

Лекция 10

Типовые алгоритмы обработки массивов

Преподаватель Палехова Ольга Александровна,
кафедра О7 БГТУ «Военмех»

Максимальный элемент

Значение максимума

```
max = a[0];  
for ( i = 1 ; i < N ; i++ )  
    if ( a[i] > max )  
        max = a[i];
```

Индекс максимального элемента

```
imax = 0;  
for ( i = 1 ; i < N ; i++ )  
    if ( a[i] > a[imax] )  
        imax = i;
```

Адрес максимального элемента

```
pmax = &a[0];  
for ( i = 1 ; i < N ; i++ )  
    if ( a[i] > *pmax )  
        pmax = &a[i];
```

Максимальный элемент

Единственный ли максимум?

```
pMax = a; flag = 1;
for ( i = 1 ; i < N ; i++ )
    if ( a[i] > *pMax )
    {
        pMax = &a[i];
        flag = 1;
    }
    else
        if ( a[i] == *pMax )
            flag = 0;
```

! Поиск максимального или минимального элемента массива требует **ОДНОКРАТНОГО** перебора элементов.

Максимальный элемент

А есть ли вообще максимум?

```
pMax = a; count = 1;
for ( i = 1 ; i < N ; i++ )
    if ( a[i] > *pMax )
    {
        pMax = &a[i];
        count = 1;
    }
    else
        if ( a[i] == *pMax )
            count++;
```

Ориентируемся на значение count:

- `count = 1` → единственный максимум, его адрес в `pMax`
- `1 < count < N` → максимальное значение повторяется
- `count = N` → все элементы массива равны между собой, максимума нет

Максимальный среди каких-то

Максимум среди отрицательных:

1) первый отрицательный

```
for ( i = 0; i < N && a[i] >= 0; i++ );
```

Или найдем максимум, или уйдем за пределы массива.
Проверка обязательна

2) максимальный отрицательный

```
if ( i < N )
{
    max = a[i];
    for ( i++; i < N; i++ )
        if ( a[i] < 0 && a[i] > max )
            max = a[i];
}
```

Обмен элементов массива

Поменять местами 2 элемента

```
temp = a[i];  
a[i] = a[j];  
a[j] = temp;
```

Реверс массива

```
for ( i = 0, j = N - 1; i < j ; i++, j-- )  
{  
    temp = a[i];  
    a[i] = a[j];  
    a[j] = temp;  
}
```

Формирование нового массива

Задача: Сформировать новый массив из положительных элементов заданного

```
for ( i = 0, k = 0; i < N ; i++)  
    if ( a[i] > 0 )  
        b[k++] = a[i];
```

Выделять память надо под **N** элементов, но реально в новом массиве будет **k** элементов.

Если память под массив **b** предполагается выделять динамически, то сначала выделяем **N** элементов, потом при необходимости размер массива можно уменьшить.

Удаление элементов из массива

Удалить ОДИН k -й элемент

```
for ( i = k - 1; i < N - 1 ; i++)  
    a[i] = a[i+1];
```

!

При вставке элементов
цикл сдвига начинаем с
конца массива!

Множественное удаление

Вместо того, чтобы удалять лишнее, оставляем нужное.

Задача: Удалить из массива все отрицательные числа.

```
for ( i = 0, k = 0; i < N ; i++ )  
    if ( a[i] >= 0 )  
        a[k++] = a[i];
```


Сортировка массива

Задача: Упорядочить элементы массива по возрастанию

Метод «пузырька»

```
for ( i = N - 1; i > 0 ; i-- )  
    for ( j = 0; j < i; j++ )  
        if ( a[j] > a[j+1] )  
        {  
            tmp = a[j];  
            a[j] = a[j+1];  
            a[j+1] = tmp;  
        }
```



За один проход по массиву на место гарантированно встает **один** элемент.

Сортировка массива

Задача: Упорядочить элементы массива по возрастанию

Метод выбора

```
for ( i = N - 1; i > 0 ; i-- )
{
    imax = 0;
    for ( j = 1; j <= i; j++ )
        if ( a[j] > a[imax] )
            imax = j;
    tmp = a[i];
    a[i] = a[imax];
    a[imax] = tmp;
}
```



За один проход по массиву на место гарантированно встает **один** элемент.

Пример

Задача: Удалить из целочисленного массива максимальный элемент, если он есть и он единственный.

