Программирование на языке Python

Циклы по переменной

© К.Ю. Поляков, Е.А. Ерёмин, 2018

Цикл с переменной

Задача. Вывести 10 раз слово «Привет!».

? Можно ли сделать с циклом «пока»?

```
i = 0
while i < 10:
    print("Привет!")
    i += 1
```

Цикл с переменной:

```
for i in range(10):
    print("Привет!")
```

в диапазоне [0,**10**)

! Не включая **10**!

`range(10)` → `0, 1, 2, 3, 4, 5, 6, 7, 8, 9`

Цикл с переменной

Задача. Вывести все степени двойки от 2^1 до 2^{10}.

? Как сделать с циклом «пока»?

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

Цикл с переменной:

```
for k in range(1,11) :
    print ( 2**k )
```

в диапазоне [1,**11**)

! Не включая **11**!

```
range(1,11) → 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
```

Цикл с переменной: другой шаг

10,9,8,7,6,5,4,3,2,1

шаг

```
for k in range(10,0,-1):
  print ( k**2 )
```

```
100
81
64
49
36
25
16
9
4
1
```

Что получится?

1,3,5,7,9

```
for k in range(1,11,2):
  print ( k**2 )
```

```
1
9
25
49
81
```

Сколько раз выполняется цикл?

```
a = 1
for i in range( 3): a += 1
```

a = 4

```
a = 1
for i in range( 3,1): a += 1
```

a = 1

```
a = 1
for i in range( 1,3,-1): a += 1
```

a = 1

```
a = 1
for i in range( 3,1,-1): a += 1
```

a = 3

Задачи

«A»: Найдите все пятизначные числа, которые при делении на 133 дают в остатке 125, а при делении на 134 дают в остатке 111.

«B»: Натуральное число называется **числом Армстронга**, если сумма цифр числа, возведенных в N-ную степень (где N – количество цифр в числе) равна самому числу. Например, $153 = 1^3 + 5^3 + 3^3$. Найдите все трёхзначные Армстронга.

Задачи

«C»: Натуральное число называется автоморфным, если оно равно последним цифрам своего квадрата. Например, $25^2 = 625$. Напишите программу, которая получает натуральное число N и выводит на экран все автоморфные числа, не превосходящие N.

Пример:

```
Введите N:
1000
1*1=1
5*5=25
6*6=36
25*25=625
76*76=5776
```

Вложенные циклы

Задача. Вывести все простые числа в диапазоне от 2 до 1000.

```
сделать для n от 2 до 1000
  если число n простое то
    вывод n
```

нет делителей [2.. n-1]: проверка в цикле!

? Что значит «простое число»?

```
for n in range(2, 1001):
  if число n простое:
    print( n )
```

Вложенные циклы

```
for n in range(2, 1001):
    count = 0
    for k in range(2,n):
        if n % k == 0:
            count += 1
    if count == 0:
        print( n )
```

вложенный цикл

Вложенные циклы

```
for i in range(1,4):
  for k in range(1,4):
    print( i, k )
```

```
1 1
1 2
1 3
2 1
2 2
2 3
3 1
3 2
3 3
```

? Как меняются переменные?

! Переменная внутреннего цикла изменяется быстрее!

Вложенные циклы

```
for i in range(1,5):
  for k in range(1,i+1):
    print( i, k )
```

1 1
2 1
2 2
3 1
3 2
3 3
4 1
4 2
4 3
4 4

? Как меняются переменные?

! Переменная внутреннего цикла изменяется быстрее!

Поиск простых чисел – как улучшить?

$$n = k \cdot m, \quad k \le m \quad \Rightarrow k^2 \le n \quad \Rightarrow k \le \sqrt{n}$$

```
while k <= math.sqrt(n):
    ...
```

? Что плохо?

```
count = 0
k = 2
while k*k <= n :
    if n % k == 0:
        count += 1
    k += 1
```

? Как ещё улучшить?

выйти из цикла

```
while k*k <= n:
    if n % k == 0: break
    k += 1
if k*k > n:
    print ( n )
```

если вышли по условию

Задачи

«A»: Напишите программу, которая получает натуральные числа A и B (A<B) и выводит все простые числа в интервале от A до B.

Пример:

```
Введите границы диапазона:
10 20
11 13 17 19
```

«B»: В магазине продается мастика в ящиках по 15 кг, 17 кг, 21 кг. Как купить ровно 185 кг мастики, не вскрывая ящики? Сколькими способами можно это сделать?

Задачи

«С»: Ввести натуральное число N и вывести все натуральные числа, не превосходящие N и делящиеся на каждую из своих цифр.

Пример:

```
Введите N:
15
1 2 3 4 5 6 7 8 9 11 12 15
```