

Руководство по программированию



Python3

Майкоп

2017

ЧАСТЬ 1 ВВОД И ВЫВОД

Как не кажется, банально, но начать стоит с самой простой программы!

Пример программы

```
print('Hello, world!')
```

Результат работы

```
Hello, world!
```

`print()` – это функция в Python, которая занимается выводом. Все, что вам необходимо вывести, как результат работы вашей программы вам необходимо поместить внутри круглых скобок через запятую.

Пример программы

```
print("s =", 10+3)
```

Результат работы

```
s = 13
```

Как видно из примеров выводить можно не только комментарии, заключенные между знаками апостроф 'комментарий' или кавычки "комментарий" но и арифметические действия.

Арифметические действия в Python

Пример программы	Результат работы	Операция
<pre>print(10+3)</pre>	13	сложение
<pre>print(10-3)</pre>	7	вычитание
<pre>print(10*3)</pre>	30	умножение
<pre>print(10/3)</pre>	3.3333333333333335	деление
<pre>print(10**3)</pre>	1000	возведение в

<code>print(10//3)</code>	<code>3</code>	степень целая часть от деления
<code>print(10%3)</code>	<code>1</code>	Остаток от деления нацело

Для ввода данных в Python используют функцию `input()`. Эта функция читает данные из строки и может ассоциировать их с переменными.

Пример программы

```
a=input('введите ваше имя ')
print('Привет ',a)
#после запуска программы на экране появиться
#приглашение для ввода данных в виде
#введите ваше имя
#если вы введете, например Василий
#результат будет таким
```

Результат работы

```
Привет  Василий
```

При этом если мы не указываем тип считываемых данных, python считает их строкой, а значит, арифметические действия либо не будут выполнены, либо будут выполнены для строковых переменных. В примере выше `a` является переменной, а знак `"="` – присваивание.

Типы данных

Класс данных	Хранимые данные
str	строки
int	целые числа
float	вещественные числа
bool	логический тип, хранит два значения False (ложь) и True (истина)

Для решения задач, необходимо понимать какого типа будут входные данные и каким будет результат работы алгоритма.

Введем два целых числа с клавиатуры и найдем их произведение.

Пример программы

```
a=int(input()) #ввод первого целого числа
b=int(input()) #ввод второго целого числа
print(a*b)     #вывод произведения этих чисел
```

Результат работы

```
5          #первое вводимое число
10         #второе вводимое число
50         #вывод результата
```

В примере выше конструкция `int(input())` читает целое число, на что указывает `int()`.

Поскольку Python очень гибкий язык программирования то программу решения предыдущей задачи можно записать в виде так называемой однострочной программы.

Пример программы

```
print(int(input())*int(input())) #работает аналогично
```

#предыдущей программе

Результат работы

5 #первое вводимое число

10 #второе вводимое число

50 #вывод результата

ЗАДАЧИ

Дано три целых числа, напишите программу, вычисляющую и выводящую сумму, произведение и среднее арифметическое этих чисел.

Входные данные	Выходные данные
2	$s = 9$
3	$p = 24$
4	$sa = 3.0$

1. Определить площадь прямоугольного треугольника, если известны длины его катетов. Известно, что катеты треугольника вещественные числа.

Входные данные	Выходные данные
2.4	$s = 4.32$
3.6	

2. Дано четырехзначное число. Определить сумму и произведение цифр этого числа.

Входные данные	Выходные данные
1234	$s = 10$
	$p = 24$

3. На вход программы подаётся одно целое число, напишите программу выводящую результат извлечения корня квадратного из этого числа.

Входные данные	Выходные данные
25	5.0

4. На вход программы подается одно целое число.

Напишите программу выводящая следующее по порядку нечетное число идущее за данным.

Входные данные	Выходные данные
10	11
Входные данные	Выходные данные
11	13

РЕШЕНИЕ ЗАДАЧ

Задача 1.

```
a=int(input())
b=int(input())
c=int(input())
print('s =',a+b+c)
print('p =',a*b*c)
print('sa =', (a+b+c)/3)
```

Задача 2.

```
a=float(input())
b=float(input())
print('s =',a*b/2)
```

Задача 3.

```
x=int(input())
a=x//1000
b=(x%1000)//100
c=(x%100)//10
d=x%10
print('s =',a+b+c+d)
print('p =',a*b*c*d)
```

Задача 4.

```
print(int(input())**(1/2))
```

Задача 5.

