

Метод бульбурька

Сортировка массива

Алгоритм

1. **for $i := 1$ to $m - 1$ do**
2. **for $j := 1$ to $m - i$ do**
3. **if $A[j] > A[j + 1]$ then begin**
4. **$k := A[j];$**
5. **$A[j] := A[j + 1];$**
6. **$A[j + 1] := k;$**
7. **end;**

1 9 2 4 8 3 5 4 6 10

k

j := 1

i=1
j := 1 to 9

```
if A[1] > A[ 2 ] then  
begin  
  k := A[ 1 ];  
  A[ 1 ] := A[ 2 ];  
  A[ 2 ] := k;
```

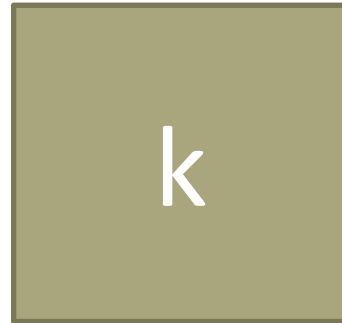
1 9 2 4 8 3 5 4 6 10

k

j := 2

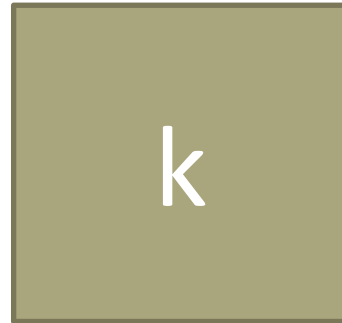
i=1
j := 1 to 9

```
if A[ 2 ] > A[ 3 ] then  
begin  
  k := A[ 2 ];  
  A[ 2 ] := A[ 3 ];  
  A[ 3 ] := k;
```



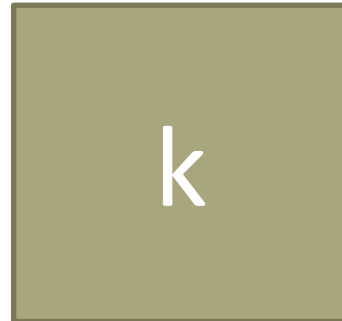
i=1
j :=1 to 9

```
if A[ 3 ] > A[ 4 ] then  
begin  
    k := A[ 3 ];  
    A[ 3 ] := A[ 4 ];  
    A[ 4 ] := k;
```



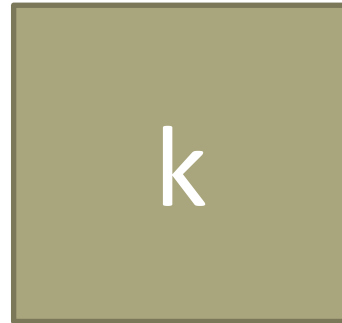
i=1
j :=1 to 9

```
if A[ 4 ] > A[ 5 ] then  
begin  
    k := A[ 4 ];  
    A[ 4 ] := A[ 5 ];  
    A[ 5 ] := k;
```



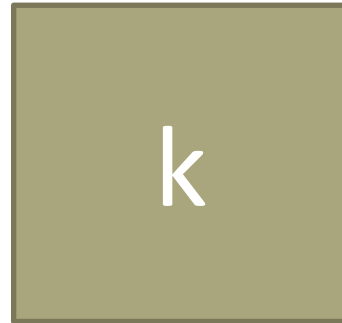
i=1
j :=1 to 9

```
if A[ 5 ] > A[ 6 ] then  
begin  
  k := A[ 5 ];  
  A[ 5 ] := A[ 6 ];  
  A[ 6 ] := k;
```



