https://course.acciojob.com/idle?question=05d6b1b0-d843-4974-8a19-c65be520d5a9

- EASY
- Max Score: 30 Points
-

Two Sum

You are given a non-decreasing array `A` of `N` integers. You are given another integer `target`.

You have to find two unique indices of the array such that the values at those indices have a sum equal to `target`.

Return the indices as a sorted integer array of size 2.

It is guaranteed that the test cases are made such that only one solution exists.

The array is 1-indexed.

Note Complete the given function. The input and output would be handled by the driver code.

Your solution must use only constant extra space.

Input Format

The first line contains a single integer `N`.

The second line contains `N` space-separated integers of array `A`.

The third line contains a single integer `target`.

Output Format

Print the answer in a new line.

Example 1

Input

```
4
1 2 3 4
6
```

Output

```
2 4
```

Explanation

A[2] = 2. A[4] = 4. 2 + 4 = 6.

Example 2

Input

```
5
-10 1 2 5 7
3
```

Output

```
2 3
```

Explanation

A[2] = 1. A[3] = 2. 1 + 2 = 3.

Constraints

1 <= N <= 10000

-100000 <= A[i] <= 100000

-200000 <= target <= 200000

Topic Tags

- 2-Pointers
- Arrays

My code

```
// in java
import java.util.*;
import java.lang.*;
import java.io.*;

class Main {

    public static int[] twoSum(int[] arr, int t) {
      // Your code here
            int n=arr.length;
            int ar[]=new int[2];
            for(int i=0;i<n-1;i++)
                {
                    for(int j=i+1;j<n;j++)
                        if(arr[i]+arr[j]==t)
                        {
                            //System.out.print(i+" "+j);

                            ar[0]=i+1;
                            ar[1]=j+1;
                            return ar;
                        }
                }
                return ar;
```

```
    }

    public static void main (String[] args) throws IOException {
            BufferedReader br = new BufferedReader(new
InputStreamReader(System.in));
        long n = Long.parseLong(br.readLine().trim());
        String inputLine[] = br.readLine().trim().split(" ");
        int[] arr = new int[(int)n];
        for(int i=0; i<n; i++)arr[i]=Integer.parseInt(inputLine[i]);
        int m = Integer.parseInt(br.readLine().trim());
        int[] ans = (twoSum(arr, m));
        System.out.println(ans[0] + " " + ans[1]);
        }
}
```