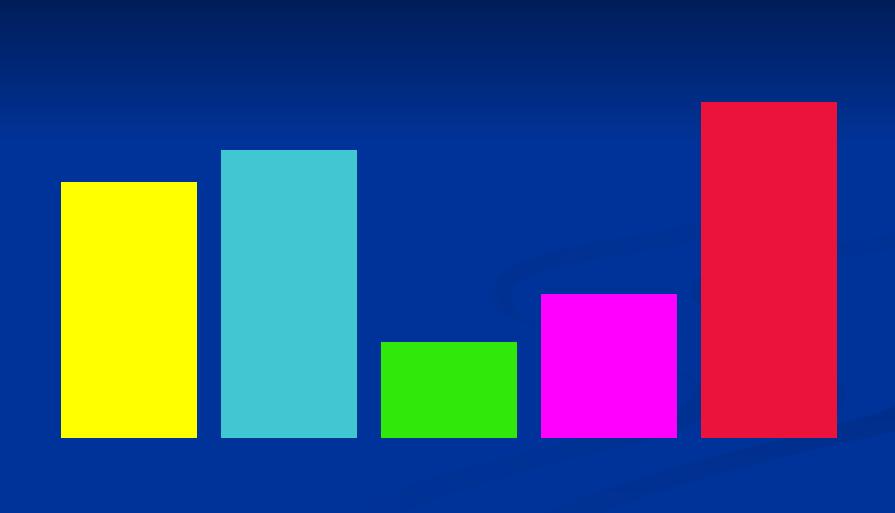
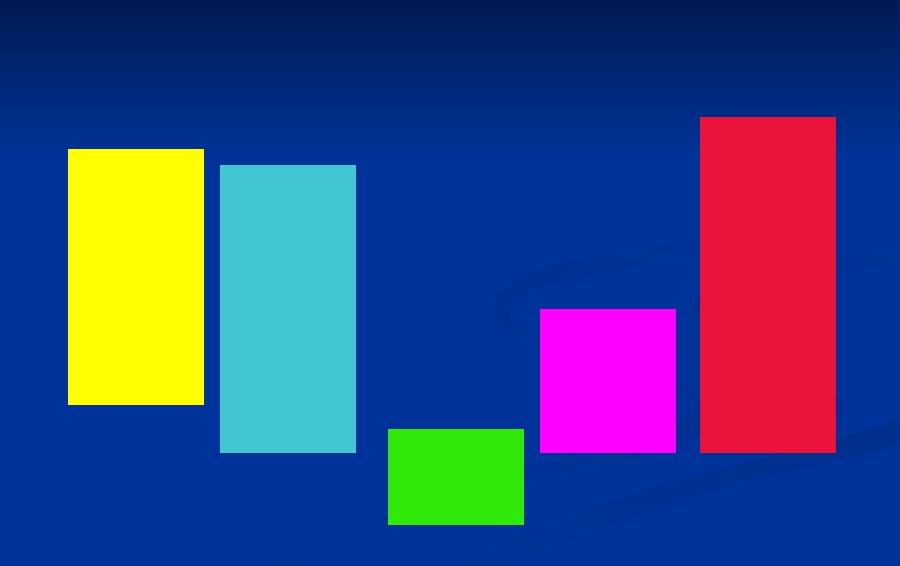
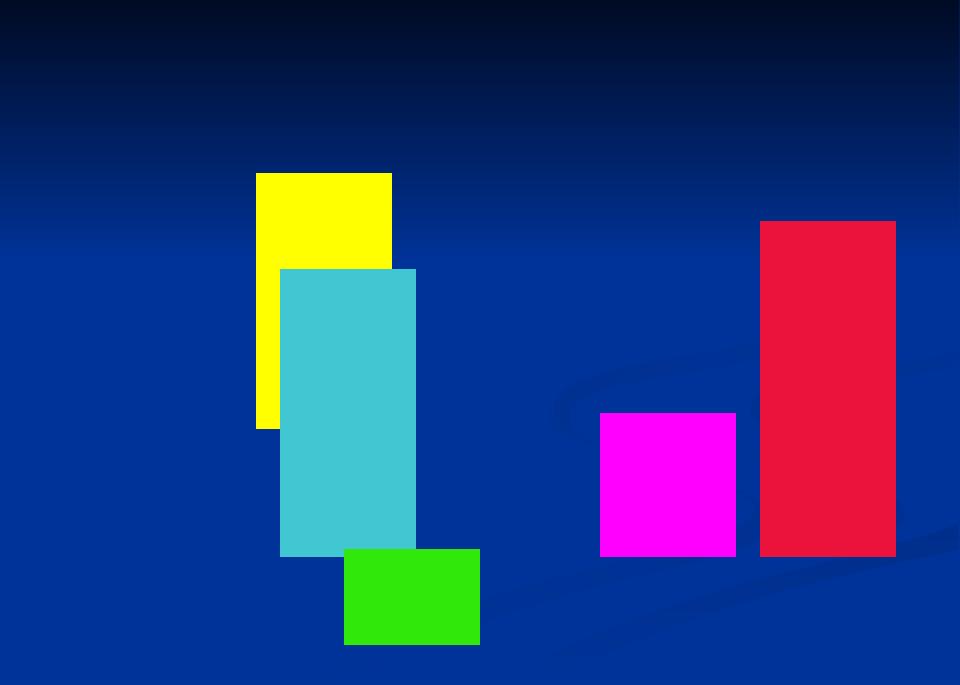
