

SLIDING WINDOW

```
class Solution {  
    public int[] maxSlidingWindow(int[] nums, int k) {  
  
        if (nums.length == 0) return new int[0];  
  
        Deque<Integer> dq = new LinkedList<>();  
        int n = nums.length;  
        int[] result = new int[n - k + 1];  
        int idx = 0;  
  
        for (int i = 0; i < n; i++) {  
            if (!dq.isEmpty() && dq.peekFirst() <= i - k) {  
                dq.pollFirst();  
            }  
            while (!dq.isEmpty() && nums[dq.peekLast()] < nums[i]) {  
                dq.pollLast();  
            }  
  
            dq.offerLast(i);  
  
            // Window ready  
            if (i >= k - 1) {  
                result[idx++] = nums[dq.peekFirst()];  
            }  
        }  
  
        return result;  
    }  
}
```

}

}