# **Name**: Saii Anish R

**Roll No.** : 2116241501175

**Week 07**

**Q1) Problem Statement:**

Sunny and Johnny like to pool their money and go to the ice cream parlor. Johnny never buys the same flavor that Sunny does. The only other rule they have is that they spend all of their money.Given a list of prices for the flavors of ice cream, select the two that will cost all of the money they have.For example, they have m = 6 to spend and there are flavors costing cost = [1, 2, 3, 4, 5, 6]. The two flavors costing 1 and 5 meet the criteria. Using 1-based indexing, they are at indices 1 and 4 Complete the code in the editor below. It should return an array containing the indices of the prices of the two flavors they buy, sorted ascending.

It has the following:

m: an integer denoting the amount of money they have to spend cost: an integer array denoting the cost of each flavor of ice cream **Input Format:**

The first line contains an integer, t, denoting the number of trips to the ice cream parlor.

The next t sets of lines each describe a visit. Each trip is described as follows:

1. The integer m, the amount of money they have pooled.
2. The integer n, the number of flavors offered at the time.
3. n space-separated integers denoting the cost of each flavor: cost[cost[1], cost[2], ..., cost[n]].Note: The index within the cost array represents the flavor of the ice cream purchased. Constraints: 1 ≤ t ≤ 50

2 ≤ m ≤ 104

2 ≤ n ≤ 104

1 ≤ cost[i] ≤ 104, ∀ i ∈ [1, n]

There will always be a unique solution. **Output Format:**

For each test case, print two space-separated integers denoting the indices of the two flavors purchased, in ascending order. **Sample Input:**