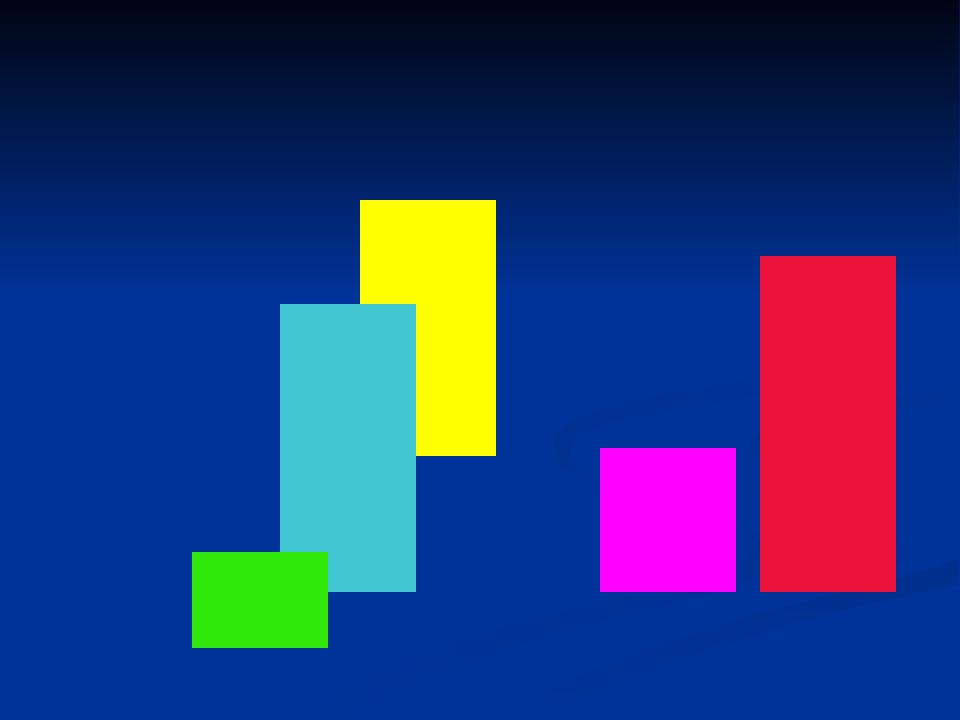
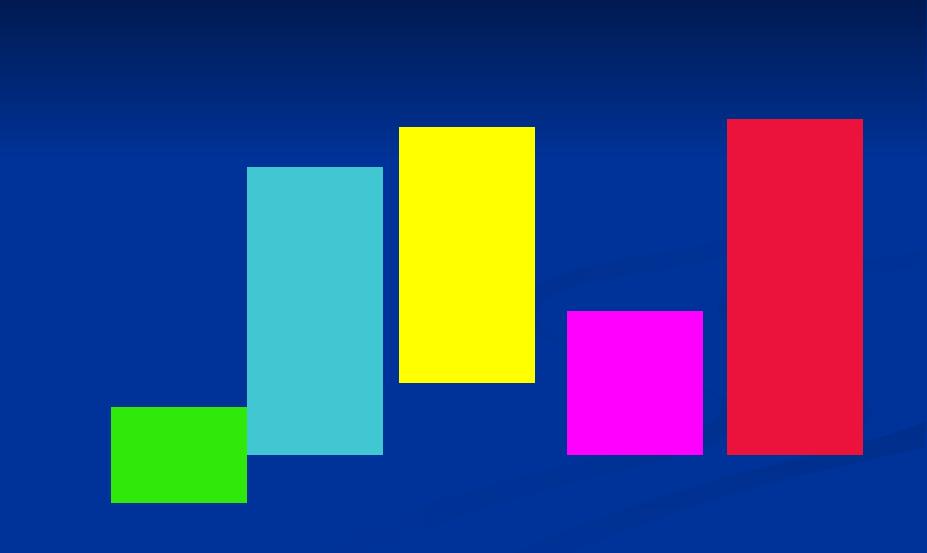
### Сортировка массивов



