

Программирование (Python)

§ 21. Массивы

Что такое массив?



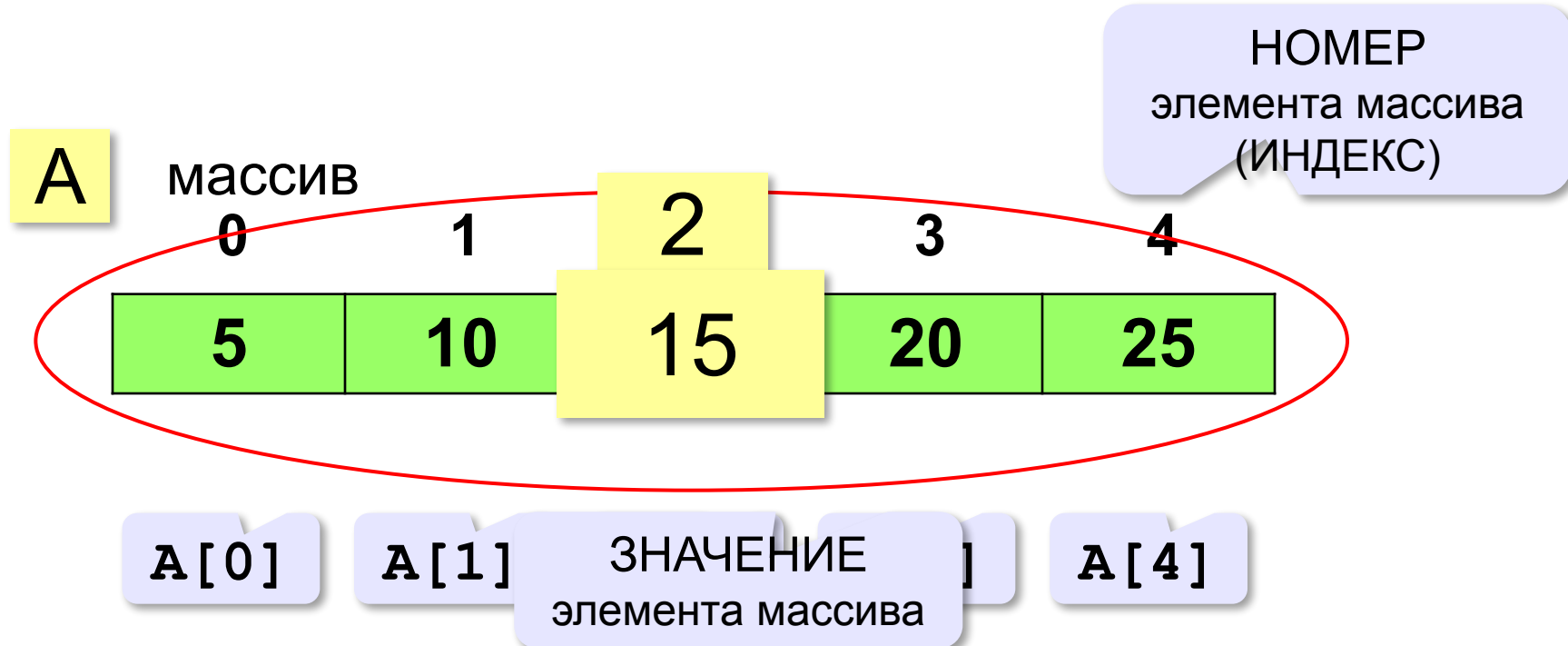
Как ввести 10000 переменных?

Массив – это группа переменных одного типа, расположенных в памяти рядом (в соседних ячейках) и имеющих общее имя.

Надо:

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

Обращение к элементу массива



Индекс элемента — это значение, которое указывает на конкретный элемент массива.



Нумерация с нуля!

Создание массива

11	22	35	41	53
----	----	----	----	----

```
A = [11, 22, 35, 41, 53]
```

```
A = [11, 22] + [35, 41] + [53]
```

```
A = [11] * 5
```

```
A = [11] + [11] + [11] + [11] + [11]
```

11	11	11	11	11
----	----	----	----	----

Обращение к элементу массива

A[2]

ИНДЕКС элемента массива: 2

ЗНАЧЕНИЕ элемента массива

0	1	2	3	4
23	12	7	43	51

```
i = 1  
A[2] = A[i] + 2*A[i-1] + A[2*i+1]  
print( A[2]+A[4] )
```



Что получится?

```
A[2] = A[1] + 2*A[0] + A[3]  
print( A[2]+A[4] )
```

101

152

Что неверно?

