

Python

Введение

Урок 1



Что такое алгоритм?

Алгоритм — Это пошаговый план действий, которые должен выполнить исполнитель для решения задачи за конечное время.





Свойства алгоритма

Дискретность — алгоритм является конструктором из отдельных частей (команд), каждая из которых должна быть выполнена за конечное время.

Детерминированность (определённость) — при использовании алгоритма с одинаковыми исходными данными получается один и тот же результат.

Конечность (результативность) — для правильного набора данных алгоритм должен завершаться через определенное время.

Корректность — для допустимых исходных данных алгоритм должен приводить к правильному результату.

Массовость — алгоритм можно использовать для разных исходных данных.



Регистрация на сайте repl.it

[teacher? start here](#) [sign in](#)



repl.it is a **cloud coding environment** for Unlambdа

join a community of
engineers, teachers, and students



Search for a language, e.g. c++

Teachers start here





Регистрация на сайте repl.it

5



Log In

Log In



Log in

or sign up

[Forgot password?](#)

Log in





Регистрация на сайте repl.it

6



Sign Up

Log In



Sign Up

or log in

username

Ник

email

Почта

password

Пароль

Sign up





Регистрация на сайте repl.it

repl.it Sessions my repls community BETA Wonderful ▾

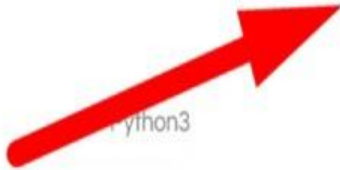
search for repls

Continue with Python3

New Repl

1-1 of 1 ◀ ▶

HelloWorld



Python3

5 days ago



Classrooms you might be interested in



 python™
Auto-Graded Course
with Solutions

JS
Auto-Graded Exercises
JavaScript

 React
React Native 101
React Native

browse more
classrooms



Регистрация на сайте repl.it



Select a Language

my repls

community
BETA

Wonderful ▾

search for a language, e.g. c++

▾ Popular

Python3: A dynamic language emphasizing readability.

Ruby: A natural dynamic object-oriented language.

JavaScript: The de facto language of the Web.

HTML, CSS, JS: The languages that make up the web.

Scheme: An elegant dynamic dialect of Lisp.





Регистрация на сайте repl.it

9

 @Wonderful/untitled 
No description 

my repls community BETA Wonderful ▾



 1 | not sure what to do? Run some [examples](#) (dismiss)

```
Python 3.6.1 (default, Dec 2015, 13:05:11)
[GCC 4.8.2] on linux
>
```





Простейшая программа

```
# В программе ничего нет
```

? Смысл этой программы?

комментарии после #
не учиваются

кодировка utf-8
по умолчанию)

```
# coding: utf-8  
# напишем чтонибудь из кода?)
```




```
print ( "3-1=?" )  
▶ print ( "Ответ: 2" )
```

переход на новую строку
осуществляется
автоматически

```
print ( '3-1=?' )  
print ( 'Ответ: 2' )
```




Вывод на экран

```
AnxiousElaborateCryptocurren...   
```

```
main.py  saved
```

```
1 print('3-1=?')  
2 print('Ответ: 2')
```

```
Python 3.6.1 (default, Dec 2015, 13:05:11)  
[GCC 4.8.2] on linux  
3-1=?  
Ответ: 2  
❯ 
```



Задание 1: Вывести на экран текст «елочкой»

```
Никита
      пошел
домой
```

Задание 2: Вывести на экран рисунок из букв

```
А
АА
АААА
АААААА
АААААААА
АААААААААА
```



Решение

main.py



saved

```
1 print('Никита')
2 print('    Пошел')
3 print('Домой')
4
```

```
Python 3.6.1 (default, Dec 2015, 13:05:11)
```

```
[GCC 4.8.2] on linux
```

```
Никита
```

```
    Пошел
```

```
Домой
```

```
> |
```

main.py



saved

```
1 print('A')
2 print('AAA')
3 print('AAAAA')
4 print('AAAAAAA')
5 print('AAAAAAAAA')
6
7
```

```
Python 3.6.1 (default, Dec 2015, 13:05:11)
```

```
[GCC 4.8.2] on linux
```

```
A
```

```
AAA
```

```
AAAAA
```

```
AAAAAAA
```

```
AAAAAAAAA
```

```
> |
```

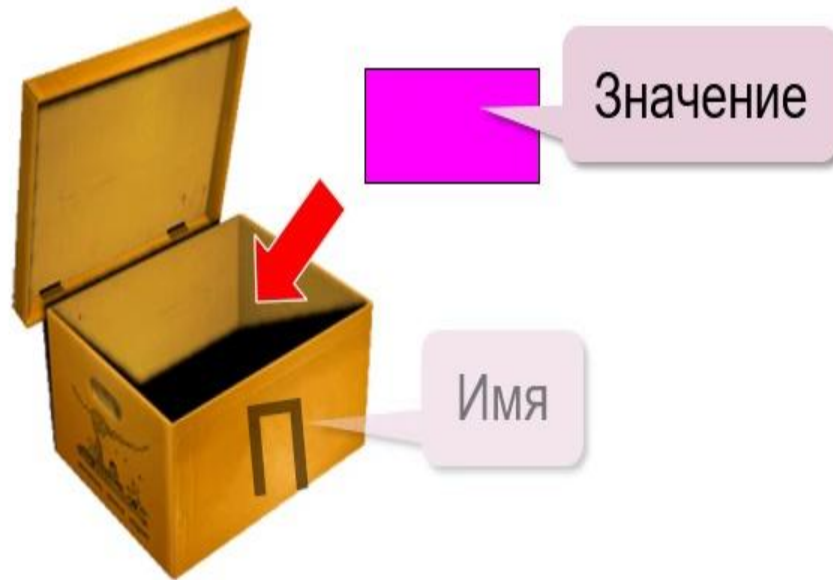


ввести два числа
вычислить их произведение
вывести ответ на экран

Псевдокод – алгоритм написанный на русском языке , при участии элементов языка программирования.