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

```
print(a+(1+a%2))
```


ЧАСТЬ 2 РАБОТА СО СТРОКАМИ

Если арифметические действия и работа с числовыми переменными обычно не вызывает больших затруднений, то работа со строковыми переменными не всегда очевидны. Рассмотрим базовые операции со строками.

Пример программы

```
a='Дарт'  
b='Вейдер'  
c=a+b  
print(c) #сумма строк  
print(a,b)#вывод строк
```

Результат работы

```
ДартВейдер  
Дарт Вейдер
```

Каждая строка это набор символов стоящие на определенном месте, первый символ строки имеет номер 0, второй 1 и т.д. Но в отличие от большинства языков программирования последовательности в Python имеют две нумерации, прямую и обратную.

	-6	-5	-4	-3	-2	-1	обратная
	П	Р	И	В	Е	Т	
прямая	0	1	2	3	4	5	

Давайте рассмотрим порядок обращения к строкам.

Пусть на входе дана строка `s='Hello, world!'`.

Пример программы	Результат работы	Описание
<code>print(s)</code>	Hello, world!	Вывод всей строки

<code>print(s[0])</code>	<code>H</code>	Вывод первого элемента
<code>print(s[-1])</code>	<code>!</code>	Вывод последнего элемента
<code>print(s[2:])</code>	<code>llo, world!</code>	Вывод со второго элемента до конца строки
<code>print(s[2:5])</code>	<code>llo</code>	Вывод подстроки со второго по четвертый элемент
<code>print(s[::2])</code>	<code>Hlo ol!</code>	Вывод всей строки начиная с нулевой позиции с шагом 2
<code>print(s[::-1])</code>	<code>!dlrow ,olleH</code>	Вывод строки в обратном порядке
<code>print(len(s))</code>	<code>13</code>	Вывод длины строки

ЗАДАЧИ

1. Дана строка. Вывести первый символ строки, вывести последний символ строки, вывести подстроку с номера `a` по номер `b`, вывести строку в обратном порядке, определить длину строки.

Входные данные	Выходные данные
<code>abcdefg</code>	<code>a</code>
<code>2 5</code>	<code>g</code>
	<code>cdef</code>
	<code>gfedcba</code>
	<code>7</code>

РЕШЕНИЕ ЗАДАЧ

Задача 1.

```
s=str(input())
a,b=map(int,input().split())
print(s[0])
print(s[-1])
print(s[a:b+1])
print(s[::-1])
print(len(s))
```

ЧАСТЬ 3 УСЛОВНЫЙ ОПЕРАТОР

Все ранее рассмотренные примеры являются линейными алгоритмами, часто необходимо организовать проверку какого либо критерия для решения задачи, для этого и существуют условные конструкции.

Пример программы

```
a=int(input())
if a>0:
    print(1)
else:
    print(-1)
```

Результат работы

```
#при вводе положительного числа (например, 3)
1      # вывод результата
#при вводе отрицательного числа (например, -3)
-1     # вывод результата
```

Если в задачах необходимо проверять более одного критерия в одной задаче, можно использовать несколько вложенных условных структур или структуру elif.

Пример программы

```
a=int(input())
if a>0 and a<5:
    print('YES')
elif a>10 and a<15:
    print('YES')
else:
    print('NO')
```

Результат работы

#при вводе числа из промежутков (0, 5) или (10, 15)

YES # вывод результата

#при вводе любых других значений

NO # вывод результата

Давайте рассмотрим операторы сравнения.

Оператор	Описание
<	Меньше - условие верно, если значение слева меньше значению справа
>	Больше - условие верно, если значение слева больше значению справа
<=	Меньше или равно - условие верно, если значение слева меньше или равно значению справа
>=	Больше или равно - условие верно, если значение слева больше или равно значению справа
==	Равно - условие верно, если значения слева и справа совпадают
!=	Не равно - условие верно, если значения слева и справа не совпадают

Для объединения нескольких условий в одну структуру используют логические операторы, давайте рассмотрим их.

Оператор	Описание
and	И - условие верно, если условия слева и справа истинные

or	ИЛИ - условие верно, если хотя бы одно из условий слева и справа истинно
not	НЕ - условие верно, если отрицаемое условие ложно

ЗАДАЧИ

1. Даны два положительных числа, вывести наибольшее из них.

Входные данные	Выходные данные
2 5	5

2. Даны три неотрицательных числа вывести наименьшее из них.

Входные данные	Выходные данные
3 1 5	1

3. Определить к какой координатной четверти относиться точка с координатами (x,y).