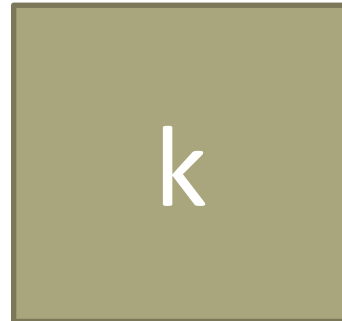
i=1
j :=1 to 9

```
if A[ 6 ] > A[ 7 ] then  
begin  
    k := A[ 6 ];  
    A[ 6 ] := A[ 7 ];  
    A[ 7 ] := k;
```

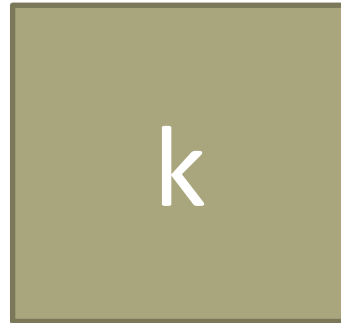
i=1
j :=1 to 9

```
if A[ 7 ] > A[ 8 ] then  
begin  
    k := A[ 7 ];  
    A[ 7 ] := A[ 8 ];  
    A[ 8 ] := k;
```



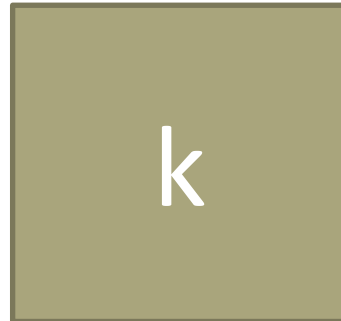
i=1
j :=1 to 9

```
if A[ 8 ] > A[ 9 ] then  
begin  
    k := A[ 8 ];  
    A[ 8 ] := A[ 9 ];  
    A[ 9 ] := k;
```



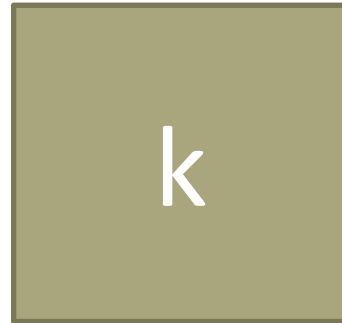
i=1
j := 1 to 9

```
if A[ 9 ] > A[ 10 ] then  
begin  
    k := A[ 9 ];  
    A[ 9 ] := A[ 10 ];  
    A[ 10 ] := k;
```



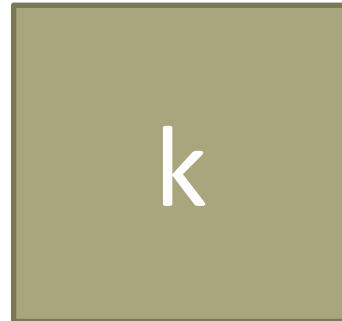
i=2
j := 1 to 8

```
if A[ 1 ] > A[ 2 ] then  
begin  
    k := A[ 1 ];  
    A[ 1 ] := A[ 2 ];  
    A[ 2 ] := k;
```



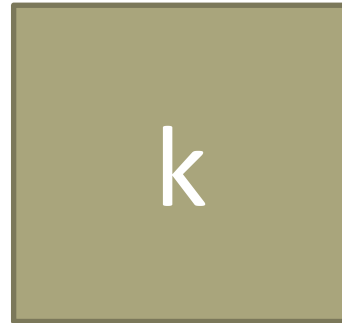
i=2
j := 1 to 8

```
if A[ 2 ] > A[ 3 ] then  
begin  
    k := A[ 2 ];  
    A[ 2 ] := A[ 3 ];  
    A[ 3 ] := k;
```



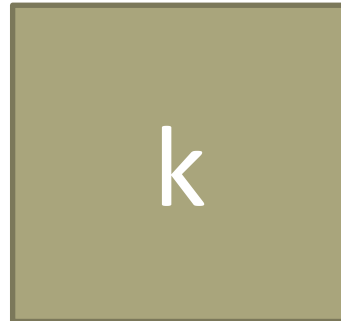
i=2
j :=1 to 8

```
if A[ 3 ] > A[ 4 ] then  
begin  
    k := A[ 3 ];  
    A[ 3 ] := A[ 4 ];  
    A[ 4 ] := k;
```