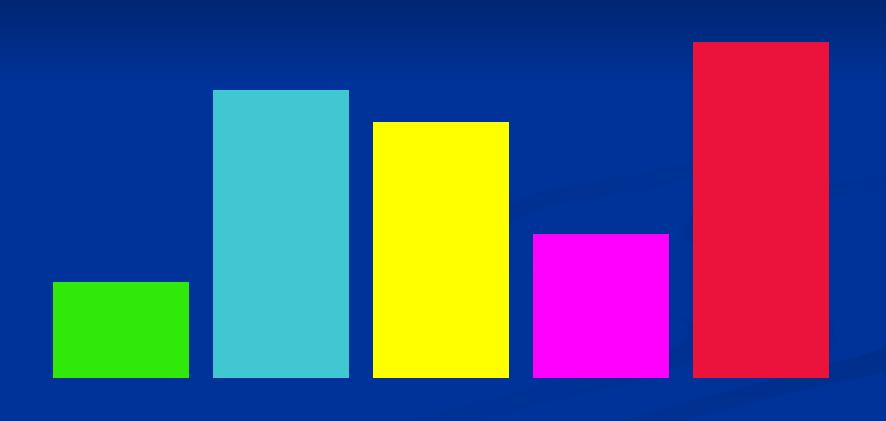




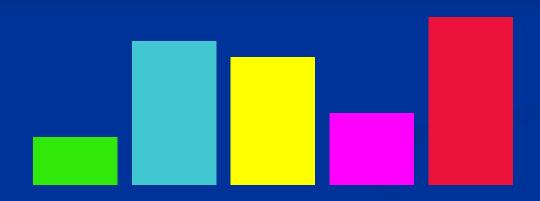




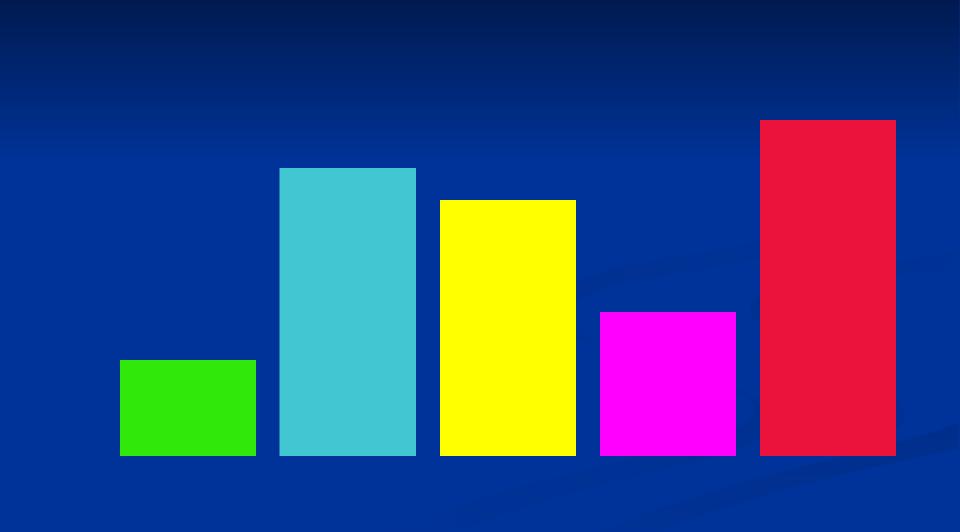
### Что изменилось?