Входные данные	Выходные данные
2 5	1 четверть

РЕШЕНИЕ ЗАДАЧ

Задача 1.

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

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

```
if a>b:
    print(a)
else:
    print(b)
```

Задача 2.

```
a=int(input())
b=int(input())
c=int(input())
if a<=b and a<=c:
    print(a)
elif b<=a and b<=c:
    print(b)
else:
    print(c)
```

Задача 3.

```
x,y = map(int,input().split())
if x>0 and y>0:
    print('1 четверть')
elif x<0 and y>0:
    print('2 четверть')
elif x<0 and y<0:
    print('3 четверть')
else:
    print('4 четверть')
```

ЧАСТЬ 4 ЦИКЛИЧЕСКИЕ СТРУКТУРЫ (ЦИКЛ FOR)

Часто в задачах необходимо повторять одни и те же действия определенное количество раз, в этом нам и помогут циклические структуры.

Пример программы

```
for i in 1, 2, 4.5, 'Hello':  
    print(i)
```

Результат работы

```
1  
2  
4.5  
Hello
```

В примере выше показано, что не в зависимости от типа данных в наборе переменной *i*, все значения перебираются по очереди и выводятся без ошибок. Это еще один плюс динамической типизации Python.

Для перебора целочисленных значений используют функцию `range`.

Пример программы

```
for i in range(5):  
    print(i)
```

Результат работы

```
0  
1  
2  
3  
4
```


Из примера выше видно, что цикл работает ровно 5 раз, при этом выводятся значения от 0 до 4. Об этом надо помнить при работе с функцией `range`.

Запись функции	Описание
<code>range (n)</code>	Будет перебирать значение чисел от 0 до $n-1$, с шагом 1
<code>range (a ,b)</code>	Будет перебирать значение чисел от a до $b-1$, с шагом 1
<code>range (a ,b ,3)</code>	Будет перебирать значение чисел от a до $b-1$, с шагом 3
<code>range (a ,b , -2)</code>	Будет перебирать значение чисел от a до $b-1$, с шагом -2 , при условии, что $a > b$

ЗАДАЧИ

1. Вычислить сумму чисел от 1 до вводимого с клавиатуры числа n .

Входные данные	Выходные данные
5	15

2. Даны два числа a и b ($a < b$), вычислить произведение четных чисел в диапазоне от a до b включительно.

Входные данные	Выходные данные
2 5	8

3. Даны два положительных числа a и b , вывести в строку через пробел все числа кратные 3 в диапазоне от a до b включительно.

Входные данные	Выходные данные
2 7	3 6

РЕШЕНИЕ ЗАДАЧ

Задача 1.

```
n=int(input())
s=0
for i in range(1,n+1):
    s+=i
print(s)
```

Задача 2.

```
a,b=map(int,input().split())
s=1
for i in range(a,b+1):
    if i%2==0:
        s*=i
print(s)
```

Задача 3.

```
a,b=map(int,input().split())
s=1
for i in range(a,b+1):
    if i%3==0:
        print(i,end=' ')
```

ЧАСТЬ 5 ЦИКЛИЧЕСКИЕ СТРУКТУРЫ (ЦИКЛ WHILE)

Цикл `while` в отличие от цикла `for` не проходит определенное количество шагов, он проверяет истинность условия и работает, пока оно возвращает истину.

Пример программы

```
a=1
while a<10:
    print(a, end=' ')
    a+=1
```

Результат работы

1 2 3 4 5 6 7 8 9

В данном примере показано, что значения `a` будут выводиться на экран до тех пор, пока условие `a<10` истинно. При этом надо не забыть про изменение параметра `a`, иначе мы получим бесконечный цикл.

ЗАДАЧИ

1. Дано число `n`, вывести все его делители, кроме 1 и `n`.

Входные данные	Выходные данные
15	3 5

2. Даны два числа `a` и `b` ($a \neq b$), вычислить наибольший общий делитель этих чисел.

Входные данные	Выходные данные
6 15	3

3. Даны два положительных числа `a` и `b`, вывести в строку через пробел все числа кратные 3 и оканчивающиеся на 1.

