Largest Sum Subarray of Size at least K

Given an array **a** of length **n** and a number **k**, find the **largest sum** of the subarray containing **at least k** numbers. It is guaranteed that the size of array is **at-least k**.

Example 1:

```
Input :
n = 4
a[] = {1, -2, 2, -3}
k = 2
Output :
1
Explanation :
The sub-array of length at-least 2
that produces greatest sum is {1, -2, 2}
```

Example 2:

```
Input :
n = 6
a[] = {1, 1, 1, 1, 1, 1}
k = 2
Output :
6
Explanation :
The sub-array of length at-least 2
that produces greatest sum is {1, 1, 1, 1, 1, 1}
```

Your Task:

You don't need to read input or print anything. Your task is to complete the function **maxSumWithK()** which takes the array **a[]**, its size **n** and an integer **k** as inputs and returns the value of the **largest sum** of the subarray containing **at least k** numbers.

Expected Time Complexity: O(n)
Expected Auxiliary Space: O(n)

Constraints:
$1 <= n <= 10^5$
$-10^5 <= a[i] <= 10^5$
1 <= k <= n