



# СОРТИРОВКА ЭЛЕМЕНТОВ МАССИВА

## Урок 1

Учитель: Н.В. Фоменко

## АКТУАЛИЗАЦИЯ ОПОРНЫХ ЗНАНИЙ

- Что такое массив?
- Как найти максимальный (минимальный) элемент массива?
- Как обменять местами два элемента массива?
- Как найти второй максимум?
- Как подсчитать количество максимумов?



# ПОНЯТИЕ СОРТИРОВКИ

- Сортировка – один из наиболее распространенных процессов обработки данных
- Сортировка массива – это **упорядочение** его элементов по **возрастанию** или **убыванию**
- Порядок, при котором в массиве первый элемент имеет самое маленькое значение, а значение каждого последующего элемента не меньше значения предыдущего, называется **неубывающим (возрастающим)**
- Порядок, при котором в массиве первый элемент имеет самое большое значение, а значение каждого последующего элемента не больше значения предыдущего, называется **невозрастающим (убывающим)**



# ПУЗЫРЬКОВАЯ СОРТИРОВКА

1. Просматриваются слева направо все пары соседних элементов:  $a_1$  и  $a_2$ ,  $a_2$  и  $a_3 \dots a_{n-1}$  и  $a_n$
2. Если  $a_i > a_{i+1}$ , то они меняются местами
3. В результате такого просмотра массива максимальный элемент окажется на крайнем правом (своем) месте – «всплывает как пузырек»

Первый просмотр

50	40	10	20	30	7
----	----	----	----	----	---



40	50	10	20	30	7
----	----	----	----	----	---



40	10	50	20	30	7
----	----	----	----	----	---



40	10	20	50	30	7
----	----	----	----	----	---



40	10	20	30	50	7
----	----	----	----	----	---



40	10	20	30	7	<b>50</b>
----	----	----	----	---	-----------

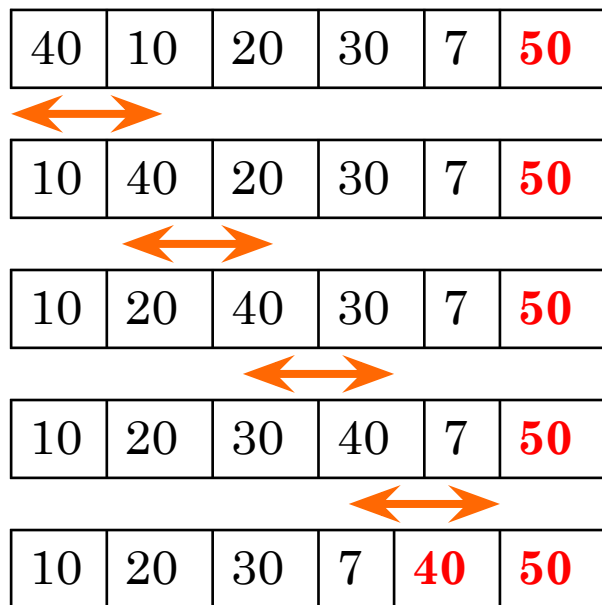
```
for j:=1 to n -1 do
  if a[j]>a[j+1] then
    begin
      t:=a[j]; a[j]:=a[j+1]; a[j+1]:=t
    end;
```



# ПУЗЫРЬКОВАЯ СОРТИРОВКА

4. Затем массив просматривается снова за исключением крайнего правого элемента. В результате второй по величине элемент окажется на предпоследнем месте

Второй просмотр



```
for j:=1 to n -2 do
  if a[j]>a[j+1] then
    begin
      t:=a[j]; a[j]:=a[j+1]; a[j+1]:=t
    end;
```



# ПУЗЫРЬКОВАЯ СОРТИРОВКА

5. Так продолжается, пока весь массив не окажется упорядоченным. В последнем просмотре будут участвовать только первый и второй элементы

Третий просмотр

10	20	30	7	40	50
----	----	----	---	----	----



10	20	7	30	40	50
----	----	---	----	----	----

Четвертый просмотр

10	20	7	30	40	50
----	----	---	----	----	----



10	7	20	30	40	50
----	---	----	----	----	----

Пятый просмотр

7	10	20	30	40	50
---	----	----	----	----	----



```
for i:=1 to n -1 do
```

```
  for j:=1 to n -1 do
```

```
    if a[j]>a[j+1] then
```

```
      begin
```

```
        t:=a[j]; a[j]:=a[j+1]; a[j+1]:=t
```

```
      end;
```

По другому пузырьковая сортировка называется **обменной**



# СОРТИРОВКА ПРЯМЫМ ВЫБОРОМ

- Находится минимальный элемент в массиве и меняется местами с первым элементом
- Затем находится минимальный элемент среди оставшихся и меняется местами со вторым и т.д.

Первый просмотр

40	20	10	50	30	7
----	----	----	----	----	---



7	20	10	50	30	40
---	----	----	----	----	----

Второй просмотр

7	20	10	50	30	40
---	----	----	----	----	----



7	10	20	50	30	40
---	----	----	----	----	----

```
imin:= 1;  
for j:=2 to n do  
    if a[j]<a[imin] then imin:=j;  
t:=a[ 1 ]; a[ 1 ]:=a[imin]; a[imin]:=t;
```



# СОРТИРОВКА ПРЯМЫМ ВЫБОРОМ

Третий просмотр

7	10	20	50	30	40
---	----	----	----	----	----

Четвертый просмотр

7	10	20	50	30	40
---	----	----	----	----	----



7	10	20	30	50	40
---	----	----	----	----	----

Пятый просмотр

7	10	20	30	50	40
---	----	----	----	----	----



7	10	20	30	40	50
---	----	----	----	----	----

```
for i:=1 to n-1 do
begin
  imin:= i ;
  for j:=i+1 to n do
    if a[j]<a[imin] then imin:=j;
  t:=a[ i ]; a[ i ]:=a[imin]; a[imin]:=t;
end;
```

Сколько просмотров получилось?