Переменные





МОЖНО использовать

- латинские буквы (A-Z, a-z)
- русские буквы (**лучше обойтись без них!**)
- цифры

НЕЛЬЗЯ использовать

- скобки () [] { }
- знаки +, =, !, ? и др.
- Не пишите транслейтом!



Какие имена правильные?

AXeand

Q&P

4Char

Володя

“Dart”

MI29

[QuNa]

_QUEN

А-А



Типы переменных

```
a = 5  
print ( type (a) )
```

int

целое число (*integer*)

```
a = 2.8  
print ( type (a) )
```

float

вещественное число

```
a = "Ваня"  
print ( type (a) )
```

str

символьная строка



Простейшая программа

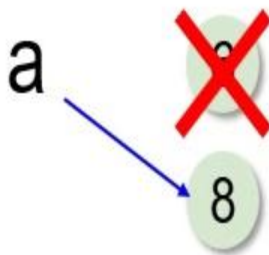
Если существующей переменной присвоить новое значение, то старое будет удалено из памяти!



оператор присваивания

a = 6

a = 8





Вывод данных

```
print ( c )
```

значение
переменной

```
print ( "Ответ: ", c )
```

значение и
текст

перечисление через запятую

```
print ( "Ответ: ", a+b )
```

вычисление
выражения

```
print ( a, "+", b, "=", c )
```

1 + 4 = 5

через пробелы

```
print ( a, "+", b, "=", c, sep = "" )
```

1+4=5

убрать разделители



Разность: простое решение

```
a = int ( input ( ) )  
b = int ( input ( ) )  
c = a - b  
print ( c )
```



Есть замечания?



Сложение чисел: полное решение

```
print ( "Введите два числа: " )  
a = int ( input () )  
b = int ( input () )  
c = a - b  
print ( a, "-", b, "=", c, sep=" " )
```

сообщение
пользователю

компилятор

Протокол:

Введите два целых числа

20

15

20-15=5



Задание : Ввести три числа, найти их сумму, произведение и среднее арифметическое.

Пример:

Введите три числа:

4

5

7

$$4+5+7=16$$

$$4*5*7=140$$

$$(4+5+7) / 3=5.333333$$



Решение

```
main.py  saved
1 print('Введите три числа')
2 a = int(input())
3 b = int(input())
4 c = int (input())
5 d = a + b + c
6 print( a, "+", b, "+", c, "=", d, sep = "" )
7 f = a * b * c
8 print( a, "*", b, "*", c, "=", f, sep = "" )
9 g = d / 3
10 print(d,"/",3,"=",g,sep="")
11
```

```
Python 3.6.1 (default, Dec 2015, 13:05:11)
[GCC 4.8.2] on linux
Введите три числа
4
5
7
4+5+7=16
4*5*7=140
16/3=5.333333333333333
> □
```



Арифметическое выражения

3 1 2 4 5 6
`a = (c + b**4*6 - 3) / 8 * d`

Приоритет (*старшинство*):

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

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

перенос на
следующую строку



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 = 55
b = d // 10    # 8
a = d % 10    # 5
d = a % b     # 5
d = b % a     # 3
```

```
a = 10
b = 28
d = a // b    # 0
a = a % b    # 15
```



Операторы // и %

```
a = 4321
d = a % 10; print( d )
a = a // 10 # 123
d = a % 10; print( d )
a = a // 10 # 12
d = a % 10; print( d )
a = a // 10 # 1
d = a % 10; print( d )
a = a // 10 # 0
```

1

2

3

4



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
единицу



Ввод двух значений в одной строке

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

21 33 input()

21 33 input().split()

целые

применить

разделяет строку на части

21 33 map (int, input().split())

эту операцию

для всех частей

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




Задание 1: Ввести три числа: цену пирожка (два числа: рубли, потом – копейки) и количество пирожков. Найти сумму, которую нужно заплатить (рубли и копейки)

Пример:

Стоимость пирожка:

12 50

Сколько пирожков:

5

К оплате: 62 руб. 50 коп.

Задание 2: Ввести число, обозначающее количество секунд. Вывести то же самое время в часах, минутах и секундах.

Пример:

Число секунд:

8325

2 ч. 18 мин. 45 с

Решение 1 задания

```
1 r, k = map (int, input().split())
2 n = int (input())
3 sumk = k * n
4 first = r * n + sumk // 100
5 second = sumk % 100
6 print(first, second)
```

Решение 2 задания

```
1  n = int (input())
2  s = n % 60
3  m = n // 60 % 60
4  h = n // 3600
5  print(h, m, s)
```



Задание 1: Занятия в школе начинаются в 8-30. Урок длится 45 минут, перерывы между уроками – 10 минут. Ввести номер урока и вывести время его окончания.

Пример:

Введите номер урока:

5

12-10



Молодцы!

