

Программирование на языке Python

Массивы (списки)

Списки

Большинство программ работает не с отдельными переменными, а с набором переменных. Например, программа может обрабатывать информацию об учащемся класса, считывая список учащихся с клавиатуры или из файла.

Для хранения таких данных можно использовать структуру данных, называемую в Питоне **список** (в большинстве же языков программирования используется другой термин “массив”).

Список представляет собой последовательность элементов, пронумерованных от 0, как символы в строке. Список можно задать перечислением элементов списка в квадратных скобках, например, список можно задать так:

P = [2, 3, 5, 7, 11, 13]

R = ['Red', 'Orange', 'Yellow', 'Green', 'Blue', 'Indigo', 'Violet']

Здесь P[0]==2, R[2]==‘Yellow’.

Что такое массив?



Как ввести 10000 переменных?

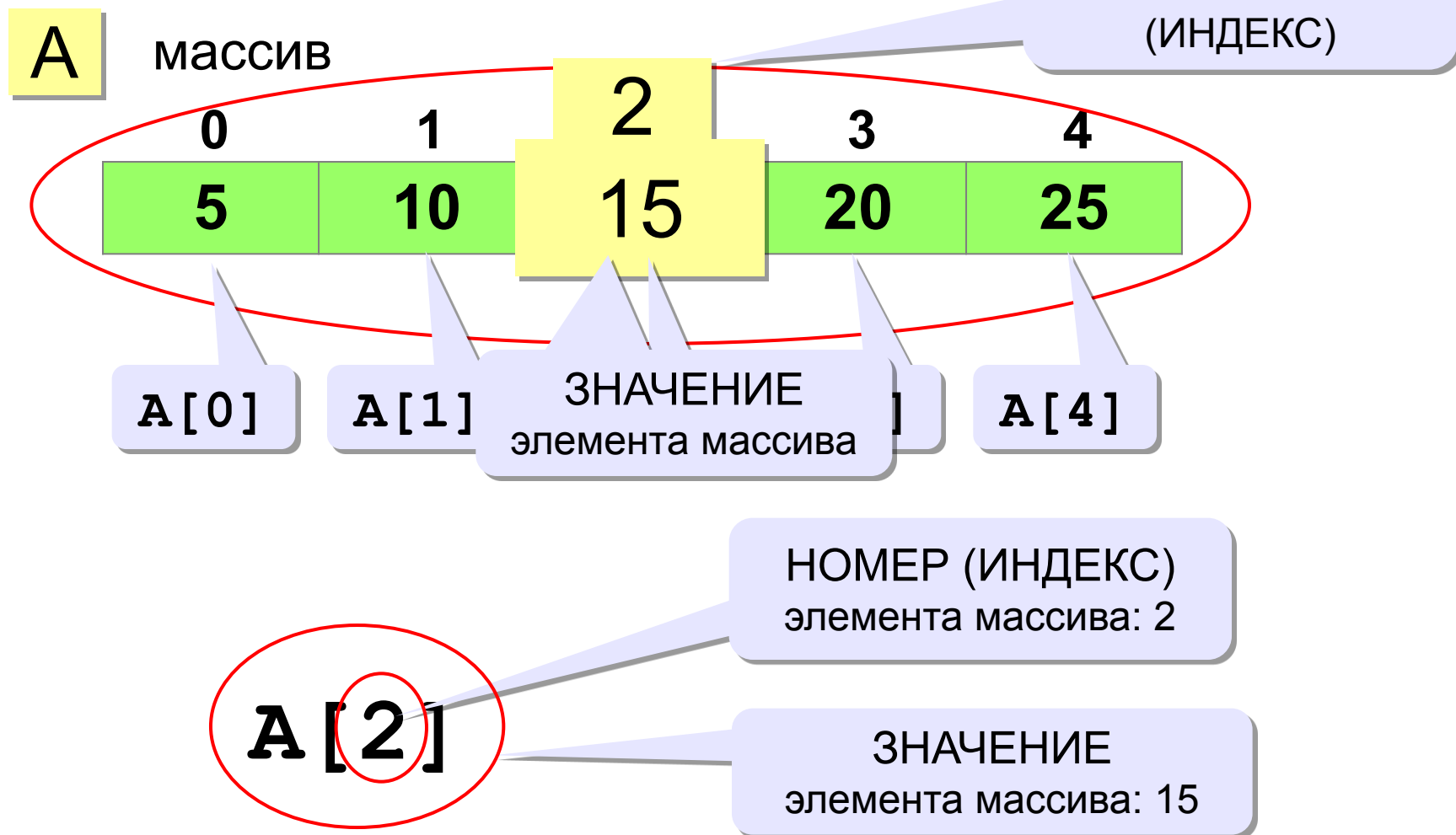
Массив – это группа переменных одного типа, расположенных в памяти рядом (в соседних ячейках) и имеющих общее имя. Каждая ячейка в массиве имеет уникальный номер (индекс).

Надо:

- выделять память
- записывать данные в нужную ячейку
- читать данные из ячейки

Что такое массив?

! Массив = таблица!