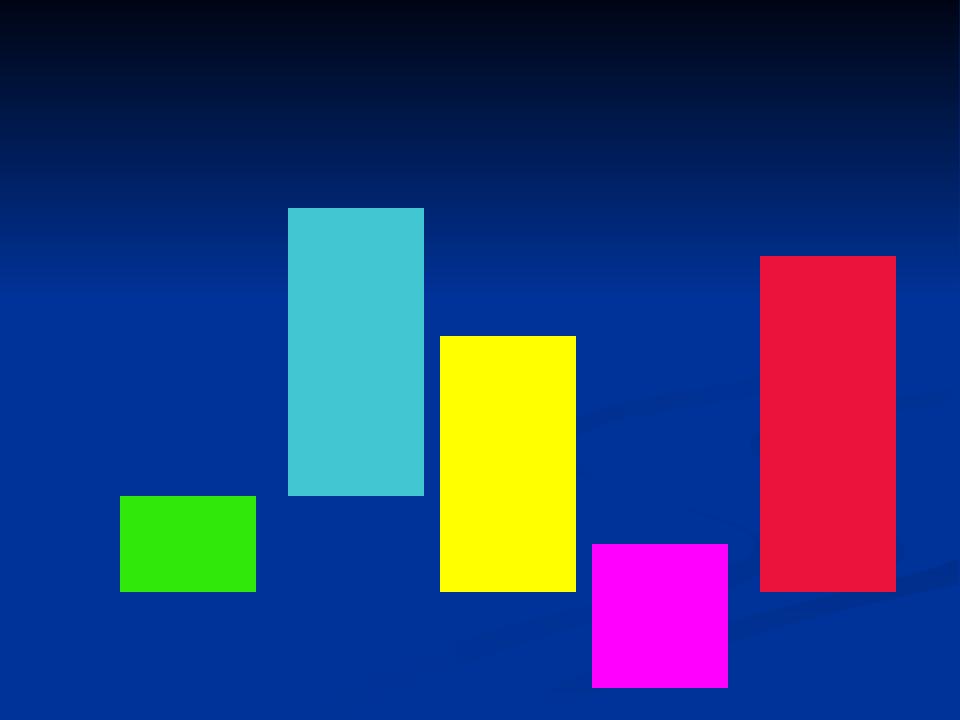


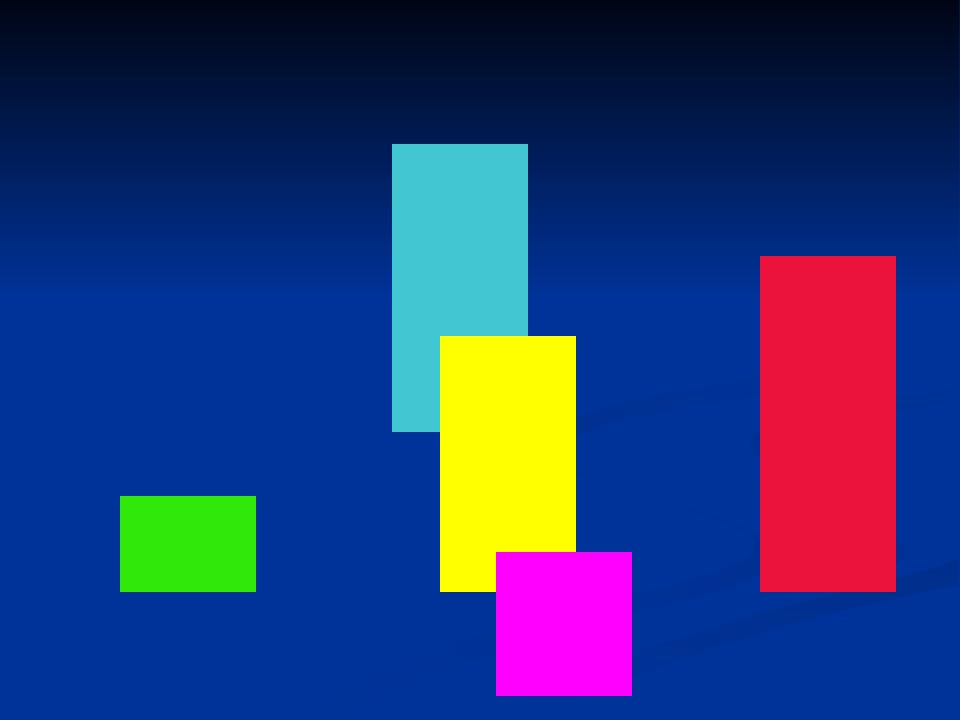
### ЧТО ДАЛЬШЕ?

