Given an array *nums* and a target value *k*, find the maximum length of a subarray that sums to *k*. If there isn't one, return 0 instead.

**Note:**  
The sum of the entire *nums* array is guaranteed to fit within the 32-bit signed integer range.

**Example 1:**

**Input:** *nums* = [1, -1, 5, -2, 3], *k* = 3

**Output:** 4

**Explanation:** The subarray [1, -1, 5, -2] sums to 3 and is the longest.

**Example 2:**

**Input:** *nums* = [-2, -1, 2, 1], *k* = 1

**Output:** 2

**Explanation:** The subarray [-1, 2] sums to 1 and is the longest.

**Follow Up:**  
Can you do it in O(*n*) time?