

```
1 // https://toph.co/p/subarray-sum
2 #include<bits/stdc++.h>
3 using namespace std;
4 int n,k,a[5001], dp[5001][5001];
5 int fun(int pos,int cnt){
6     if(pos>n) return 0;
7     if(dp[pos][cnt]==-1) return dp[pos][cnt];
8     int ret = -1000000000;
9     if(a[pos]<0 && cnt<k) ret = max(ret, fun(pos+1,cnt+1));
10    ret = max(ret, a[pos]+fun(pos+1,cnt));
11    return dp[pos][cnt] = max(ret,a[pos]);
12 }
13 int main(){
14     int tt; scanf("%d",&tt);
15     for(int ks=1; ks<=tt; ks++){
16         scanf("%d%d",&n,&k);
17         for(int i=1; i<=n; i++) scanf("%d",&a[i]);
18         memset(dp,-1,sizeof(dp));
19         int ans = -1000000000;
20         for(int i=1; i<=n; i++){
21             ans = max(ans, a[i]);
22             ans = max(ans, fun(i,0));
23         }
24         printf("Case %d: %d\n",ks,ans);
25     }
26     return 0;
27 }
28 -----
29 #include<bits/stdc++.h>
30 using namespace std;
31 int a[5001], dp[5001][5001];
32 int fun(int pos,int cnt){
33     if(pos==0) return 0;
34     int &ret = dp[pos][cnt];
35     if(ret!=-2000000000) return ret;
36     ret = -1000000000;
37     if(a[pos]>=0) {
38         ret = max(ret, a[pos]+fun(pos-1,cnt));
39     }else{
40         if(cnt>0) ret = max(ret, fun(pos-1,cnt-1));
41         ret = max(ret, a[pos]+fun(pos-1,cnt));
42     }
43     return ret = max(ret, a[pos]);
44 }
45 int main(){
46     int tt; scanf("%d",&tt);
47     for(int ks=1; ks<=tt; ks++){
48         int n,k; scanf("%d%d",&n,&k);
49         for(int i=1; i<=n; i++) scanf("%d",&a[i]);
50
51         for(int i=0; i<5001; i++){
52             for(int j=0; j<5001; j++){
53                 dp[i][j] = -2000000000;
54             }
55         }
56
57         int ans = -1000000000;
58         for(int i=1; i<=n; i++){
59             ans = max(ans, fun(i,k));
60         }
61         printf("Case %d: %d\n",ks,ans);
62     }
63     return 0;
64 }
```