



ОДНОМЕРНЫЕ МАССИВЫ

АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ

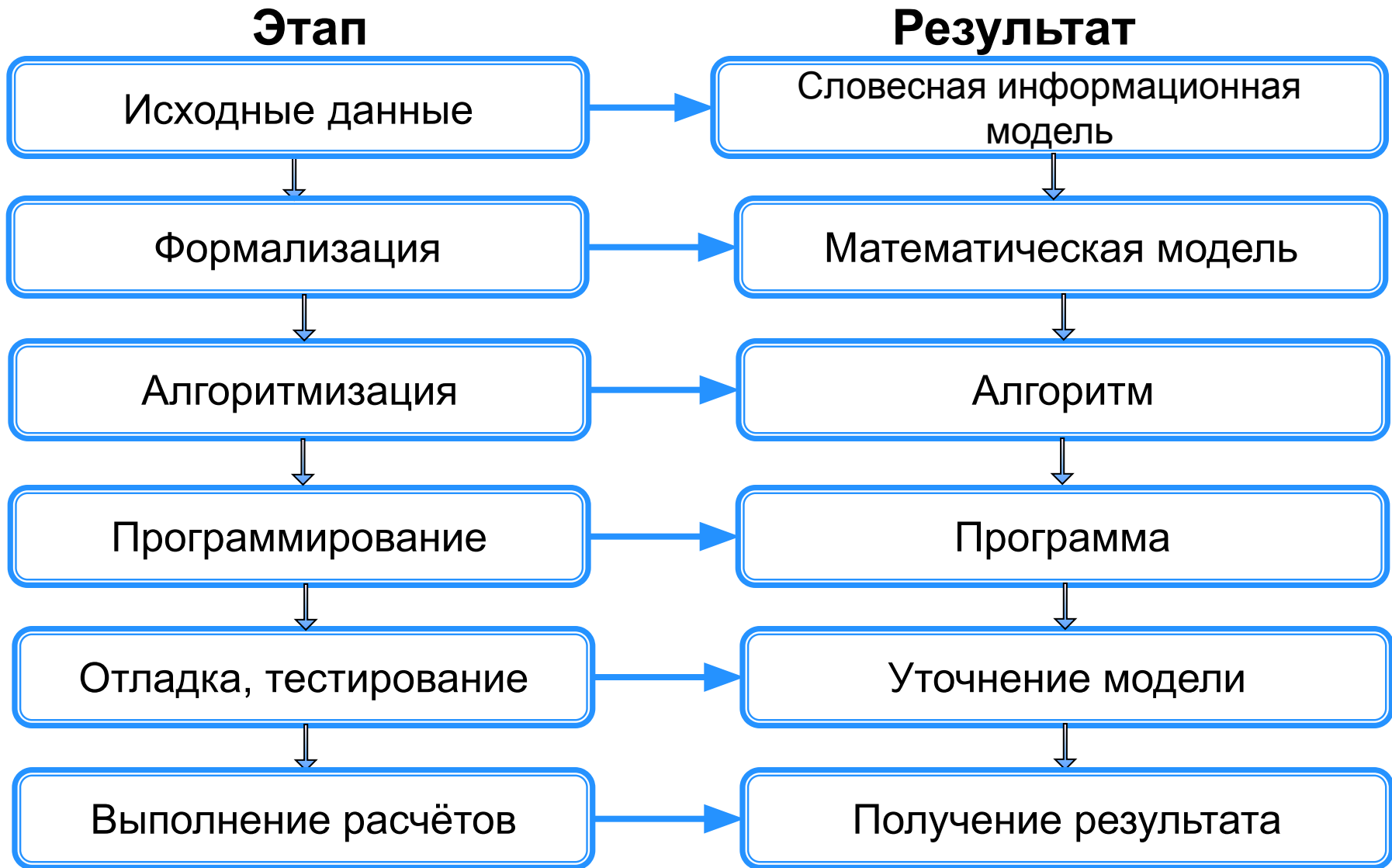
9 класс



ИЗДАТЕЛЬСТВО

БИНОМ

Этапы решения задач на компьютере



Проблемный вопрос

Надо расположить команды в турнирной таблице футбольного чемпионата в порядке убывания набранных ими очков. Команд немного, всего 16. После каждого тура количество набранных ими очков меняется и сортировку надо выполнять заново.

Что будем делать?

Будущая программа должна принимать с клавиатуры очки, набранные командами, и распечатывать команды в порядке убывания этих чисел.