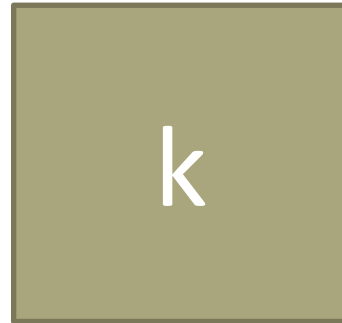
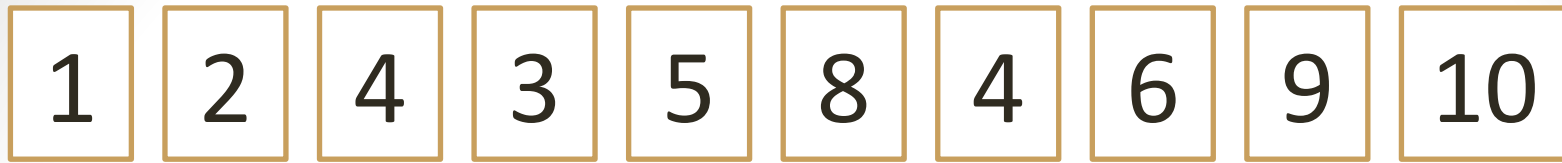
i=2
j :=1 to 8

```
if A[ 4 ] > A[ 5 ] then  
begin  
    k := A[ 4 ];  
    A[ 4 ] := A[ 5 ];  
    A[ 5 ] := k;
```



i=2
j :=1 to 8

```
if A[ 5 ] > A[ 6 ] then  
begin  
  k := A[ 5 ];  
  A[ 5 ] := A[ 6 ];  
  A[ 6 ] := k;
```

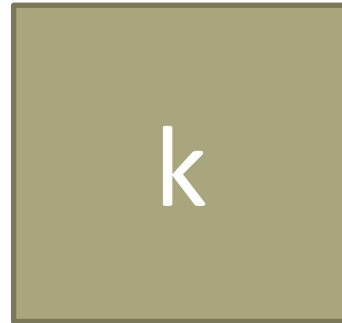
i=2
j :=1 to 8

```
if A[ 6 ] > A[ 7 ] then  
begin  
    k := A[ 6 ];  
    A[ 6 ] := A[ 7 ];  
    A[ 7 ] := k;
```



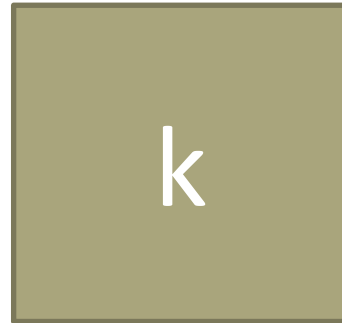
i=2
j :=1 to 8

```
if A[ 7 ] > A[ 8 ] then  
begin  
    k := A[ 7 ];  
    A[ 7 ] := A[ 8 ];  
    A[ 8 ] := k;
```



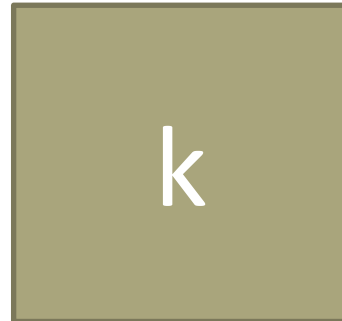
i=2
j := 1 to 8

```
if A[ 8 ] > A[ 9 ] then  
begin  
    k := A[ 8 ];  
    A[ 8 ] := A[ 9 ];  
    A[ 9 ] := k;
```



