

Python

Одномерные массивы Список

Массив - это упорядоченный набор однотипных элементов, имеющих общее имя.

a: 2, 5, 8, 45, 12, 46, 7, 10, 15, 19
№ 0 1 2 3 4 5 6 7 8 9
a[4]=12

a - имя массива

4 - порядковый номер элемента (номера всегда с 0)

a[4] - имя элемента (4-го)

12 - значение элемента

Описание массива

Объявление массива известного размера

$n=100$ задается размер массива

$a = [0]*n$ объявляется целочисленный массив a размером n и всем элементам массива присваивается нулевое значение

Объявление массива с неизвестным числом элементов (пустой массив, в нем нет элементов, поэтому нельзя к ним обращаться

например $a[3]$ вызовет ошибку)

$a = []$

Примеры заполнения массивов

Ввод значений непосредственно в программе

Способ №1.

$a = [5, 13, 22, 37, 45]$

Способ №2.

$a[0] = 5$

$a[1] = 13$

$a[2] = 22$

$a[3] = 37$

$a[4] = 45$

Способ 3. Оператором ввода с клавиатуры.

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

```
    a[i]=int(input())
```

{в качестве индекса используется параметр цикла }

Этот способ задания значений

используется для небольших массивов

Примеры заполнения массивов

Способ 4. Оператором ввода с клавиатуры.

```
a=[]
```

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

```
    a.append(int(input()))
```

a.append() - добавление элемента в конец массива

Примеры заполнения массивов

Способ 5. Заполнение массива с использованием генератора случайных целых чисел.

```
from random import randint
```

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

```
    a[i]=randint(-10,20)
```

{массив заполнится целыми числами в диапазоне от [-10 до 20]}

генератора случайных вещественных чисел (тип float)

```
import random
```

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

```
    a[i]=random.uniform(1,20)
```

Вывод массива на экран в строку

<pre>print(a) [1,2,3,4,5]</pre>	<pre>без скобок print(*a) 1 2 3 4 5</pre>
---------------------------------	---

Вывод массива на экран в столбик

<pre>for i in range(n): print(a[i])</pre>	<pre>for x in a: print(x)</pre>
---	-------------------------------------

№0 Заготовка.

Описание, заполнение случайными
числами, вывод массива

```
n=10
a=[0]*n
from random import randint
for i in range(n):
    a[i]=randint(1,100)
print(a)
```

№1

Дан массив целых чисел. Напечатать:

1.1 все четные элементы

```
print('четные элементы')
for i in range(n):
    if a[i]%2==0:
        print(a[i],end=' ')
```

Варианты условий

1.2 положительных элементов $a[i]>0$

1.3 нечетных элементов $a[i] \%2 \neq 0$

1.4 элементов, кратных 7 $a[i] \%7==0$

1.5 элементов, оканчивающихся на цифру 5 $a[i] \%10 ==5$

№2

Дан массив целых чисел. Заменить:

2.1 Все отрицательные элементы на их модуль

```
for i in range(n):
    if a[i]<0:
        a[i]=abs(a[i])
print('конечный массив')
print(a)
```

2.2 Все четные элементы заменить на их квадраты, а нечетные удвоить

```
for i in range(n):
    if a[i]%2==0:
        a[i]=a[i]*a[i]
    else:
        a[i]=a[i]*2
print('конечный массив')
print(a)
```

2.3 Все элементы, кратные числу 10, заменить на 1000.

```
for i in range(n):
    if a[i]%10==0:
        a[i]=1000
print('конечный массив')
print(a)
```

№3

Дан массив целых чисел. Найти количество:

3.1 четных элементов

```
k=0
for i in range(n):
    if a[i]%2==0:
        k+=1
print('Количество четных=',k)
```

3.2 элементов кратных 7

```
k=0
for i in range(n):
    if a[i]%7==0:
        k+=1
print('Количество кратных 7=',k)
```

3.3 элементов принадлежащих промежутку от x до y

```
x=int(input())
y=int(input())
k=0
for t in a:
    if x<t<y:
        k+=1
print('Количество четных=',k)
```

№4

Дан массив из 5 целых чисел. Найти:

4.1 сумму всех элементов массива

```
s=sum(a)
print('сумма всех=',s)
```

4.2 сумму трех первых элементов массива

```
s=0
for i in range(3):
    s=s+a[i]
print('сумма =',s)
```

4.3 сумму элементов массива с 2-го по 4-й

```
s=0
for i in range(2,5):
    s=s+a[i]
print('сумма =',s)
```

4.4 сумму нечетных элементов массива

```
s=0
for i in range(n):
    if a[i]%2==1:
        s=s+a[i]
print('сумма нечетных=',s)
```

Classic style

```
s=0
for i in range(n):
    s=s+a[i]
print('сумма всех =',s)
```

