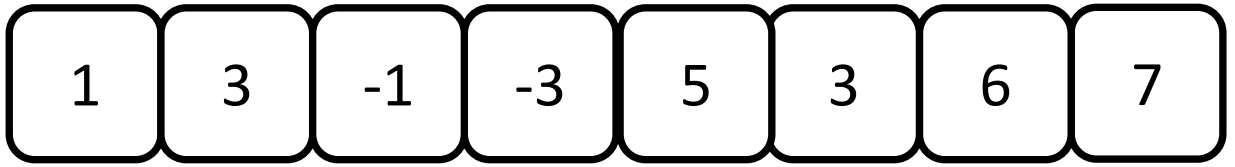

FACE
Prep

Maximum Sliding Window

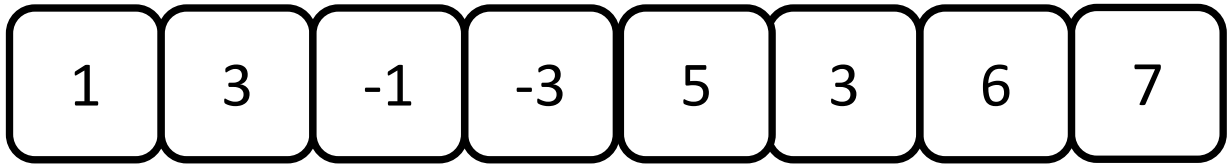
Maximum Sliding Window

Problem:

You are given an array of integers, there is a sliding window of size k which is moving from the very left of the array to the very right. You can only see the k numbers in the window. Each time the sliding window moves right by one position. Return the max sliding window.

