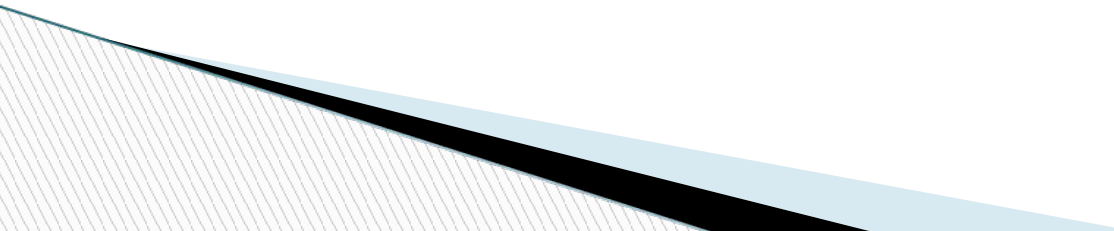


# Обработка массивов

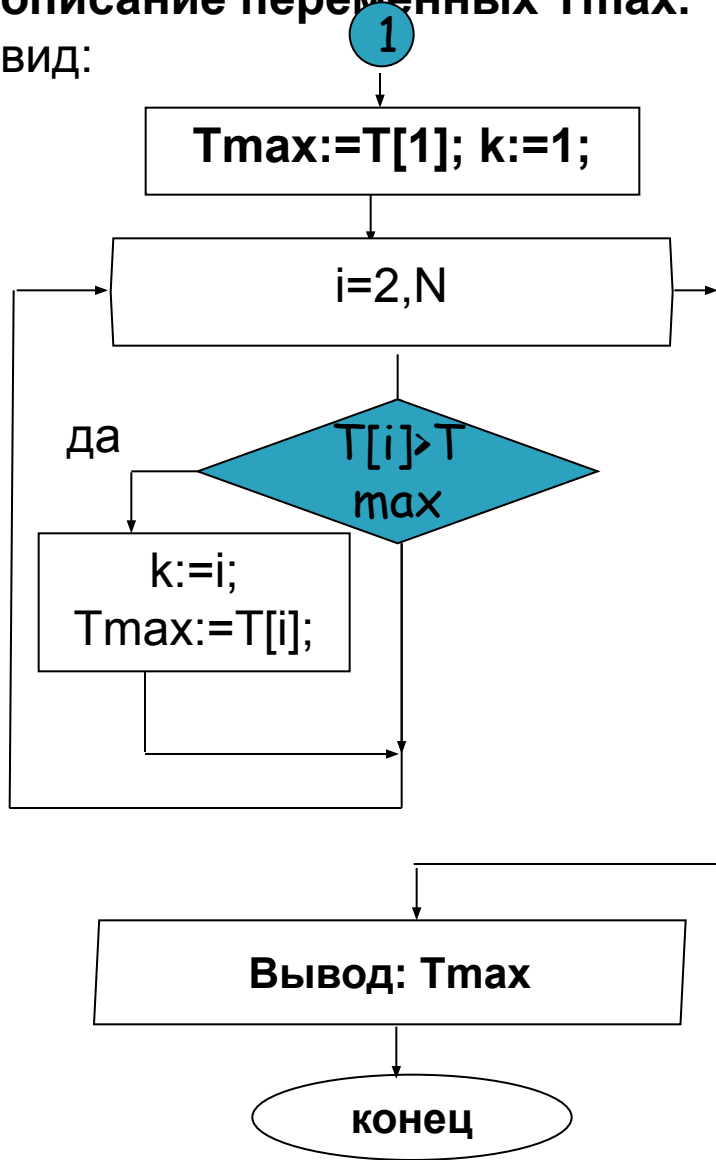
# Обработка массива:

1. Сложение элементов массива .
  2. Поиск элементов по заданному признаку.
  3. Поиск максимального (минимального) элементов.
  4. Сортировка элементов массива.
- 

### 3. Поиск максимального (минимального) элементов.

**Пример М\_О\_3:** Определение максимальной температуры

Для заполнения массива можно использовать **пример М\_2**, добавив в описание переменных **Tmax**. Тогда основная часть задачи имеет следующий вид:



```
Tmax:=T[1]; k:=1;
```

```
For i:=2 to N do
```

```
begin
```

```
if T[i]>Tmax then
```

```
begin
```

```
k:=i;
```

```
Tmax:=T[i];
```

```
end;
```

```
end;
```

```
writeln(' Tmax= ',Tmax, k:3, ' -ий№');
```

```
End.
```