0	1	2	3	4
1	2	3	4	5

```
A = [1, 2, 3, 4, 5]
x = 1
print( A[x-8] )
A[x+4] = A[x-1] + A[2*x]
```



Что плохо?



```
print( A[-7] )
A[-7] = A[0] + A[2]
```

Выход за границы массива — это обращение к элементу с индексом, который не существует в массиве.

Перебор элементов массива

Перебор элементов: просматриваем все элементы массива и, если нужно, выполняем с каждым из них некоторую операцию.

```
N = 10
```

```
A = [0]*N    # память уже выделена
```

```
0, 1, 2, 3, ..., N-1
```

```
for i in range(N) :
```

```
    # здесь работаем с A[i]
```

Заполнение массива

$[0, 1, 2, 3, \dots, N-1]$

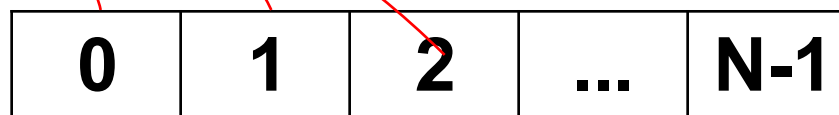
```
for i in range(N):  
    A[i] = i
```



Что произойдёт?

В развёрнутом виде

```
A[0] = 0  
A[1] = 1  
A[2] = 2  
...  
A[N-1] = N-1
```



В стиле Python:

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


Заполнение массива в обратном порядке

N	...	3	2	1
---	-----	---	---	---

```
A[0] = N  
A[1] = N-1  
A[2] = N-2  
...  
A[N-1] = 1
```

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



Как меняется X?

X = N, N-1, ..., 2, 1

начальное
значение

уменьшение
на 1

Заполнение массива в обратном порядке

N	...	3	2	1
---	-----	---	---	---

$$A[i] = X$$

? Как связаны i и X ?

i	X
0	N
1	N-1
2	N-2
...	...
N-1	1

+1 -1

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

В стиле Python:

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

! Сумма i и X не меняется!

$$i + X = N$$

$$X = N - i$$

Вывод массива на экран

Весь массив сразу:

```
print( A )
```

```
[1, 2, 3, 4, 5]
```

По одному элементу:

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

в столбик

или так:

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

для всех элементов в массиве A



Как вывести
в строчку?

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

пробел между
элементами

Вывод массива на экран (Python)

[1, 2, 3, 4, 5]

```
print ( *A )
```



```
print ( 1 , 2 , 3 , 4 , 5 )
```

разбить список
на элементы

1 2 3 4 5

Ввод с клавиатуры

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

или так:

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

С подсказкой для ввода:

```
for i in range(N) :  
  
    print("A[{}]=".format(i) ,  
          end="")  
    A[i] = int(input())
```

```
A[0] = 5  
A[1] = 12  
A[2] = 34  
A[3] = 56  
A[4] = 13
```

Ввод с клавиатуры (Python)

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()]
```

В других языках программирования

Паскаль:

объявление массива

```
const N = 10;  
var A: array[0..N-1] of integer;  
...  
for i:=0 to N-1 do  
    A[i] = i;  
for i:=0 to N-1 do  
    write(A[i], ' ');
```

В других языках программирования

C++:

```
int A[N], i;  
for (i = 0; i < N; i++)  
    A[i] = i;  
for (i = 0; i < N; i++)  
    cout << A[i] << " ";
```



Нумерация элементов
всегда с нуля!

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) ]
```

Обработка элементов массива

```
N = 10
```

```
A = [0]*N    # память уже выделена
```

```
for i in range(N):  
    # здесь работаем с A[i]
```

Вывести на экран в столбик:

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



Что вместо «???»?

Вывести на экран в строчку:

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

```
print( *A )
```

Обработка элементов массива

Вывести числа, на 1 большие, чем $A[i]$:

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



Что вместо «???»?

Вывести последние цифры:

```
for i in range(N):  
    print( A[i]%10 )
```

Обработка элементов массива

Заполнить нулями:

```
for i in range(N) :  
    A[i] = 0
```



Что вместо «???»?

Увеличить на 1:

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

Умножить на 2:

```
for i in range(N) :  
    A[i] *= 2
```

Программирование (Python)

§ 21. Алгоритмы обработки массивов

Сумма элементов массива

Задача. Найти сумму элементов массива из N элементов.