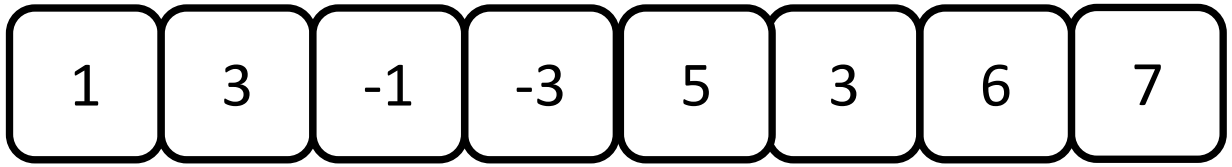


Key idea



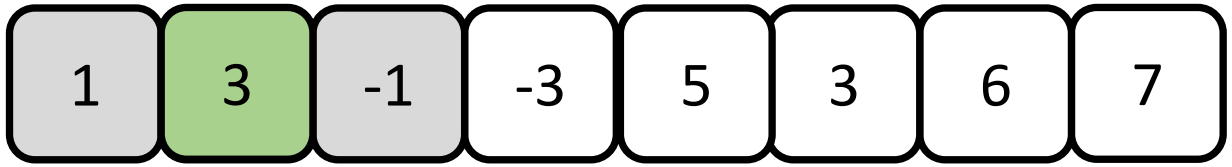
Key idea

K = 3



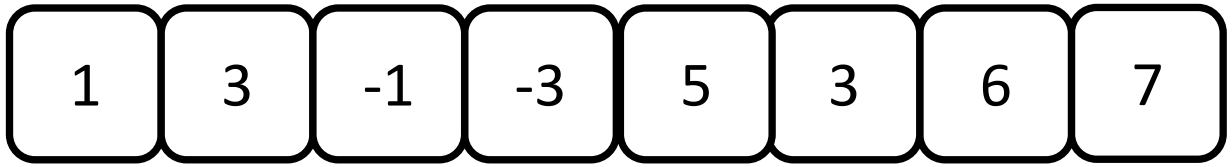
Key idea

K = 3



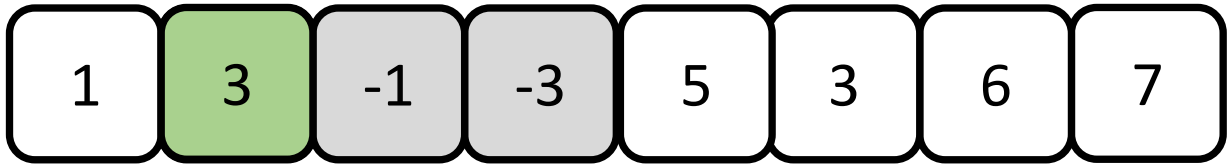
Key idea

K = 3



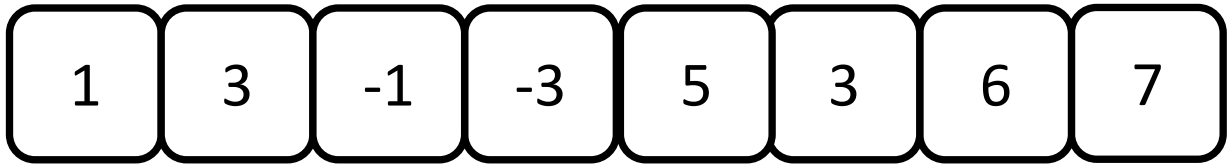
Key idea

K = 3



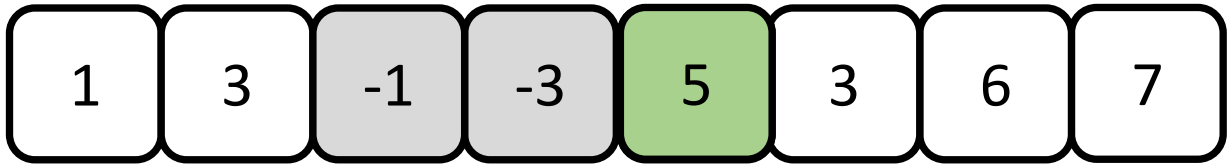
Key idea

K = 3



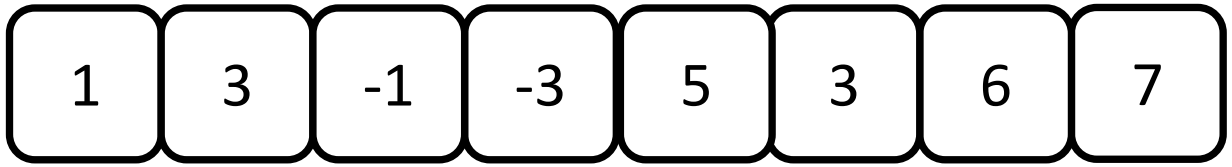
Key idea

K = 3



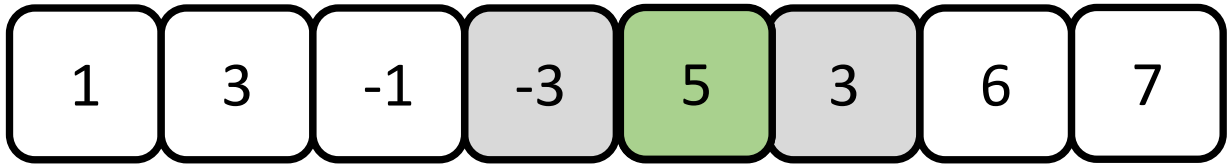
Key idea

K = 3



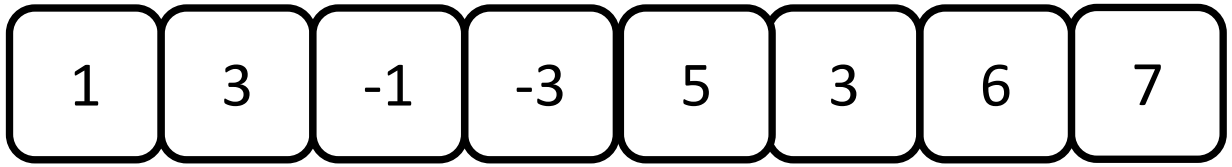
Key idea

K = 3



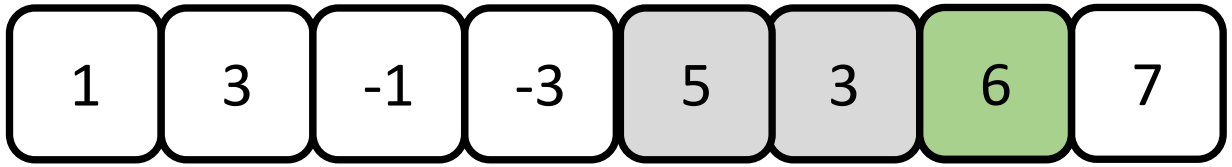
Key idea

K = 3



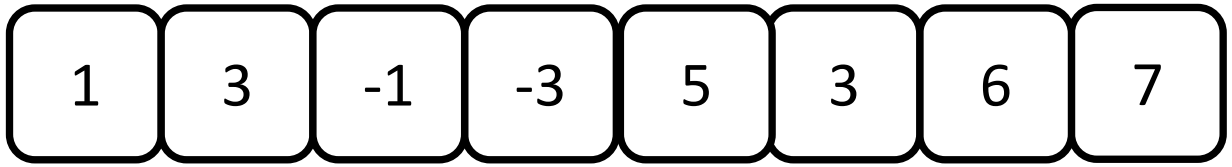
Key idea

K = 3



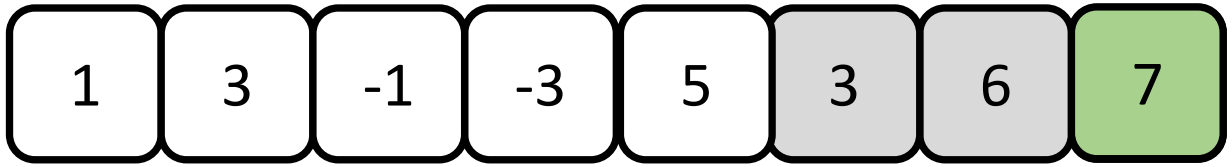
Key idea

K = 3



Key idea

K = 3



```java
import java.util.*;

public class Main {
    public static int[] maxsliding( int[] in,  int w) {
     int[] max_left = new int[in.length];
     int[] max_right = new int[in.length];
    max_left[0] = in[0];
    max_right[in.length - 1] = in[in.length - 1];
    for (int i = 1; i < in.length; i++) {
        max_left[i] = (i % w == 0) ? in[i] : Math.max(max_left[i -
 1], in[i]);

        final int j = in.length - i - 1;
        max_right[j] = (j % w == 0) ? in[j] : Math.max(max_right[j + 1], in
[j]);
    }
    final int[] sliding_max = new int[in.length - w + 1];
    for (int i = 0, j = 0; i + w <= in.length; i++) {
        sliding_max[j++] = Math.max(max_right[i], max_left[i + w - 1]);
    }
    return sliding_max;
}
```

```
23 public static void main(String[] args) throws Exception {
24         Scanner sc=new Scanner(System.in);
25         int n=sc.nextInt();
26         int k=sc.nextInt();
27     int a[]=new int[n];
28     for (int i = 0; i < n; i++)
29     {
30         a[i]=sc.nextInt();
31     }
32     int ans[]=maxsliding(a,k);
33     for(int i=0;i<ans.length;i++)
34     System.out.print(ans[i]+" ");
35     }
36
37 }
38
39
40
41
42
43
44
```

THANK YOU

FACE Prep