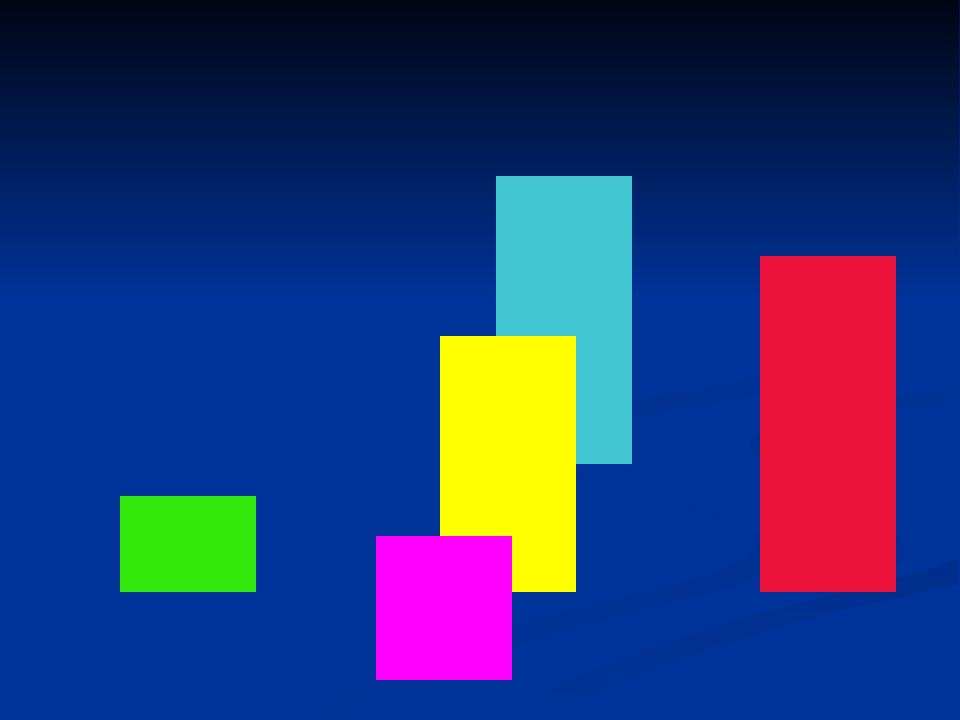


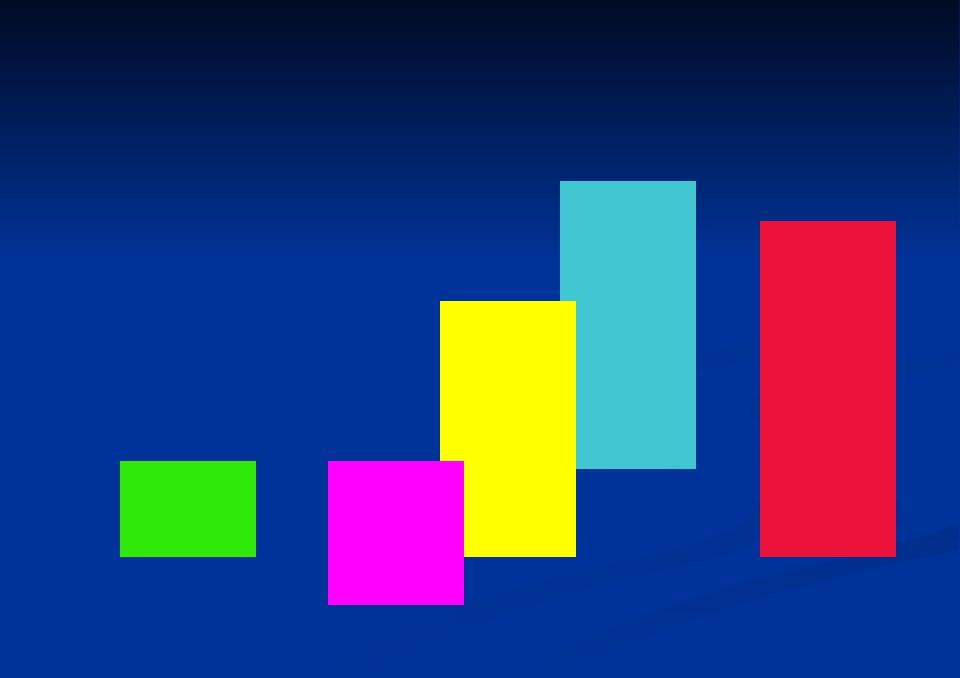
Поменяем местами голубой и лиловый прямоугольники.