Массивы в Python: списки

```
A = [1, 3, 4, 23, 5]
```

```
A = [1, 3] + [4, 23] + [5]
```

```
[1, 3, 4, 23, 5]
```

```
A = [0] * 10
```

```
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
```



Что будет?

Создание массива из N элементов:

```
N = 10
```

```
A = [0] * N
```

Заполнение массива

Целыми числами (начиная с 0!):

```
N = 10      # размер массива
A = [0]*N   # выделить память
for i in range(N):
    A[i] = i
```

В краткой форме:

```
N = 10      # размер массива
A = [ i for i in range(N) ]
```



Как заполнить, начиная с 1?



Как заполнить квадратами чисел?

Заполнение случайными числами

из библиотеки (модуля)
random

взять функцию randint

```
from random import randint
N = 10      # размер массива
A = [0]*N   # выделить память
for i in range(N):
    A[i] = randint(20, 100)
```

В краткой форме:

```
from random import randint
N = 10
A = [ randint(20, 100)
      for i in range(N) ]
```

Вывод массива на экран

Как список:

```
print ( A ) [1, 2, 3, 4, 5]
```

В строчку через пробел:

```
for i in range(N):  
    print ( A[i], end = " " )
```

1 2 3 4 5

или так:

```
for x in A:  
    print ( x, end = " " )
```

пробел после
вывода
очередного числа

1 2 3 4 5

или так:

```
print ( *A ) ↔ print ( 1, 2, 3, 4, 5 )
```

разбить список
на элементы

Задачи

«3»: Ввести два натуральных числа **a** и **b** ($a < b$) и заполнить массив из 10 элементов случайными числами в диапазоне от **a** до **b**.

Пример:

Введите границы диапазона:

5 10

10 9 10 6 8 5 9 6 10 9

«4»: Ввести два натуральных числа **a** и **b** и заполнить массив из 10 элементов случайными числами в диапазоне между **a** и **b** (**a** может быть больше **b**).

Пример:

Введите границы диапазона:

10 5

10 9 10 6 8 5 9 6 10 9

Задачи

«5»: Ввести два натуральных числа **a** и **b** и заполнить массив из 10 элементов: первая половина массива заполняется случайными числами в диапазоне между **a** и **b** (**a** может быть больше **b**), а вторая половина массива содержит их квадраты в том же порядке.

Пример:

Введите границы диапазона:

10 5

5 8 7 10 6 25 64 49 100 36

Ввод массива с клавиатуры

Создание массива:

```
N = 10  
A = [0]*N
```

Ввод по одному элементу в строке:

```
for i in range(N):  
    A[i] = int( input() )
```

или кратко:

```
A = [int(input())  
      for i in range(N)]
```

Ввод массива с клавиатуры

Ввод всех чисел в одной строке:

```
data = input()      # "1 2 3 4 5"
s = data.split()    # ["1", "2", "3", "4", "5"]
A = [ int(x) for x in s ]
                    # [1, 2, 3, 4, 5]
```

или так:

```
A = [int(x) for x in input().split()]
```

Как обработать все элементы массива?

Создание массива:

```
N = 5  
A = [0] * N
```

Обработка:

```
# обработать A[0]  
# обработать A[1]  
# обработать A[2]  
# обработать A[3]  
# обработать A[4]
```



1) если N велико (1000, 1000000)?

2) при изменении N программа не должна меняться!

Как обработать все элементы массива?

Обработка с переменной:

```
i = 0
# обработать A[i]
i += 1
# обработать A[i]
i += 1
# обработать A[i]
i += 1
# обработать A[i]
i += 1
# обработать A[i]
```

```
i += 1
```



Обработка в цикле:

```
i = 0
while i < N:
    # обработать A[i]
    i += 1
```

Цикл с переменной:

```
for i in range(N):
    # обработать A[i]
```



Перебор элементов

Общая схема (можно изменять $A[i]$):

```
for i in range(N):  
    ... # сделать что-то с A[i]
```

```
for i in range(N):  
    A[i] += 1
```

Если не нужно изменять $A[i]$:

```
for x in A:  
    ... # сделать что-то с x
```

$x = A[0], A[1], \dots, A[N-1]$

```
for x in A:  
    print ( x )
```

Что выведет программа?

```
A = [2, 3, 1, 4, 6, 5]
```

```
print( A[3] ) # 4
```

```
print( A[0]+2*A[5] ) # 12
```

```
A[1] = A[0] + A[5] # 7
```

```
print( 3*A[1]+A[4] ) # 27
```

```
A[2] = A[1]*A[4] # 18
```

```
print( 2*A[1]+A[2] ) # 22
```

```
for k in range(6):  
    A[k] += 2 # [4, 5, 3, 6, 8, 7]  
print( 2*A[3]+3*A[4] ) # 36
```


Подсчёт нужных элементов

Задача. В массиве записаны данные о росте баскетболистов. Сколько из них имеет рост больше 180 см, но меньше 190 см?



Как решать?

```
count = 0
for x in A:
    if 180 < x and x < 190:
        count += 1
```

Перебор элементов

Задача. Найти сумму чётных элементов массива.

```
summa = 0
for x in A:
    if x % 2 == 0:
        summa += x
print ( summa )
```



Как определить, что элемент чётный?

