https://course.acciojob.com/idle?question=f7a7790c-2f0f-49ef-aadb-ee0031f88e28

MEDIUM

Max Score: 40 Points

Two Sum in sorted Array

Given an array `A` sorted in non-decreasing order, find two numbers such that they add up to a specific `target` number. Let these two numbers be A[index1] and A[index2] where 1 <= index1 < index2 <= A.length.

Return the indices of the two numbers, index1 and index2 in one-based indexing.

Note: The tests are generated such that there is exactly one solution. You may not use the same element twice.

Your solution must use only constant extra space.

Input Format

For each test case: The first line contains an integer `N` denoting the number of elements and a target element `target`.

The second line contains N space separated integers denoting the elements of the array `A`.

Output Format

For each test case return an array, containing the index of the required element.

Example 1

Input

```
4 9
2 7 11 15
```

Output

```
1 2
```

Explanation

As 2 and 7 add up to 9, therefore answer is 1 and 2 i.e position of 2 and 7.

Example 2

Input

```
3 6
2 3 4
```

Output

```
1 3
```

Explanation

As 2 and 4 add up to 6, therefore answer is 1 and 3 i.e position of 2 and 4.

Constraints:

1 <= N <= 10000

1 <= target <= 200000

1 <= A[i] <= 100000

Topic Tags

Arrays

My code

```
// in java
import java.util.*;
class Main{
        public static void main(String[] args){
                try (Scanner sc = new Scanner(System.in)) {
        int n=sc.nextInt();
        int target=sc.nextInt();
        int []A=new int[n];
        for(int i=0;i<n;i++){
            A[i]=sc.nextInt();
        }
        Solution ob =new Solution();
        int []ans = ob.twosum(A,n,target);
        System.out.println(ans[0]+" "+ans[1]);
                }
        }

}

class Solution{

    public int[] twosum(int arr[], int n, int target){
        //Write code here
                 for(int i=0;i<arr.length-1;i++)
      for(int j=i+1;j<arr.length;j++){
          if((arr[i]+arr[j])==target)
          {
              return new int[] {i+1 , j+1 };
          }
```

```
      }
    return null;
    }
}
```