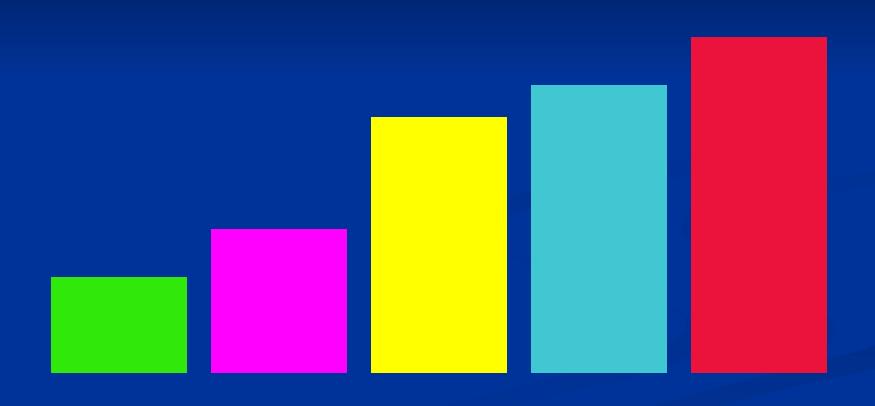








#### Все прямоугольники расположены в порядке увеличения



## Задача этого урока – рассмотреть алгоритм сортировки массива по возрастанию.

- Необходимость отсортировать какие-либо величины возникает в программировании очень часто.
- Существует разные способы сортировки массивов.

# Сформулируйте определение сортировки

Сортировка - это процесс упорядочения заданного множества объектов в некотором, заранее определённом порядке.

### Рассмотрим один из алгоритмов сортировки

### Принцип метода:

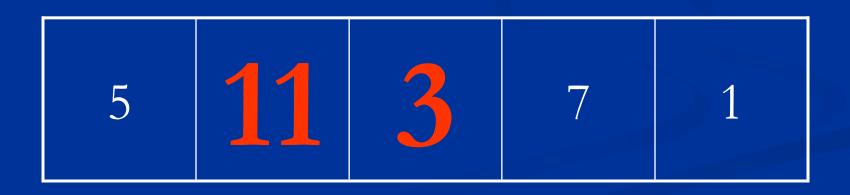
Слева на право поочерёдно сравниваются два соседних элемента,

<b>11 5</b> 3	7	1
---------------	---	---

Если их взаимное расположение не соответствует заданному условию упорядоченности, то они меняются местами



Далее берутся два следующих соседних элемента и так до конца массива



После одного прохода на последней *n*-ой позиции массива будет стоять максимальный элемент