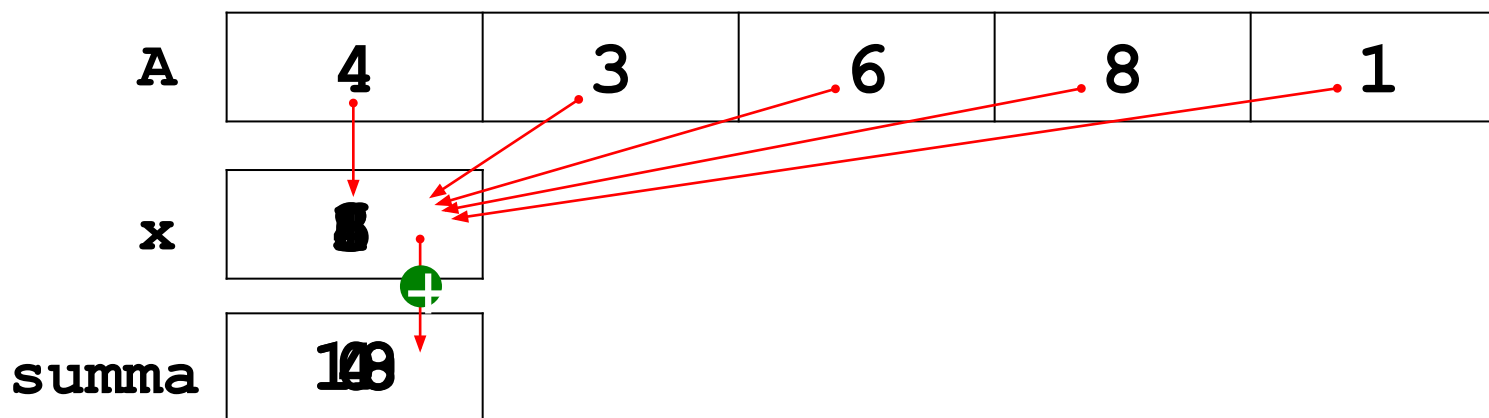
или так:

```
B = [x for x in A
      if x % 2 == 0]
print ( sum(B) )
```

сумма массива B

Как работает цикл?

```
summa = 0
for x in A:
    if x % 2 == 0:
        summa += x
```



Среднее арифметическое

Задача. Найти среднее арифметическое элементов массива, которые оканчиваются на цифру 5.

```
count = 0
summa = 0
for x in A:
    if x % 10 == 5:
        count += 1
        summa += x
print ( summa/count )
```



Как определить, что оканчивается на 5?

среднее
арифметическое

или так:

```
B = [ x for x in A
      if x % 10 == 5 ]
print ( sum(B)/len(B) )
```

отбираем нужные

Пример № 6

Сформировать и вывести целочисленный массив размера N , содержащий N первых положительных нечетных чисел: 1, 3, 5,

Задачи

«3»: Введите массив из 5 элементов с клавиатуры и найдите среднее арифметическое его значений.

Пример:

Массив :

1 2 3 4 5

Среднее арифметическое 3.000

«4»: Заполните массив из 10 элементов случайными числами в интервале $[0,100]$ и подсчитайте отдельно среднее значение всех элементов, которые <50 , и среднее значение всех элементов, которые ≥ 50 .

Пример:

Массив :

3 2 52 4 60 50 1 2 60 58 6

Ср. арифм. элементов < 50 : 3.000

Ср. арифм. элементов ≥ 50 : 56.000

Самостоятельно:

1. Дано целое число $N (> 0)$. Сформировать и вывести целочисленный массив размера N , содержащий степени двойки от первой до N -й: 2, 4, 8, 16,

Ввод: 7

Вывод: 2 4 8 16 32 64 128

2. Дано целое число $N (> 1)$, а также первый член A и разность D *арифметической прогрессии*. Сформировать и вывести массив размера N , содержащий N первых членов данной прогрессии:

$A, A + D, A + 2 \cdot D, A + 3 \cdot D,$

Ввод: 4 1.14 3.04

Вывод: 1.14 4.18 7.23 10.27

Самостоятельно:

3. Выведите значение наименьшего из всех положительных элементов в списке. Известно, что в списке есть хотя бы один положительный элемент, а значения всех элементов списка по модулю не превосходят 1000.

Ввод: 5 -4 3 -2 1

Вывод: 1

4. Дан список чисел. Выведите все элементы списка, которые больше предыдущего элемента.

Ввод: 1 5 2 4 3

Вывод : 5 4

5*. Дан список. Выведите те его элементы, которые встречаются в списке только один раз. Элементы нужно выводить в том порядке, в котором они встречаются в списке.

Ввод: 2 5 2 4 4 3 2

Вывод : 5 3

Задачи

«5»: Введите размер массива N и заполните массив из N элементов **числами Фибоначчи**. Первые два числа Фибоначчи равны 1, а каждое следующее равно сумме двух предыдущих.

Пример:

Введите размер массива:

6

Числа Фибоначчи:

1 1 2 3 5 8