75 Days of Code

Day13 Problem no: 1679 Max Number of K-Sum Pairs (leetcode)

You are given an integer array nums and an integer k.

In one operation, you can pick two numbers from the array whose sum equals k and remove them from the array.

Return the maximum number of operations you can perform on the array.

Example 1:

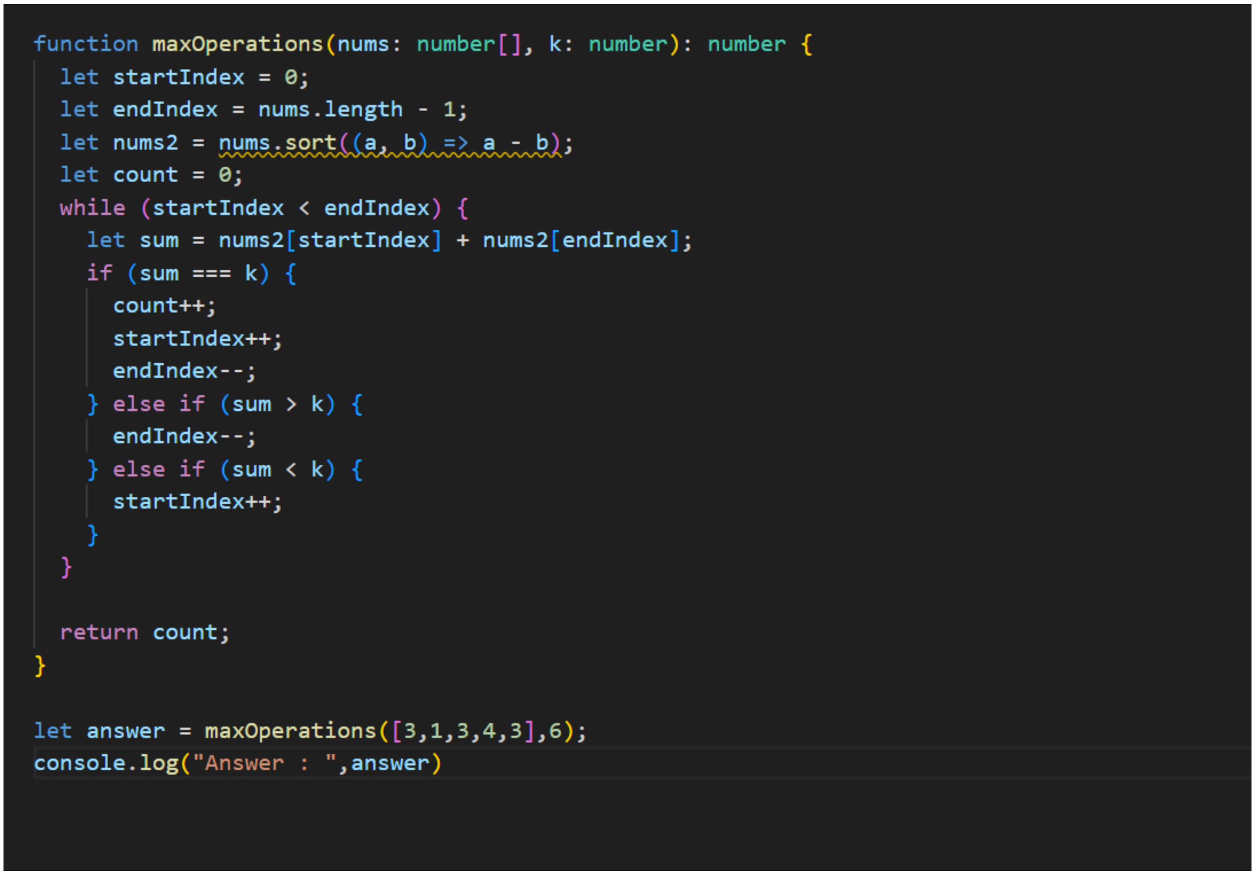
Input: nums = [1,2,3,4], k = 5
Output: 2
Explanation: Starting with nums = [1,2,3,4]:
- Remove numbers 1 and 4, then nums = [2,3]
- Remove numbers 2 and 3, then nums = []
There are no more pairs that sum up to 5, hence a total of 2 operations.
Example 2:

Input: nums = [3,1,3,4,3], k = 6
Output: 1
Explanation: Starting with nums = [3,1,3,4,3]:
- Remove the first two 3's, then nums = [1,4,3]
There are no more pairs that sum up to 6, hence a total of 1 operation.

For this problem ,

1. Take two pointers as startIndex , endIndex and loop the array while startIndex< endIndex
2. Sort the num array

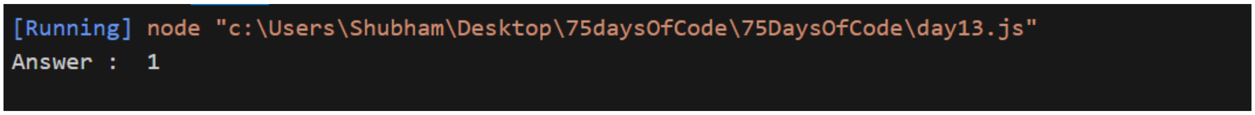
3. Compare the sum of two pointer , if they equal to given k then increase startIndex and decrease endIndex
4. If sum is greater than k ,reduce the endIndex else increase startIndex



```
function maxOperations(nums: number[], k: number): number {
  let startIndex = 0;
  let endIndex = nums.length - 1;
  let nums2 = nums.sort((a, b) => a - b);
  let count = 0;
  while (startIndex < endIndex) {
    let sum = nums2[startIndex] + nums2[endIndex];
    if (sum === k) {
      count++;
      startIndex++;
      endIndex--;
    } else if (sum > k) {
      endIndex--;
    } else if (sum < k) {
      startIndex++;
    }
  }

  return count;
}

let answer = maxOperations([3,1,3,4,3],6);
console.log("Answer : ",answer)
```



```
[Running] node "c:\Users\Shubham\Desktop\75daysOfCode\75DaysOfCode\day13.js"
Answer :  1
```