

Одномерные массивы

Циклические сдвиги, сжатие

Алгоритм циклического сдвига на k позиций

I способ

1. определить сколько раз необходимо произвести одноэлементный сдвиг

$$k := k \bmod n;$$

2. k раз применить одноэлементный сдвиг
Алгоритм одноэлементного сдвига.

- 1) Запомнить в дополнительной ячейке первый (или последний) элемент массива
- 2) Сдвинуть все элементы влево (вправо)
- 3) На последнее (первое) место записать тот, который запоминали.



СДВИГ ВПРАВО И ВЛЕВО

```
Program test;
Uses crt;
Const n=10;
Var a:array[1..n] of integer;
    i,j,t,k:integer;
Begin
    clrscr;
    {ВВОД МАССИВА}
    K:=k mod n;
    For j:=1 to k do
    Begin
        t:=a[n];
        for i:=n downto 2 do
            a[i]:=a[i-1];
    A[1]:=t;
    End;
    {ВЫВОД МАССИВА}
End.
```

```
Program test;
Uses crt;
Const n=10;
Var a:array[1..n] of integer;
    i,j,t,k:integer;
Begin
    clrscr;
    {ВВОД МАССИВА}
    K:=k mod n;
    For j:=1 to k do
    Begin
        t:=a[1];
        for i:=1 to n-1 do
            a[i]:=a[i+1];
    A[n]:=t;
    End;
    {ВЫВОД МАССИВА}
End.
```



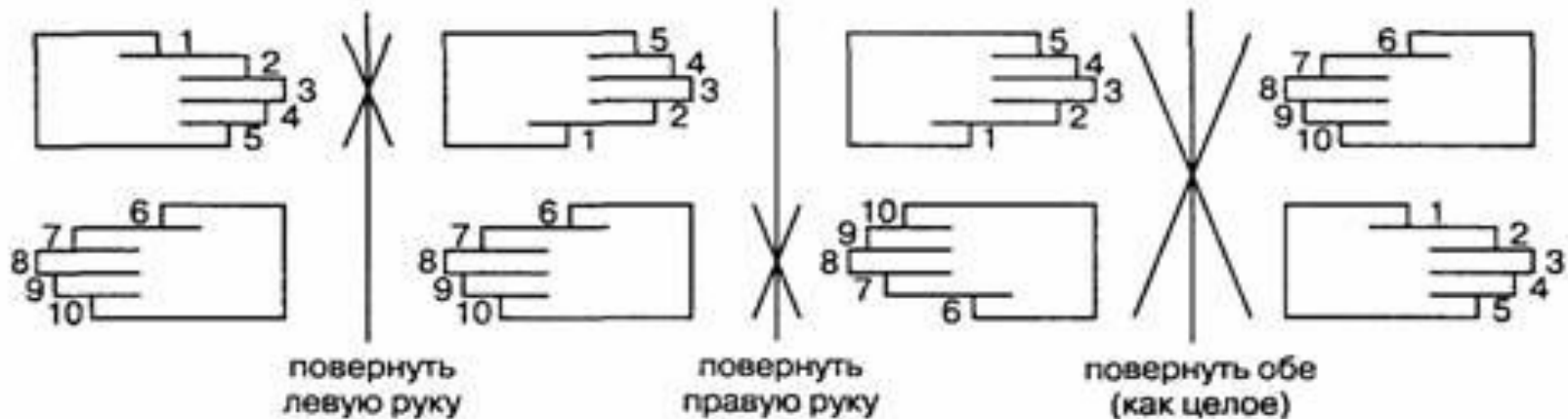
II способ

1. Скопировать первые k элементов массива во временный массив
2. Сдвинуть оставшиеся $n-k$ элементов влево на k позиций
3. Скопировать данные из временного массива обратно в основной массив на последние k позиций



III способ

1. отобразить элементы массива(1, k)
2. отобразить элементы массива (k+1, n)
3. отобразить элементы массива (l, n)



j-сколько раз произвести обмен, **left** - левая граница отображения, **right** - правая граница отображения,

Dlina - длина отображаемой части массива

```
j:=1; left:=1; right:=k; dlina:=right-left+1;
```

```
(***) while j<=dlina div 2 do begin
```

```
    temp:=a[left];
```

```
    a[left]:=a[right];
```

```
    a[right]:=temp;
```

```
    inc(left);dec(right);inc(j);
```

```
end;
```

```
j:=l; left:=k+l; right:=n; dlina:=right-left+l;
```

```
(***) {повторить цикл}
```

```
j:=l; left:=l; right:=n; dlina:=right-left+l;
```

```
(***) {повторить цикл}
```



Сжатие массива.

Удаление каждого k -го элемента:

i – индекс активного элемента

l - индекс просматриваемого элемента

kol – количество элементов после всех удалений.

```
i:=k; l:=k;
```

```
while l<=n do begin
```

```
    if l mod k=0 then inc(l);
```

```
    if l<=n then a[i]:=a[l];
```

```
    inc(i); inc(l);
```

```
end;
```

```
kol:=n-n div k;
```

