```
x = math.floor(1.6)  # 1
x = math.ceil(1.6)   # 2
```

```
x = math.floor(-1.6)  #-2
x = math.ceil(-1.6)   #-1
```

Документирование программы

```
from math import sqrt
print("Введите a, b, c:")
a, b, c = map(float, input().split())
D = b*b - 4*a*c
if D < 0:
  print("Нет")
else:
  x1 = (-b + sqrt(D))/(2*a)
  x2 = (-b - sqrt(D))/(2*a)
  print("x1={:5.3f} x2={:5.3f}".format(
                                x1, x2))
```

Что делает?

Случайные числа

Случайно…

- встретить друга на улице
- разбить тарелку
- найти 10 рублей
- выиграть в лотерею

Случайный выбор:

- жеребьевка на соревнованиях
- выигравшие номера в лотерее

Как получить случайность?

Случайные числа на компьютере

Электронный генератор

- нужно специальное устройство
- нельзя воспроизвести результаты

Псевдослучайные числа – обладают свойствами случайных чисел, но каждое следующее число вычисляется по заданной формуле.

Метод середины квадрата (Дж. фон Нейман)

зерно **564321**

в квадрате

- малый период (последовательность повторяется через 10^6 чисел)

318458191041

209938992481

Генератор случайных чисел

```
import random
```

англ. *random* – *случайный*

Целые числа на отрезке [a,b]:

```
X = random.randint(1,6) # псевдосл. число
Y = random.randint(1,6) # уже другое!
```

Генератор на [0,1):

```
X = random.random()  # псевдослучайное число
Y = random.random()  # это уже другое число!
```

Генератор на [a, b] (вещественные числа):

```
X = random.uniform(1.2, 3.5)
Y = random.uniform(1.2, 3.5)
```

Генератор случайных чисел

```
from random import *
```

подключить все!

Целые числа на отрезке [a,b]:

```
X = randint(10,60) # псевдослучайное число
Y = randint(10,60) # это уже другое число!
```

Генератор на [0,1):

```
X = random(); # псевдослучайное число
Y = random()  # это уже другое число!
```

Генератор на [a, b] (вещественные числа):

```
X = uniform(1.2, 3.5) # псевдосл. число
Y = uniform(1.2, 3.5) # уже другое число!
```

Задание

Выполнить до 08.04.2020 17.00

© К.Ю. Поляков, Е.А. Ерёмин, 2014 http://kpolyakov.spb.ru

Задание в тетради

Выполнить задание в тетради, сфотографировать и отправить фото

4. Запишите оператор для вывода значений целых переменных $a = 5$ и $b = 3$ в следующем формате:

а) `3+5=?`

б) `Z(5)=F(3)`

в) `a=5; b=3;`

г) `Ответ: (5;3)`

© К.Ю. Поляков, Е.А. Ерёмин, 2014 http://kpolyakov.spb.ru

Задание в программе

1. Три задачи ниже, необходимо выполнить в программе на языке Python. Для компьютера рекомендую программу **Thonny,** для смартфонов **Pydroid3.** Так же можно использовать другие программы, которые поддерживают Python.

2. Готовый файл и скриншот с тестами работы программы (хотя бы запуск данных из примеров) прислать мне через элжур.

Задачи в программе

«A»: Ввести с клавиатуры три целых числа, найти их сумму, произведение и среднее арифметическое.

Пример:

```
Введите три целых числа:
5 7 8
5+7+8=20
5*7*8=280
(5+7+8)/3=6.667
```

«B»: Ввести с клавиатуры координаты двух точек (A и B) на плоскости (вещественные числа). Вычислить длину отрезка AB.

Пример:

```
Введите координаты точки A:
5.5 3.5
Введите координаты точки B:
1.5 2
Длина отрезка AB = 4.272
```

Задачи

«С»: Получить случайное трехзначное число и вывести через запятую его отдельные цифры.

Пример:

```
Получено число 123.
Его цифры 1, 2, 3.
```