? Какие переменные нужны?

```
Sum = 0
for i in range(N):
    Sum += A[i]
print( Sum )
```

5	2	8	3	1
---	---	---	---	---

i	Sum
	0
0	5
1	7
2	15
3	18
4	19

В стиле Python:

```
print( sum(A) )
```

Сумма элементов массива (Python)

Задача. Найти сумму элементов массива A.

```
Sum = 0
```

```
for x in A:
```

```
    Sum += x
```

```
print( Sum )
```

для всех
элементов из A



Не нужно знать размер!

или так:

```
print( sum(A) )
```

встроенная
функция

Сумма не всех элементов массива

Задача. Найти сумму чётных элементов массива.

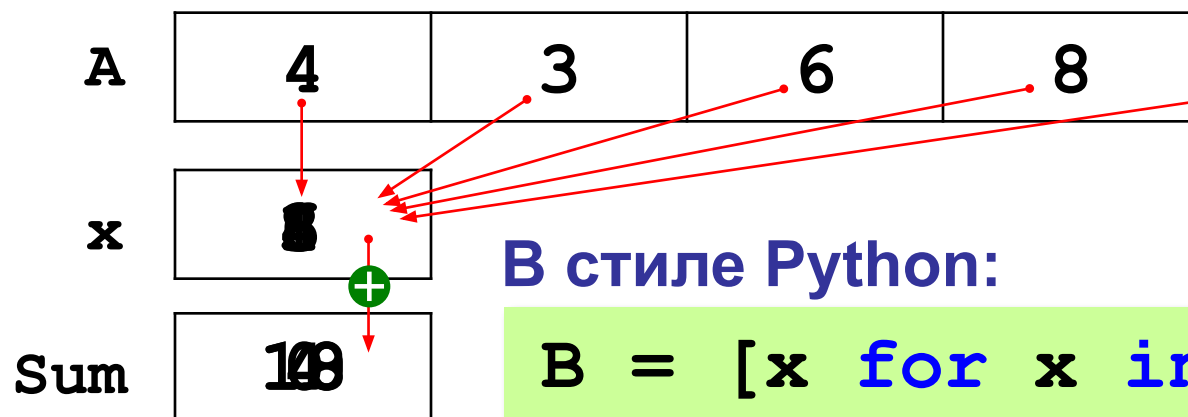
 Что делаем с нечётными?

```
Sum = 0
for i in range(N):
    if A[i] % 2 == 0:
        Sum += A[i]
print( Sum )
```


Сумма не всех элементов массива

Задача. Найти сумму чётных элементов массива.

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



отбираем в новый массив все нужные значения

В стиле Python:

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

Подсчёт элементов по условию

Задача. Найти количество чётных элементов массива.

? Какие переменные нужны?

переменная-
счётчик

```
count = 0
for i in range(N):
    if A[i] % 2 == 0:
        count += 1
print(count)
```

? Что тут делаем?

Подсчёт элементов по условию (Python)

Задача. Найти количество чётных элементов массива.

```
count = 0
for x in A:
    if x % 2 == 0:
        count += 1
print( count )
```

В стиле Python:

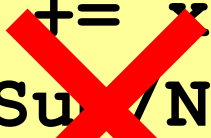
```
B = [x for x in A
      if x % 2 == 0]
print ( len(B) )
```

размер массива

Среднее арифметическое

Задача. Найти среднее арифметическое элементов массива, которые больше 180 (рост в см).

```
Sum = 0
for x in A:
    if x > 180:
        Sum += x
print( Sum / N )
```




Что плохо?

Среднее арифметическое

Задача. Найти среднее арифметическое элементов массива, которые больше 180 (рост в см).

 Какие переменные нужны?

```
Sum = 0
count = 0
for x in A:
    if x > 180:
        count += 1
        Sum += x
print( Sum/count )
```

 Что тут делаем?

Среднее арифметическое (Python)

Задача. Найти среднее арифметическое элементов массива, которые больше 180 (рост в см).

```
B = [ x for x in A  
      if x > 180]  
print ( sum(B)/len(B) )
```

отбираем нужные

Обработка потока данных

Задача. С клавиатуры вводятся числа, ввод завершается числом 0. Определить, сколько было введено положительных чисел.

- 1) нужен счётчик
- 2) счётчик увеличивается
- 3) нужен цикл
- 4) это цикл с условием (число шагов неизвестно)

?

Когда увеличивать счётчик?

?

Какой цикл?

счётчик = 0

пока не введён 0:

если введено число > 0 то

счётчик := счётчик + 1

Обработка потока данных

```
count = 0
x = int(input())
while x != 0:
    if x > 0:
        count += 1
    x = int(input())
print(count)
```

откуда взять **x**?