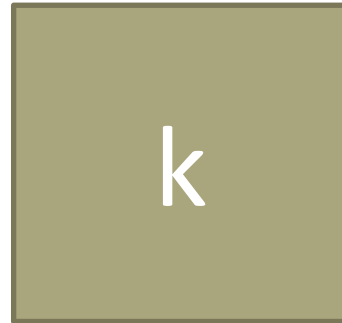
i=3
j := 1 to 7

```
if A[ 1 ] > A[ 2 ] then  
begin  
  k := A[ 1 ];  
  A[ 1 ] := A[ 2 ];  
  A[ 2 ] := k;
```



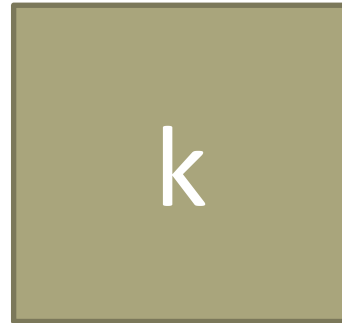
i=3
j := 1 to 7

```
if A[ 2 ] > A[ 3 ] then  
begin  
    k := A[ 2 ];  
    A[ 2 ] := A[ 3 ];  
    A[ 3 ] := k;
```



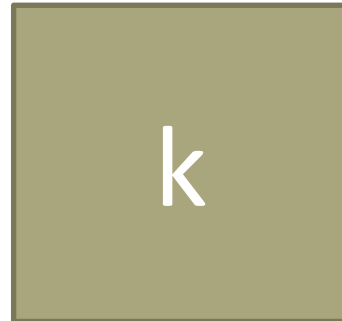
i=3
j :=1 to 7

```
if A[ 3 ] > A[ 4 ] then  
begin  
    k := A[ 3 ];  
    A[ 3 ] := A[ 4 ];  
    A[ 4 ] := k;
```



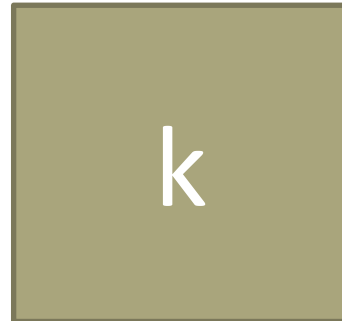
i=3
j := 1 to 7

```
if A[ 4 ] > A[ 5 ] then  
begin  
    k := A[ 4 ];  
    A[ 4 ] := A[ 5 ];  
    A[ 5 ] := k;
```



i=3
j := 1 to 7

```
if A[ 5 ] > A[ 6 ] then  
begin  
    k := A[ 5 ];  
    A[ 5 ] := A[ 6 ];  
    A[ 6 ] := k;
```

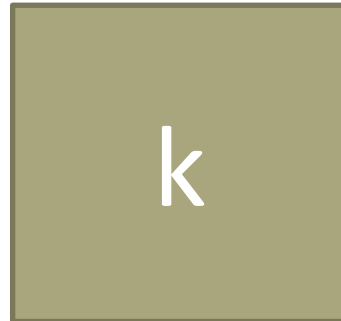
i=3
j := 1 to 7

```
if A[ 6 ] > A[ 7 ] then  
begin  
    k := A[ 6 ];  
    A[ 6 ] := A[ 7 ];  
    A[ 7 ] := k;
```



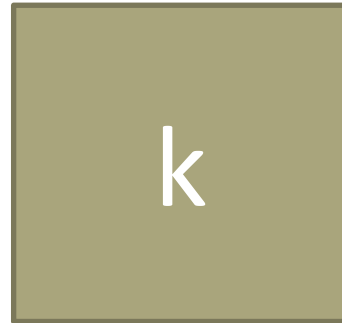
i=3
j := 1 to 7

```
if A[ 7 ] > A[ 8 ] then  
begin  
    k := A[ 7 ];  
    A[ 7 ] := A[ 8 ];  
    A[ 8 ] := k;
```