Входные данные	Выходные данные
----------------	-----------------

5 30	21
------	----

РЕШЕНИЕ ЗАДАЧ

Задача 1.

```
n=int(input())
a=2
while a < n//2:
    if n%a==0:
        print(a,end=' ')
    a+=1
```

Задача 2.

```
a,b=map(int,input().split())
while a != b:
    if a>b:
        a=a-b
    else:
        b=b-a
print(a)
```

Задача 3.

```
a,b=map(int,input().split())
while a <= b:
    if a%3==0 and a%10==1:
        print(a)
    a+=1
```

Тестовые задания для самоконтроля (Тест №1)

1. $5+7 =$ _____
2. $'5'+7 =$ _____
3. $'5'+'7' =$ _____
4. $5*7 =$ _____
5. $'5'*7 =$ _____
6. $'5'*'7' =$ _____
7. $5-7 =$ _____
8. $7/5 =$ _____
9. $7//5 =$ _____
10. $7\%5 =$ _____
11. $5>7 =$ _____
12. $5<7 =$ _____
13. $\text{not}(5>7) =$ _____
14. $8\&4 =$ _____
15. $8|4 =$ _____
16. $\text{bin}(45) =$ _____
17. $\text{oct}(45) =$ _____
18. $\text{hex}(45) =$ _____
19. $\text{int}(7>5) =$ _____
20. $\text{int}(\text{False}) =$ _____

Тестовые задания для самоконтроля (Тест №2)

1. Определите результат работы программы:

```
x=234
a=x//100
b=(x//10)%10
c=x%10
print(a+b+c)
```

Ответ: _____

2. При каком значении переменной x, программа выводит 6.0?

```
x=int(input())
a=x//100
b=(x//10)%10
c=x%10
print((a+b+c)/3)
```

Ответ: _____

3. Определите результат работы программы:

```
x=234
a=x//10
b=x%10
print(a-b)
```

Ответ: _____

4. При каком значении переменной x, программа выводит 6?

```
x=int(input())
a=x//10
```

```
b=x%10
print(a-b)
```

Ответ: _____

5. Определите результат работы программы:

```
x=1234
s=0
for i in str(x):
    s+=int(i)
print(s)
```

Ответ: _____

6. При каком значении переменной x, программа выводит 3.0?

```
x=int(input())
s=0
for i in str(x):
    s+=int(i)
print(s/2)
```

Ответ: _____

7. Определите результат работы программы:

```
x=4
y=8
s=0
for i in range(x, y+1):
    s+=i
print(s)
```

Ответ: _____

8. При каких значениях переменных x и y, программа выводит 10?

```
x=int(input())
y=int(input())
s=0
for i in range(x,y+1):
    s+=1
print(s)
```

Ответ: _____

9. Определите значение работы каждой строки:

```
x='starwars'
```

```
print(x[2])
```

```
print(x[-2])
```

```
print(x[::2])
```

```
print(x[::-1])
```

```
print(x[2:4])
```

Тестовые задания для самоконтроля (Тест №3)

1. Определите результат работы программы:

```
a=120; b=a*2; a=a+b; print(a+b)
```

Ответ: _____

2. При каких значениях переменных a и b, программа выводит 25?

```
a,b=map(int, input().split()); a=a+b; print(a+b)
```

Ответ: _____

3. Определите результат работы программы:

```
a=3;b=12;s=0
for i in range(a,b):s+=i
print(s)
```

Ответ: _____

4. Определите результат работы программы:

```
for i in range(2,8):print(i,end=' ')
```

Ответ: _____

5. Определите результат работы программы:

```
for i in range(2,8):print(5,end=' ')
```

Ответ: _____

6. Определите результат работы программы:

```
print('YES') if 6>5 else print('NO')
```

Ответ: _____

7. При каких значениях переменных a, b и c, программа выводит, “существует”?

```
a,b,c=map(int, input().split())
if a+b>c and b+c>a and c+a>b: print('существует')
else: print('не существует')
```

Ответ: _____

8. Определите результат работы программы:

```
p=1
while p<10: p*=2
print(p)
```

Ответ: _____

9. Определите результат работы программы:

```
p=1;k=0
while p<7: p+=1; k+=p
print(k)
```

Ответ: _____

10. Определите результат работы программы:

```
p='hello'
for i in p: print(i,end='*')
```

Ответ: _____

11. Определите результат работы программы:

```
p=bin(32)
print(p[2:])
```

Ответ: _____

12. Определите результат работы программы:

```
p=bin(32);k=0
for i in p[2:]:k+=int(i)
print(k)
```

Ответ: _____

13. Определите результат работы программы:

```
n='start'; print(n[::-1])
```

Ответ: _____

14. Определите результат работы программы:

```
n='python3'
print(n[3:]+n[5:2:-1])
```

Ответ: _____

15. Определите результат работы программы:

```
for i in range(10):
    for j in range(i,10):
        print('*',end=' ')
    print()
```

Ответ: _____