Что плохо?

Найди ошибку!

```
count = 0
x = int(input())
while x != 0:
    if x > 0:
        count += 1
pr x = int(input())
```

Найди ошибку!

```
count = 0      ut() )
while x == 0:
    if x > !=:
        count += 1
    x = int(input() )
print( count )
```

Обработка потока данных

Задача. С клавиатуры вводятся числа, ввод завершается числом 0. Найти сумму введённых чисел, оканчивающихся на цифру "5".

- 1) нужна переменная для суммы
- 2) число добавляется к сумме, если оно заканчивается на "5"
- 3) нужен цикл с условием

сумма = 0

пока не введён 0:

если число оканчивается на "5" то

сумма := сумма + число



Как это записать?

```
if x % 10 == 5:
```

Обработка потока данных

Задача. С клавиатуры вводятся числа, ввод завершается числом 0. Найти сумму введённых чисел, оканчивающихся на цифру "5".

```
sum = 0
x = int(input())
while x != 0:
    if x % 10 == 5:
        sum += x
    x = int(input())
print( sum )
```



Чего не хватает?

Найди ошибку!

```
sum = 0
x = int(input())
    if x % 10 == 5:
        sum += x
    x = int(input())
print( sum )
```

Перестановка элементов массива



Как поменять местами значения двух переменных *a* и *b*?

вспомогательная
переменная

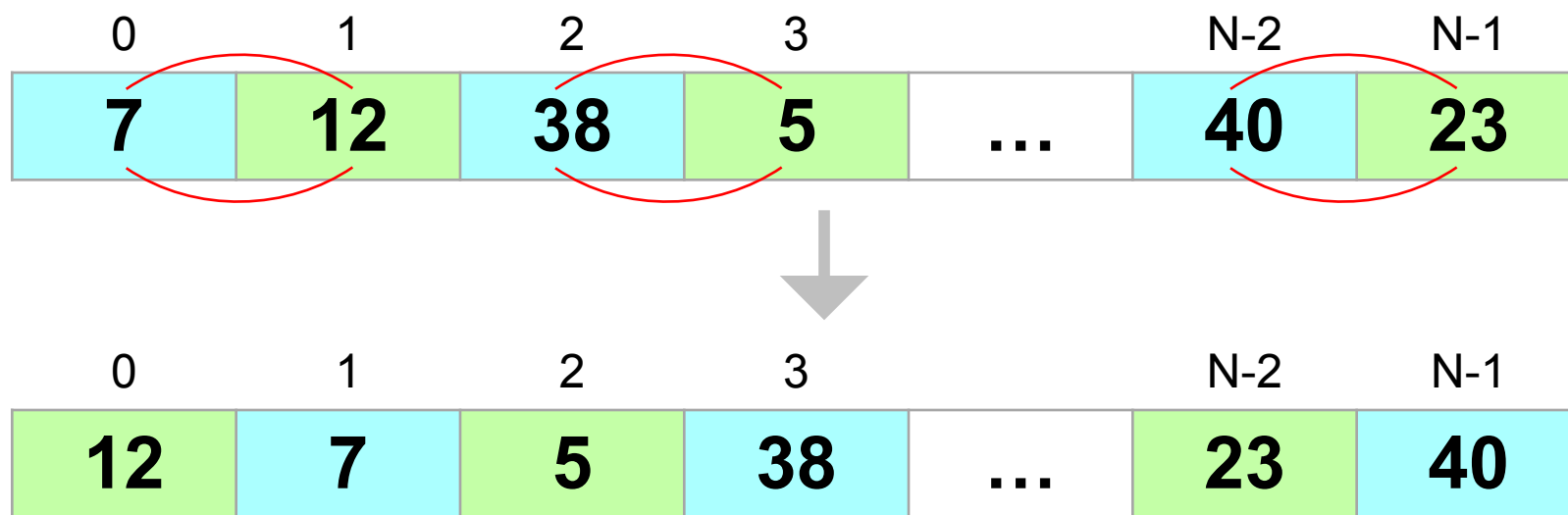
```
c = a  
a = b  
b = c
```

элементы массива:

```
c = A[i]  
A[i] = A[k]  
A[k] = c
```

Перестановка пар соседних элементов

Задача. Массив A содержит чётное количество элементов N . Нужно поменять местами пары соседних элементов: 0-й с 1-м, 2-й — с 3-м и т. д.



Перестановка пар соседних элементов

```
for i in range(N) :
```

```
    поменять местами A[i] и A[i+1]
```



Что плохо?