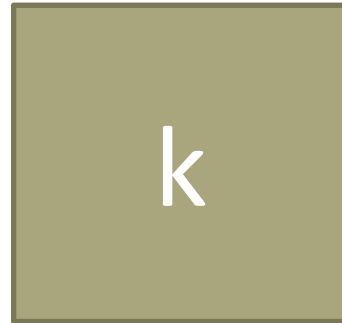
i=4
j := 1 to 6

```
if A[ 1 ] > A[ 2 ] then  
begin  
    k := A[ 1 ];  
    A[ 1 ] := A[ 2 ];  
    A[ 2 ] := k;
```



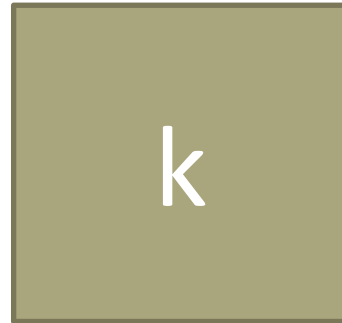
i=4
j := 1 to 6

```
if A[ 2 ] > A[ 3 ] then  
begin  
    k := A[ 2 ];  
    A[ 2 ] := A[ 3 ];  
    A[ 3 ] := k;
```



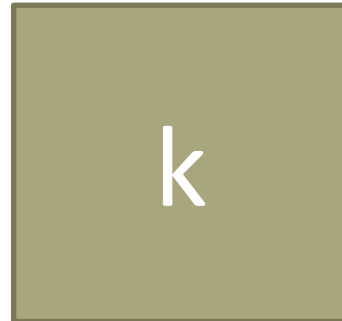
i=4
j := 1 to 6

```
if A[ 3 ] > A[ 4 ] then  
begin  
    k := A[ 3 ];  
    A[ 3 ] := A[ 4 ];  
    A[ 4 ] := k;
```



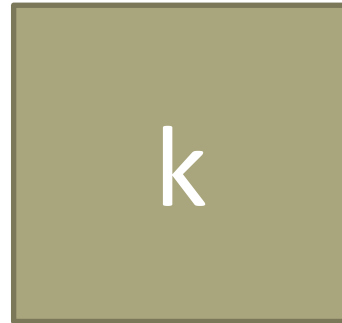
i=4
j := 1 to 6

```
if A[ 4 ] > A[ 5 ] then  
begin  
    k := A[ 4 ];  
    A[ 4 ] := A[ 5 ];  
    A[ 5 ] := k;
```



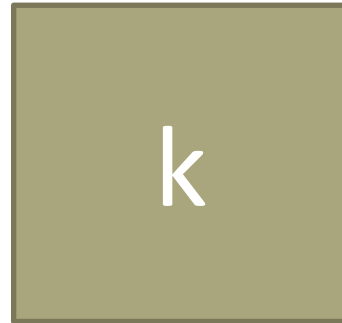
i=4
j := 1 to 6

```
if A[ 5 ] > A[ 6 ] then  
begin  
  k := A[ 5 ];  
  A[ 5 ] := A[ 6 ];  
  A[ 6 ] := k;
```



i=4
j := 1 to 6

```
if A[ 6 ] > A[ 7 ] then  
begin  
  k := A[ 6 ];  
  A[ 6 ] := A[ 7 ];  
  A[ 7 ] := k;
```

i=5
j := 1 to 5

```
if A[ 1 ] > A[ 2 ] then  
begin  
    k := A[ 1 ];  
    A[ 1 ] := A[ 2 ];  
    A[ 2 ] := k;
```



i=5
j := 1 to 5

```
if A[ 2 ] > A[ 3 ] then  
begin  
  k := A[ 2 ];  
  A[ 2 ] := A[ 3 ];  
  A[ 3 ] := k;
```



Программа будет выполняться до конца
цикла для j и для i .

$i=5$

$j := 1$ to 5

$i=6$

$j := 1$ to 4

$i=7$

$j := 1$ to 3

$i=8$

$j := 1$ to 2

$i=9$

$j := 1$ to 1

$i=10$