Пример

М\_ОМ\_О\_М\_О

3

**Поменяем местами найденное максимальное значение и первое значение массива, распечатаем полученный массив.**

{обработка массива: ищем максимум }

Tmax:=T[1]; k:=1;

For i:=2 to N do begin

    If T[i]>Tmax then begin

        k:=i;

        Tmax:=T[i];

    end;

end;

{вывод результата обработки массива}

textcolor(4);

writeln('Tmax =',Tmax:5:1, k:3,' номер');

{меняем местами найденное максимальное значение с первым элементом}

T[k]:=T[1];

T[1]:=Tmax;

{вывод нового массива, в котором первый элемент наибольший}

textcolor(1);

For i:=1 to N do   Writeln(i:3,T[i]:9:1);

End.

# Сортировка

---

**Сортировка** - это расстановка элементов массива в заданном порядке (по возрастанию, убыванию, последней цифре, сумме делителей, ...).

**Задача:** переставить элементы массива в порядке возрастания.

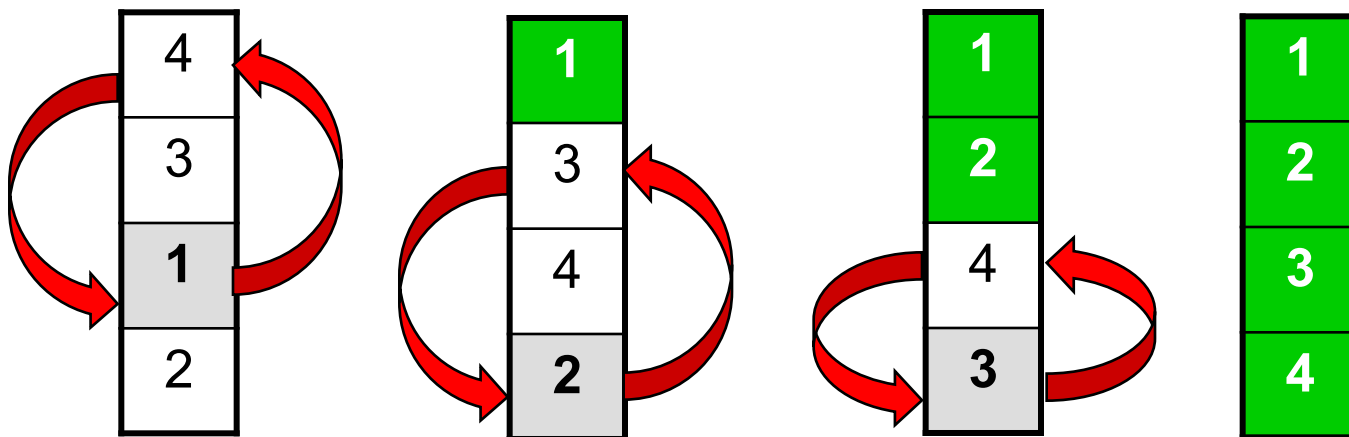
**Алгоритмы:**

- сортировка выбором
- сортировка обменом - «пузырьковая»
- сортировка вставками
- сортировка подсчетом