Актуализация знаний

Мы умеем сравнивать два числа. Используем алгоритм сравнения двух чисел для сортировки двух команд, например «Зенит» и «Спартак». Программа будет выглядеть так:

Pascal

```
var T1, T2 : integer;  
begin  
  readln (T1, T2);  
  if T1>T2 then  
    begin  
      writeln ('1.Зенит');  
      writeln ('2. Спартак');  
    end;  
  else begin  
    writeln ('1. Спартак');  
    writeln ('2. Зенит');  
  end;  
end.
```

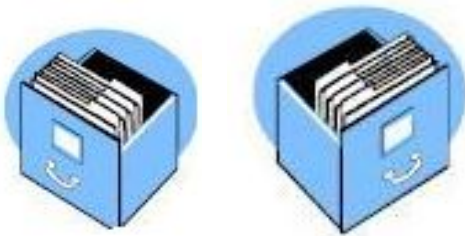
Python

```
T1 = int(input())  
T2 = int(input())  
if T1>T2:  
    print ('1.Зенит')  
    print ('2. Спартак')  
else:  
    print ('1. Спартак');  
    print ('2. Зенит');  
end.
```



Что такое массив?

Простые
переменные



Массивы
переменных



Массив

Массив – это совокупность элементов, которым присвоено общее имя, доступ к отдельному элементу массива осуществляется по его номеру (индексу)

*Python 3: **СПИСОК = МАССИВ***

Одномерный список

| Значение элемента списка | | | | | | Индекс элемента списка | | | |
|--------------------------|----|---|---|---|----|------------------------|---|---|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | |
| 7 | -5 | 9 | 1 | 0 | -2 | 4 | 3 | 6 | |

Решение разнообразных задач, связанных с обработкой списков, базируется на решении таких типовых задач, как:

- суммирование элементов списка;
- поиск элемента с заданными свойствами;
- сортировка списка.

Способы заполнения списка

1 способ. Явное заполнение

a = [] # пустой список

b = [1, -2, 3, 5, 0] # список из пяти элементов одного типа

список из пяти элементов разного типа:

my = ['abc', 'абв', 5, '78', 643]

2 способ Ввод каждого значения с клавиатуры

N=10 # размер списка

A=[0]*10

for i in range(N):

A[i] = (int (input())) # добавить в каждый элемент
списка



Способы заполнения списка

3 способ. Заполнение случайными числами

```
from random import randint
# подключить библиотеку случайных чисел
N = 10 # размер списка
A=[ 0]*10
```

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

```
    A[i] = randint(0,99) #присваивание очередному
                        элементу случайного числа от 0 до 99
```

4 способ. С помощью оператора присваивания (по формуле):

```
for i in range(10):
```

```
    A[i] = i**2 # элемент равен квадрату индекса
```

| | | | | | |
|----|----|----|----|----|----|
| 37 | 58 | 45 | 58 | 58 | 38 |
| 38 | 59 | 46 | 59 | 59 | 39 |
| 39 | 01 | 47 | 01 | 01 | 01 |
| 40 | 02 | 48 | 02 | 02 | 02 |

$a[i] = 2 * a[i] + 6$

Вывод списка

1 способ. Элементы массива можно вывести в строку, разделив их запятой:

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

```
    print ( A[i], end = ',' ) или print (*A, sep = ',')
```

```
45, 21, 56, 43, 83, 56, 69, 34, 28, 15
```

2 способ. Вывод в столбик:

```
for i in range(10):
```

```
    print (A[i])
```

Вывод с комментариями:

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

```
    print ('A[' , i, ' ]= ',A[i])
```

```
A[ 0 ]= 4
```

```
A[ 1 ]= 1
```

```
A[ 2 ]= 6
```

```
A[ 3 ]= 3
```

```
A[ 4 ]= 8
```

```
A[ 5 ]= 5
```

```
A[ 6 ]= 9
```

```
A[ 7 ]= 4
```

```
A[ 8 ]= 8
```

```
A[ 9 ]= 7
```

Заполнение списка случайными числами и вывод элементов списка

N = 10

A = [0]*N

**Заполнение
списка**



Вывод списка

```
from random import randint
```

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

```
    A[i] = randint(0,99)
```

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

```
    print ('A[' , i, ' ]= ',A[i])
```

Самое главное

Список в Python - упорядоченный изменяемый набор элементов произвольных типов (почти как массив, но типы могут отличаться). В языках программирования списки (массивы) используются для реализации таких структур данных, как последовательности и таблицы.

Перед использованием в программе списки необходимо создать.

Заполнять списки можно разными способами: с клавиатуры, присваивая элементам некоторые значения в программе. При заполнении списка и выводе его элементов на экран используется цикл с параметром.



Опорный конспект

Список в Python - упорядоченный изменяемый набор элементов произвольных типов (почти как массив, но типы могут отличаться).

Заполнение списка

Ввод с клавиатуры

Присваивание значений

Задачи по обработке списка

Суммирование элементов

Сортировка элементов списка

Поиск элементов списка с заданными свойствами