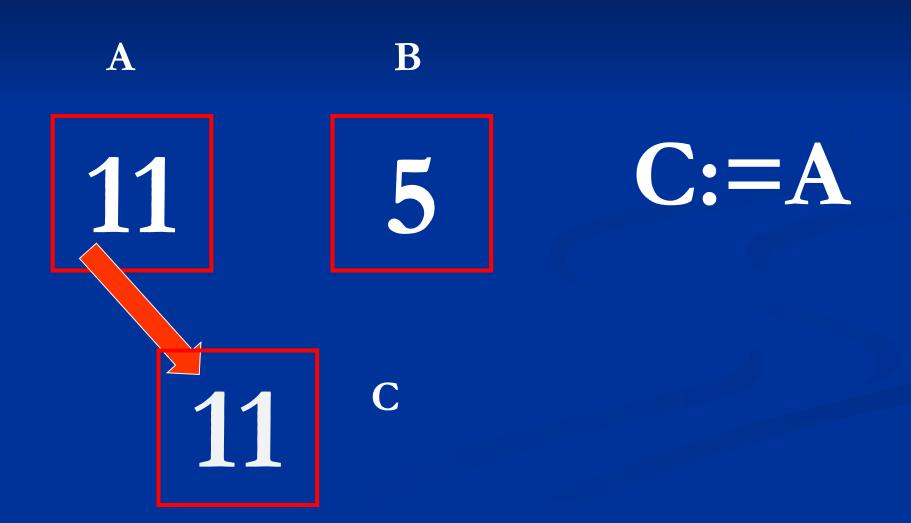
(«всплыл» первый «пузырёк»)

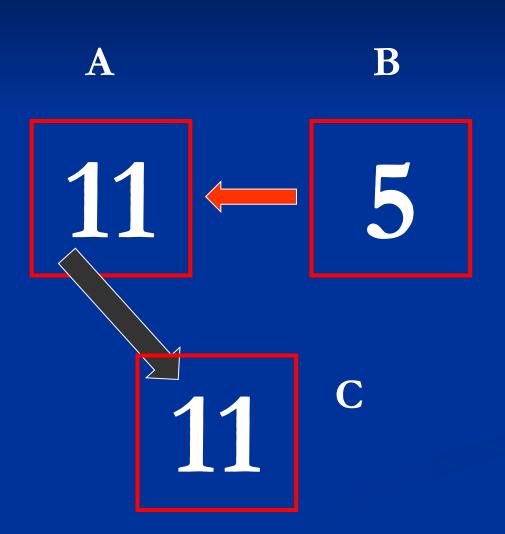
5	3	7	1	11
---	---	---	---	----

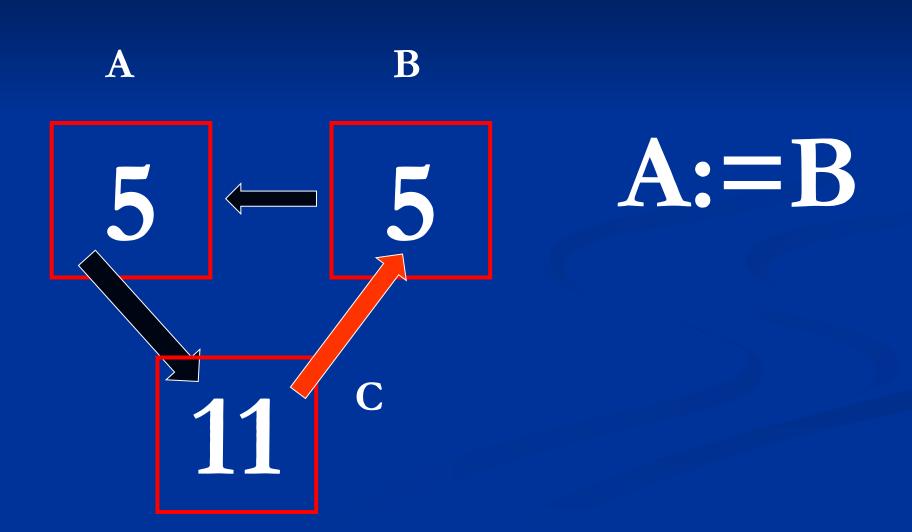
Поскольку максимальный элемент уже стоит на своей последней позиции, то второй проход обменов выполнятся до n-1 — го элемента.

