

**1.Easy\12.Longest\_subarray\_with\_given\_sum.cpp**

```
1  /*
2  QUESTION:-
3  You are given an array 'A' of size 'N' and an integer 'K'. You need to print the length of
  the longest subarray of array 'A' whose sum = 'K'.
4  Example:
5  Input: 'N' = 7 'K' = 3
6  'A' = [1, 2, 3, 1, 1, 1, 1]
7  Output: 3
8  Explanation: Subarrays whose sum = '3' are:
9  [1, 2], [3], [1, 1, 1], [1, 1, 1]
10 Here, the length of the longest subarray is 3, which is our final answer.
11 */
12
13 /*
14 APPROACH:-
15 -> Use sliding window approach using two pointers start and end
16 -> Run a loop to traverse the entire array add from end and subtract from start when sum>k
17 -> If sum==k then, update the ans now, window size = end-start+1
18 */
19
20 // CODE:-
21 int longestSubarrayWithSumK(vector<int> a, long long k)
22 {
23     int start = 0;
24     int ans = 0;
25     long long sum = 0;
26     int n = a.size();
27
28     for (int end = 0; end < n; end++)
29     {
30         sum += a[end];
31         while (sum > k)
32         {
33             sum -= a[start];
34             start++;
35         }
36         if (sum == k)
37         {
38             ans = max(ans, end - start + 1);
39         }
40     }
41     return ans;
42 }
43
44 // TIME COMPLEXITY = O(N)
45 // SPACE COMPLEXITY = O(0)
```