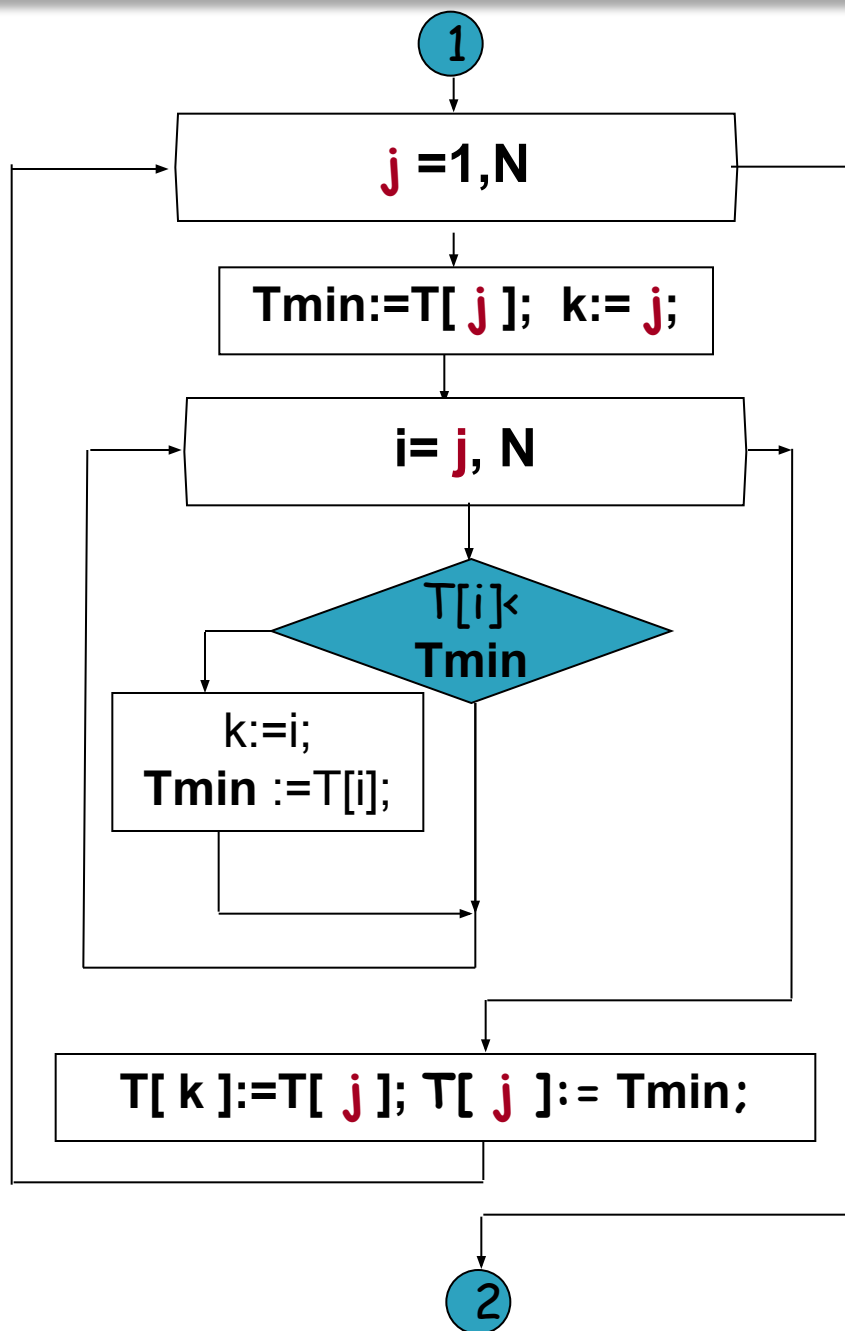
# Метод выбора

## Идея:

- найти минимальный элемент и поставить на первое место (поменять местами с  $A[1]$ )
- из оставшихся найти минимальный элемент и поставить на второе место (поменять местами с  $A[2]$ ), и т.д.



## 4. Сортировка элементов массива.



### Пример М\_О\_4:

Массив, заполнен случайными числами из диапазона от -20 до 20, количество элементов вводится с клавиатуры. Расположить элементы массива по возрастанию

## 4. Сортировка элементов массива.

**Пример M\_O\_4:** Массив, заполнен случайными числами из диапазона от -20 до 20, количество элементов вводится с клавиатуры. Расположить элементы массива по возрастанию.

### 1 часть задачи - описание и заполнения массива

```
program M_O_4sor;
uses crt;
Var
  T:array[1..365] of real;
  k,N,i ,j: integer;
Tmin :real;
Begin
  Write( 'Введите количество дней ' );
  Readln (N);
  textcolor(1);
  For k:=1 to N do
    begin
      {заполнение и печать в одном цикле}
      T[k] := -20 + Random(41);
      Writeln(k:3,T[k]:9:1);
    end;
```

**2 часть задачи -**  
**обработка массива**  
показана далее



{2 часть задачи – сортировка массива }

**For j:=1 to N do**

begin

**Tmin :=T[ j ];**

k:= j ;

For i:=j+1 to N do

begin

If T[i]< **Tmin** then

begin

k:=i;

**Tmin :=T[i ];**

end;

end;

**T[k]:=T[ j ];**

**T[ j ]:= Tmin;**

**end;**

{3 часть задачи - ВЫВОД НОВОГО МАССИВА}

For k:=1 to N do Writeln(k:3,T[k]:9:1);

End.

# Подведем итоги

Мы рассмотрели 4 вида обработки массива:

- 1 Сложение элементов.
- 2 Поиск элементов по заданному признаку.
3. Поиск максимального (минимального) элементов.
4. Сортировка элементов массива.

## Задания:

**1** Наберите программы и запустите их на выполнение.

**2.** Переделайте программу для производства убывающей сортировки

В программе поиска минимального и максимального элемента. Вывести массив, значения минимума синим цветом, максимума – красным.