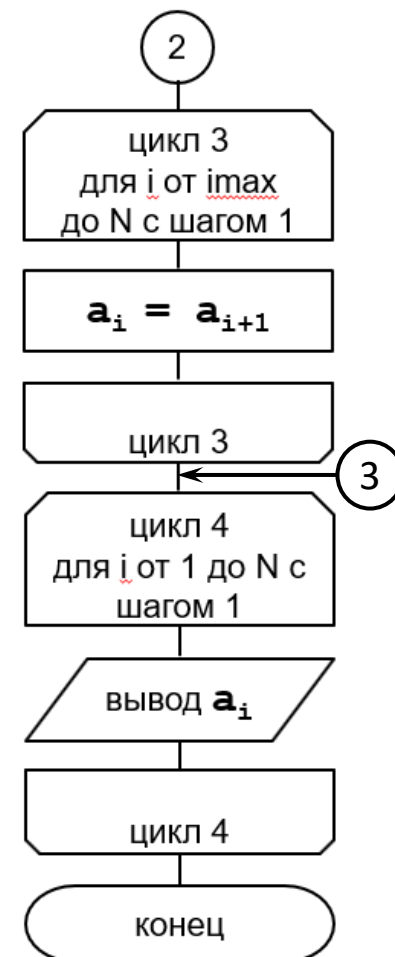
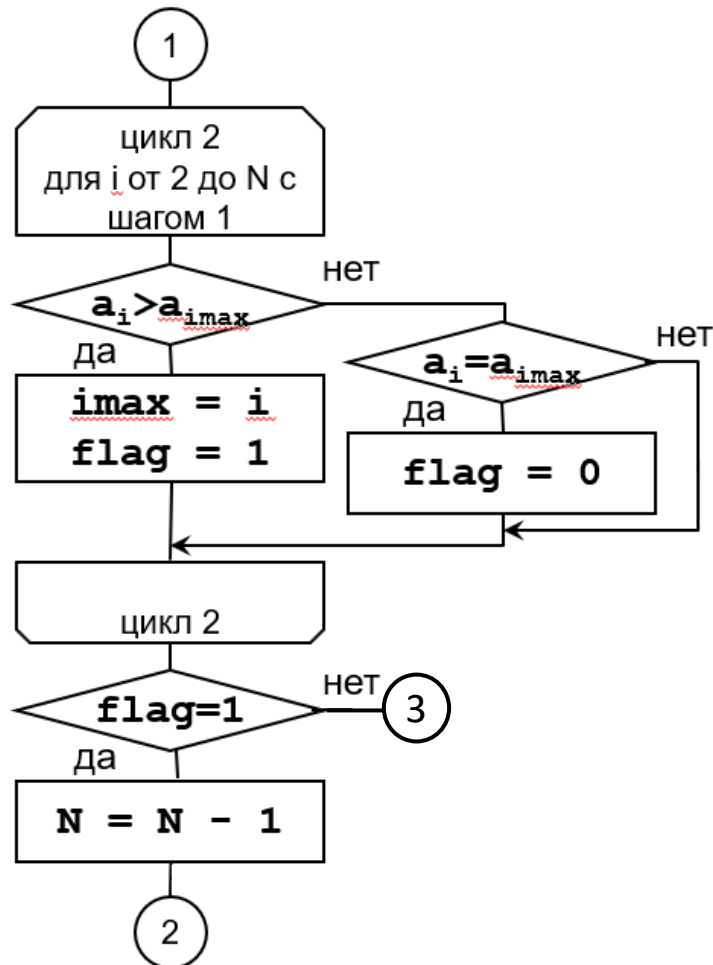
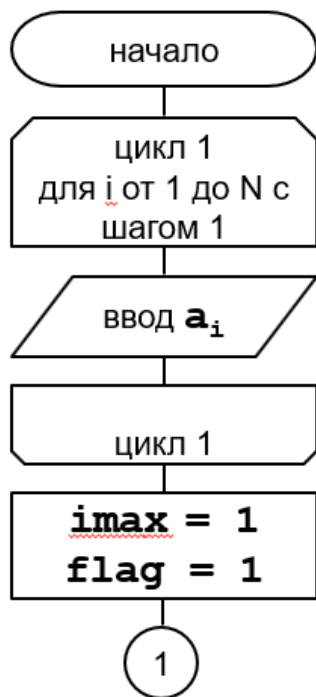
вх и вых: N – размер массива, a – массив

всп: i – индекс элемента

$imax$ – индекс максимального

$flag$ – признак

единственного максимума



Указатели для обработки массива

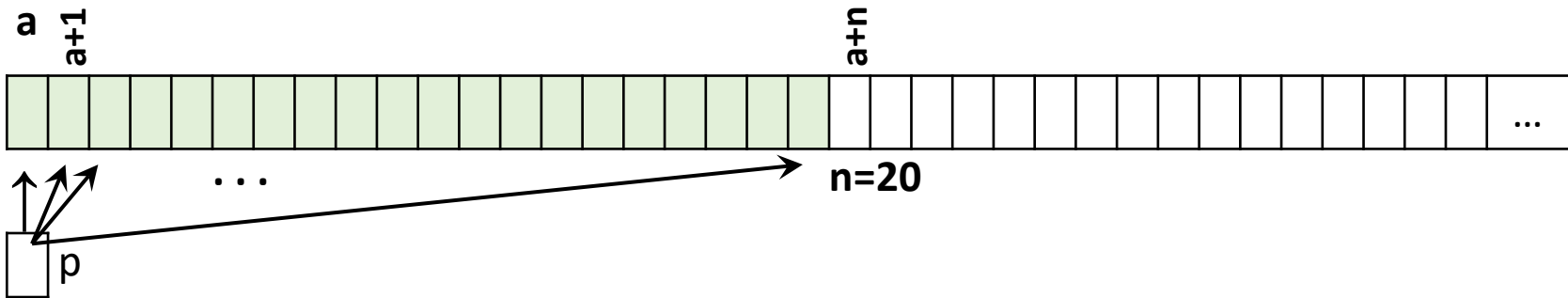
Ввод массива:

```
#include <stdio.h>
#define N 50
int main()
{
```

максимальный
размер массива

```
    int a[N], n, *p, *pmax, flag;
    printf ("Введите размер массива\n");
    scanf ("%d", &n);
    if ( n > N ) n = N;
    printf ("Задайте массив из %d элементов\n", n);
    for ( p = a; p < a + n; )
        scanf ("%d", p++);
```

реальный размер



Указатели для обработки массива

Обработка массива:

```
pmax = a; flag = 1;
for ( p = a + 1; p < a + n; p++ )
    if ( *p > *pmax )
    {
        pmax = p;
        flag = 1;
    }
    else
        if ( *pmax == *p )
            flag = 0;
n -= flag;
if (flag)
    for ( p = pmax; p < a + n; p++ )
        *p = *(p+1);
```

Указатели для обработки массива

Вывод массива:

```
printf ("Полученный массив:\n");  
for ( p = a; p < a + n; )  
    printf ("%10d", *p++);  
return 0;  
}
```