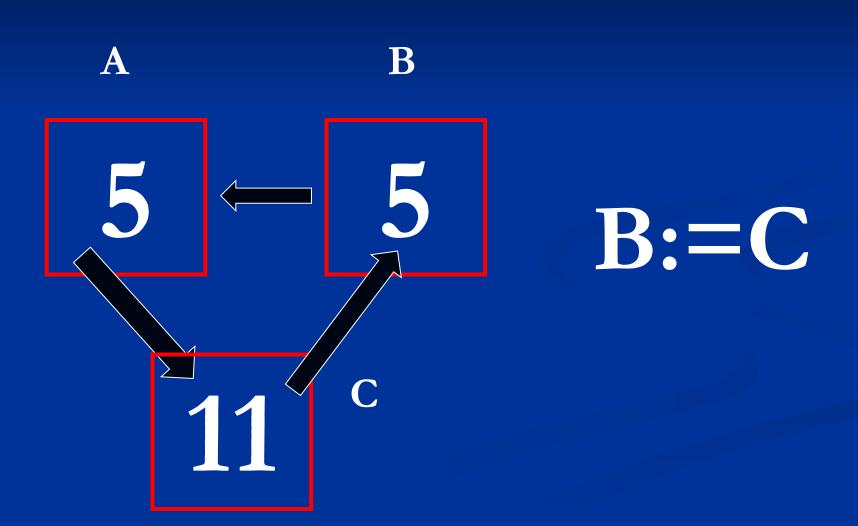
5 3 7 1 1					$\mathcal{N}$
	5	3	7	1	11

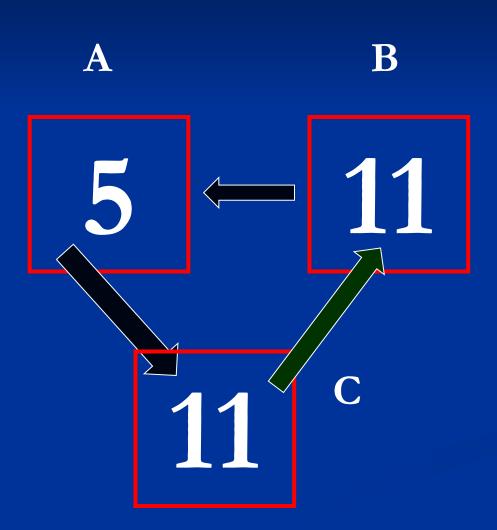
## Для реализации этого метода сортировки будем использовать алгоритм перестановки











### Схема алгоритма:

11 =	<u> </u>	3	7	1
5	11 =	<u> </u>	7	1
5	3	11 -	<del>-</del> 7	1
5	3	7	11	<del>1</del>
5	3	7	1	11

5 —	<u></u>	7	1	11
3	5	7	1	11
3	5	7 =	1	11
3	5	1	7	11

2

11 Первый и второй элементы стоят на своих местах

 3
 1
 5
 7
 11

В результате перестановок мы получим отсортированный по возрастанию массив

1 3 5 7 11



Данный массив отсортирован по возрастанию



Данный массив отсортирован по не убыванию



Данный массив отсортирован по убыванию

8 7 7 1 1

Данный массив отсортирован по не возрастанию

### Программа на Pascal

```
i:=1;
       repeat
         if Vector[i] > Vector[i+1] then
                        begin
                                     B:=Vector[i];
                                         Vector[i]:=Vector[i+1];
                                         Vector[i+1]:= B;
                                        end;
       i:=i+1;
       until i>5-k;
```

### Программа на Pascal

```
for k:=1 to 4 do
       begin
•
   i:=1;
          repeat
             if Vector[i] > Vector[i+1] then
                           begin
                                       B:=Vector[i];
                                            Vector[i]:=Vector[i+1];
                                            Vector[i+1]:= B;
                                          end;
i:=i+1;
          until i>5 - k;
   end;
```

### Программа, реализующая данный алгоритм

```
uses Crt;
     type TVector=array [1..5] of real;
•
    var Vector:Tvector;
        B: real;
        i,k:integer;
     begin
      Clrscr;
     for i:=1 to 5 do Read (Vector[i]);
       for k:=1 to 4 do
•
        begin
           i:=1;
           repeat
               if Vector[i] > Vector[i+1] then
                                 begin
                                               B:=Vector[i];
                                               Vector[i]:=Vector[i+1];
                                               Vector[i+1]:= B;
                                             end;
            i:=i+1;
            until i>5-k;
        end;
     for i:=1 to 5 do write(Vector[i]:8:2);
     end.
```