Поиск максимального элемента массива

a: 2, 5, 8, 44, 12, 46, 7, 10, 14, 19

Допускаем что максимум 0-й элемент $ma=a[0]$ и перебираем все остальные и сравниваем с ma и если элемент массива больше чем ma меняем ma на этот элемент в итоге $ma=46$

№5.1

Дан массив целых чисел. Найти:
максимальный (минимальный) элемент массива (Classic style)

```
ma=a[0]
for i in range(1,n):
    if a[i]>ma:
        ma=a[i]
print('Максимальный элемент=',ma)
```

Функция

```
ma=max(a)
print('Максимальный элемент=',ma)
mi=min(a)
print('Минимальный элемент=',mi)
```

Поиск номера максимального элемента

a : 2, 5, 8, 44, 12, 46, 7, 10, 14, 19

Допускаем что максимум 0-й элемент $imax=0$ и перебираем все остальные и сравниваем $a[i]>a[imax]$ и если элемент массива больше чем $a[imax]$ меняем $imax$ на номер этого элемента $imax=i$
в итоге $imax=5$

№5.2

Дан массив целых чисел. Найти номер максимального элемента (Classic style)

```
imax=0
for i in range(1,n):
    if a[i]>a[imax]:
        imax=i
print('Номер максимального элемента=',imax)
```

Функция

```
imax=a.index(max(a))
print('Номер максимального элемента=',imax)
```


№6

Дан массив целых чисел. Найти

1 число пар "соседних" элементов массива, являющихся четными числами

```
k=0
for i in range(n-1):
    if a[i]%2==0 and a[i+1]%2==0:
        k+=1
print('Количество четных пар =',k)
```

2 число пар "соседних" элементов массива, оканчивающихся 0

```
k=0
for i in range(n-1):
    if a[i]%10==0 and a[i+1]%10==0:
        k+=1
print('Количество пар =',k)
```

3 число элементов массива, которые больше своих "соседей", т.е. предшествующего и последующего (локальный максимум)

```
k=0
for i in range(1,n-1):
    if a[i]>a[i-1] and a[i]>a[i+1]:
        k+=1
    print(a[i])
print('Количество больших своих соседей =',k)
```

Задачи для тренировки к тесту

Дан массив: 3 5 13 2 15 12 43 45 32 23

Определите что выведет программа на экран

```
for i in range(10):
    if a[i]%5==0:
        print(a[i])

for i in range(7):
    if a[i]%10==3:
        print( a[i])
```

Задачи для тренировки к тесту

Дан массив: 4 5 3 25 13

Определите что выведет программа на экран

```
for i in range(5):  
    if a[i]%5==0:  
        a[i]=555  
print(a)
```

```
for i in range(5):  
    if a[i]%10==3:  
        a[i]=100  
print(a)
```

Задачи для тренировки к тесту

Дан массив. 3 -5 7 2 -3 12 34 45 -32 23

Определите что выведет программа на экран

```
s=0  
for i in range(10):  
    if a[i]<0:  
        s+=1  
print(s) 
```

```
s=0  
for i in range(7):  
    if a[i]%2==1:  
        s+=1  
print(s) 
```

Задачи для тренировки к тесту

Дан массив. 3 -5 7 2 -3 12 34 45 -32 -56 23

Определите что выведет программа на экран

s=0

for i in range(3):

s=s+a[i]

print(s)

s=0

for i in range(2,5):

s=s+a[i]

print(s)

min(A)	Наименьший элемент списка. Элементы списка могут быть числами или строками, для строк сравнение элементов проводится в лексикографическом порядке.
max(A)	Наибольший элемент списка.
sum(A)	Сумма элементов списка. Элементы обязательно должны быть числами.
A.sort()	Сортирует список (по возрастанию)
A.sort (reverse=True)	Сортирует список (по убыванию)
A.reverse()	Разворачивает порядок в списке
x in A	Проверить, содержится ли элемент в списке. Возвращает True или False . [x not in A то же самое, что not(x in A).]
A.index(x)	Индекс первого вхождения элемента x в список, при его отсутствии генерирует исключение ValueError.
A.count(x)	Количество вхождений элемента x в список.
A.append(x)	Добавить в конец списка A элемент x.
A.insert(i, x)	Вставить в список A элемент x на позицию с индексом i. Элементы списка A, которые до вставки имели индексы i и больше, сдвигаются вправо.
B=A.copy()	Возвращает копию списка (B копия A! Если B=A, то B будет клон A и при изменении A будет меняться и B)
A.extend(B)	Добавить в конец списка A содержимое списка B.
A.pop()	Удалить из списка последний элемент, возвращается значение удалённого элемента.
A.pop(i)	Удалить из списка элемент с индексом i, возвращается значение удаленного элемента. Все элементы, стоящие правее удаленного, сдвигаются влево.
A.remove(x)	Удаляет первое вхождение элемента x. Если элемента x нет в списке, то получаем ошибку во время выполнения.
A. clear()	Удаляет все элементы в списке

Сортировка_Выбором.pas

Сортировка_Пузырек.pas