0	1	2	3	4	5
7	12	38	5	40	23
12	7	38	5	40	23
12	38	7	5	40	
12	38	5	7	40	23
12	38	5	40	7	23
12	38	5	40	23	7

выход за границы
массива



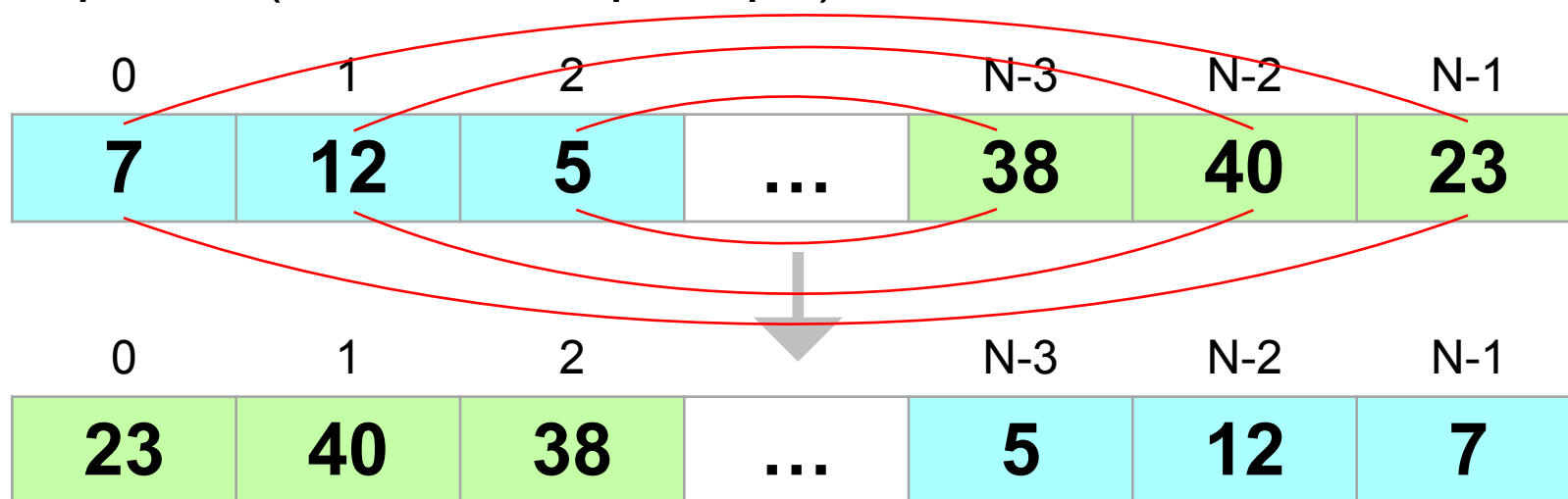
Перестановка пар соседних элементов

```
for i in range(0, N, 2):  
    # переставляем A[i] и A[i+1]  
    c = A[i]  
    A[i] = A[i+1]  
    A[i+1] = c
```

$A[1] \leftrightarrow A[2], A[3] \leftrightarrow A[4], \dots, A[N-1] \leftrightarrow A[N]$

Реверс массива

Задача. Переставить элементы массива в обратном порядке (выполнить *реверс*).



$$A[0] \leftrightarrow A[N-1]$$

$$0 + N - 1 = N - 1$$

$$A[1] \leftrightarrow A[N-2]$$

$$1 + N - 2 = N - 1$$

$$A[i] \leftrightarrow A[N-1-i]$$

$$i + ??? = N - 1$$

$$A[N-1] \leftrightarrow A[0]$$

$$N - 1 + 0 = N - 1$$

Реверс массива

```
for i in range(N//2):
```

поменять местами $A[i]$ и $A[N-1-i]$

0	1	2	3
7	12	40	23
23	12	40	7
23	40	12	7
23	12	40	7
7	12	40	23

$i=0$

$i=1$

$i=2$

$i=3$



Что плохо?



Как исправить?

Конец фильма

ПОЛЯКОВ Константин Юрьевич

д.т.н., учитель информатики

ГБОУ СОШ № 163, г. Санкт-Петербург

kpolyakov@mail.ru

ЕРЕМИН Евгений Александрович

к.ф.-м.н., доцент кафедры мультимедийной

дидактики и ИТО ПГГПУ, г. Пермь

eremin@pspu.ac.ru

Источники иллюстраций

1. иллюстрации художников издательства «Бином»
2. авторские материалы