

# Семинар 6

Итератор

range

Цикл for

Кортежи

# Функция `range`, объект итератора и цикл *for*

```
for element in <iterator object>:  
    print element
```

В качестве `<iterator object>` может быть всё, что можно перебирать: список, строка, словарь, открытый файл...

Функция **`range`** позволяет генерировать объекты типа `iterable` (к элементам которых можно получать последовательный доступ) состоящие из целых чисел.

```
>>> range(7)  
[0, 1, 2, 3, 4, 5, 6]  
>>> range(1, 8)  
[1, 2, 3, 4, 5, 6, 7]  
>>> range(0, 20, 5)  
[0, 5, 10, 15]  
>>> range(0, -7, -1)  
[0, -1, -2, -3, -4, -5, -6]  
>>> range(1, 0)  
[]
```

# Задача 1

Проверьте, для целых чисел в пределах [0, 100] является ли пара (x, y) корнями уравнения:

$$x*x - x*y - 4 = 0$$

```
for i in range(0, 101):  
    for j in range(0, 101):  
        if i * i - i * j - 4 == 0:  
            print("Найден корень (" , i, ", ", j, ")", sep='')
```

**Вывод:**

*Найден корень (2, 0)*

*Найден корень (4, 3)*

## Задача 2

Для заданного пользователем натурального  $n$  вывести значение числа  $\Pi$ , вычисленное по формуле Лейбница:

$$4 * (1/1 - 1/3 + 1/5 - 1/7 + \dots) = \Pi$$

```
# todo: проверка входных данных
n = int(input("Введите количество членов ряда: "))
sign = 1
pi = 0
for i in range(1, n + 1, 2):
    pi += sign * 1 / i
    sign = -sign
pi *= 4
print("Pi =", pi)
```

### Вывод:

Введите точность вычислений 1000

Pi = 3.139592655589785

# Задача 3

*# Найдите все 4-х значные числа, являющиеся полными квадратами (имеют целое значение кв. корня). Сколько их?*

```
import math, time
```

*# Способ 1*

```
startTime = time.clock()
```

```
count = 0
```

```
for i in range(1000, 9999 + 1):
```

```
    if (math.sqrt(i).is_integer()):
```

```
        print(i, end=" ")
```

```
        count += 1
```

```
print("\nЗатрачено времени: %g, всего чисел %d" % (time.clock() - startTime, count))
```

*# Способ 2*

```
startTime = time.clock()
```

```
count = 0
```

```
startNum = math.ceil(math.sqrt(1000))
```

```
finishNum = math.floor(math.sqrt(9999))
```

```
for i in range(startNum, finishNum + 1):
```

```
    print(i * i, end=" ")
```

```
    count += 1
```

```
print("\nЗатрачено времени: %g, всего чисел %d" % (time.clock() - startTime, count))
```

# Кортежи (tuple) в языке Python

**Кортеж (tuple)** – это неизменяемая последовательность данных. Кортежи схожи со строками, с той разницей – что элементы кортежа могут быть **любыми**, а не только символами, для задания кортежа используются обычные скобки.

# Кортежи (tuple) в языке Python

```
myTuple = 10, 5 # создание кортежа
myTuple = (10, 5) # лучше так
print(myTuple)
print(myTuple[0]) # обращение к элементу
print(myTuple[1:]) # срез, аналогичный строке
# myTuple[0] = 11 # нельзя!
myTuple = tuple("Hello") # функция tuple
myTuple1 = ("a")
print(type(myTuple1)) # кортеж???
myTuple1 = ("a",) # кортеж!
print(myTuple1 + myTuple)
print(myTuple * 3)
print(myTuple + (len(myTuple),)) # что будет напечатано?
# Только на Питоне
a = 10
b = 5
a, b = b, a # поменять переменные местами (А как в других языках?)
# проверка существования элемента в кортеже
if 'e' in myTuple: # есть и not in
    print("'e' exists!")
# r перед строкой задает "необработанную строку"
```

# Задача 4

*# Товары и стоимость представлены кортежами данных, например, ('хлеб', 23) и т.д.*

*# Окончание ввода "0" в товаре или стоимости. Необходимо рассчитать общую стоимость товара "молоко".*

```
goods = tuple() # объявление пустого кортежа
sum = 0
while not ('0' in goods):
    goods = (input("Введите название товара: "), input("Введите сумму:"))
    if (not goods[1].isdigit()):
        print ("Надо ввести сумму цифрами, повторите.")
        continue
    print(goods)
    name, val = goods
    if (name == "молоко"):
        sum += int(val)
print("Всего товара молоко:", sum)
```



## Задача 5

```
# Дана строка содержащая ФИО, возраст, телефон через запятую.  
Написать функцию, которая возвращает пару 'фамилия, возраст'  
# "Иванов Иван Иванович, 23 года, 8-999-12345678"  
# Иванов, 23
```

```
def getSurnameAge(info):  
    parts = info.split(",") # Возвращает список  
    return parts[0].split()[0], parts[1].split()[0]
```

```
personInfo = "Иванов Иван Иванович, 23 года, 8-999-12345678"  
print(getSurnameAge(personInfo))
```