Sliding Window

TestLeaf
Always Ahead

Sliding Window

TestLeaf
Always Ahead

- ✓ Used for finding subarrays inside an array or Substring in String
- ✓ Can only apply when you have deal with consecutive elements
- ✓ Subset of dynamic programming
- ✓ One of the common asked interview because of its efficiency:
  - Time Complexity : O(n)
  - Space Complexity: O(1)

Working Behaviour

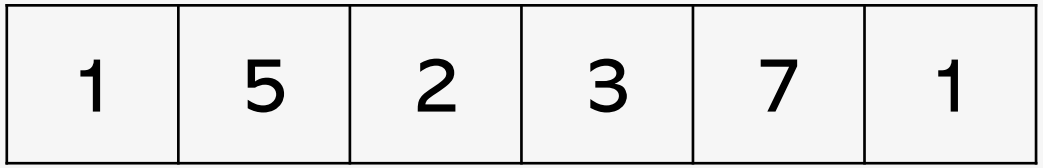
Given an array of integers and a number k.
Return the highest sum of any k consecutive elements in the array.

Input Array :

1	5	2	3	7	1

Target (k) : 3

Sliding Window



- Sliding Window Technique is a method for finding subarrays in an array that satisfy given conditions.
- We do this via maintaining a subset of items as our window and resize and move that window within the larger list until we find a solution.

Problem Statement # 2

Given an array of positive integers, find the subarrays that adds up to the given number (k)

Input Array :

1	7	4	3	1	2	1	5	1

Desired Sum (k) : 7

Understand the Problem

1	7	4	3	1	2	1	5	1

- Contiguous Subarray
- Input size can be any number
- No negative numbers or zero
- Tip: Beware of the memory space*

Brute Force [Desired Sum : 7]

1	7	4	3	1	2	1	5	1

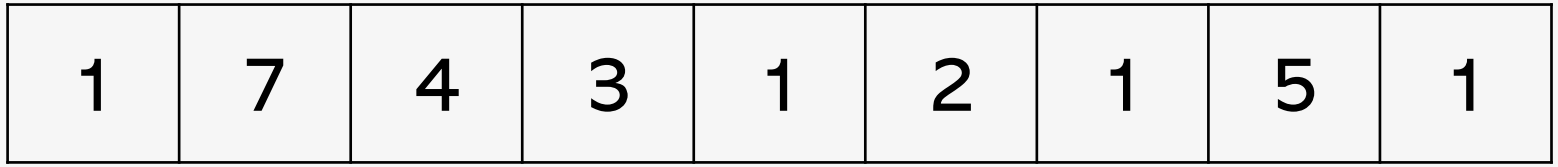
- Iterate through every element and keep adding
- And if the sum = 7 => return the sub array

Sliding Window [Desired Sum : 7]

1	7	4	3	1	2	1	5	1

- Start with first element
- If the sum is less than the desired sum, slide to the next element [Grow]
- Again, sum that and check if it is lesser, equal or greater
- If it is lesser, add the next element to slide [Grow]
- If it is greater, than shrink the last element on the left [Shrink]
- If it is equal, you got it and again do both Grow and Shrink [Slide]

Using Hash



- Start with first element
- If the sum is less than the desired sum, slide to the next element [Grow]
- Again, sum that and check if it is lesser, equal or greater
- If it is lesser, add the next element to slide [Grow]
- If it is greater, than shrink the last element on the left [Shrink]
- If it is equal, you got it and again do both Grow and Shrink [Slide]