

Найти k-ый по величине (сверху) элемент массива

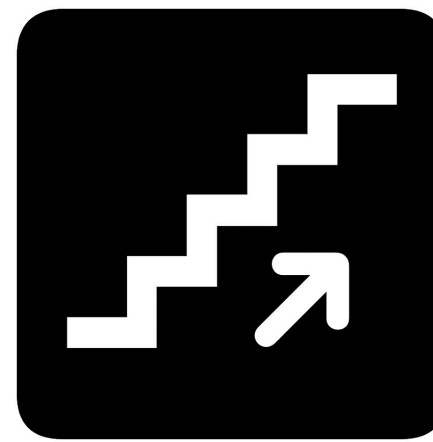
5 4 3 2 1

1 проход n операций
2 прохода 2n операций
 $n+(n-1)+(n-2)+\dots+2+1=n(n+1)/2=O(n^2)$

ты можешь пользоваться вспомогательным массивом из k+1 чисел

```
int k;  
scanf("%d",&k);  
int *help=(int *)malloc(k*sizeof(int));  
for(int i=0;i<k;i++)  
{  
    help[i]=0;  
}
```

5 1 23 2 525 6 31 43 45 7
0 0 0 0
5 0 0 0
5 1 0 0->1 5 0 0 0
1 5 23 0 0
1 5 23 2->1 5 2 23->1 2 5 23 0
1 2 5 23 525 -> 2 5 23 525 0
2 5 23 525 6 -> 2 5 23 6 525-> 2 5 6 23 525 ->5 6 23 525 0
5 6 23 525 31 ->5 6 23 31 525->6 23 31 525 0
6 23 31 525 43->6 23 31 43 525->23 31 43 525 0
23 31 43 525 45->23 31 43 45 525->31 43 45 525 0
31 43 45 525 7->31 43 45 7 525->31 43 7 45 525->31 7 43 45 525-> 7 31 43 45 525->31 7 43 45 525 0



4-ый по величине

рассмотрим n шаров. нет n много - рассмотрим k шаров

```
void uprising(int ms[], int razmer)  
{  
    int k;  
    scanf("%d",&k);  
    int *help=(int *)malloc((k+1)*sizeof(int));  
    for(int i=0;i<k+1;i++)  
    {  
        help[i]=0;  
    }  
    int s, d;  
    int t=0;  
    while(t<k)  
    {  
        help[t]=ms[t];  
        s=t;  
        while(s-1>=0 && help[s]<help[s-1])  
        {  
            d=help[s];  
            help[s]=help[s-1];  
            help[s-1]=d;  
            s--;  
        }  
        t++;  
    }  
    int j=k;  
    while(j<razmer)  
    {  
        help[k]=ms[j];  
        s=k;  
        while(s-1>=0 && help[s]<help[s-1])  
        {  
            d=help[s];  
            help[s]=help[s-1];  
            help[s-1]=d;  
            s--;  
        }  
        s=0;  
        while(s+1<=k)  
        {  
            help[s]=help[s+1];  
            s++;  
        }  
        j++;  
    }  
    printf("%d\n", help[0]);  
}
```