2

4

5

1. 4 5 3 2

4

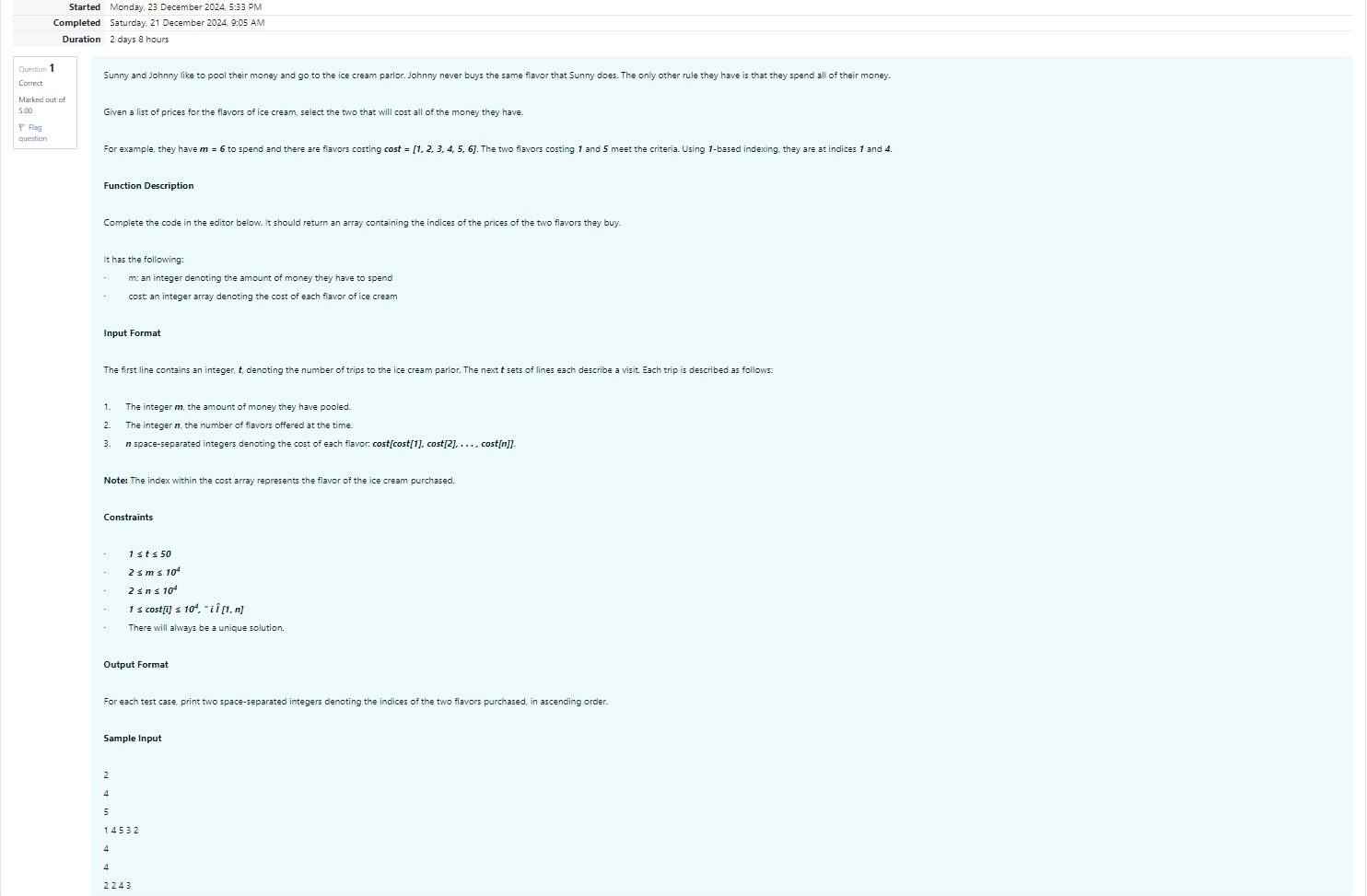
4

1. 2 4 3

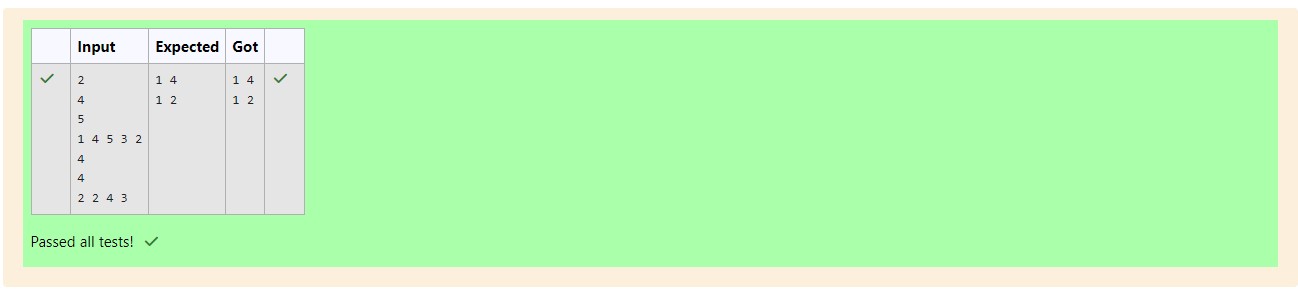
**Sample Output:**

1 4

1 2



Output:



**Q2) problem statement:**

Numerosthe Artist had two lists that were permutations of one another. He was very proud. Unfortunately, while transportingthem from one exhibition to another, some numbers were lost out ofthe first list. Can you find the missing numbers?

As an example, the array with some numbers missing, arr = [7, 2, 6, 5, 3, 5, 3]. Theoriginal array of numbers brr = [7, 2, 5, 4, 6, 3, 5, 3]. The numbers missingare [4, 6]. **Notes**:

If a number occurs multiple times in the lists, you must ensurethat the frequency of that number in bothlists is the same. If that is not the case, then it is also a missing number.

You have to print all the missing numbers in ascending order.Print each missing numberonce, even if itis missing multiple timesThe difference between the maximum and minimum number in the secondlist is less than or equal to 100.

Complete the code in the editor below. It shouldreturn a sorted array of missing numbers.

Ithas the following: arr: the arraywith missing numbers brr: the originalarray of numbers **InputFormat**:

Therewill be four lines of input:

1. n - thesize of the first list, arr

1. The next line contains n space-separated integersarr[i]

1. m - thesize of the second list, brr

1. The next line contains m space-separated integersbrr[i] Constraints:

1 ≤ n, m ≤ 2 × 105, 1 ≤ arr[i], brr[i] ≤ 2 × 104, Xmax − Xmin < 101 **OutputFormat**:

Output the missing numbersin ascending order. **SampleInput**:

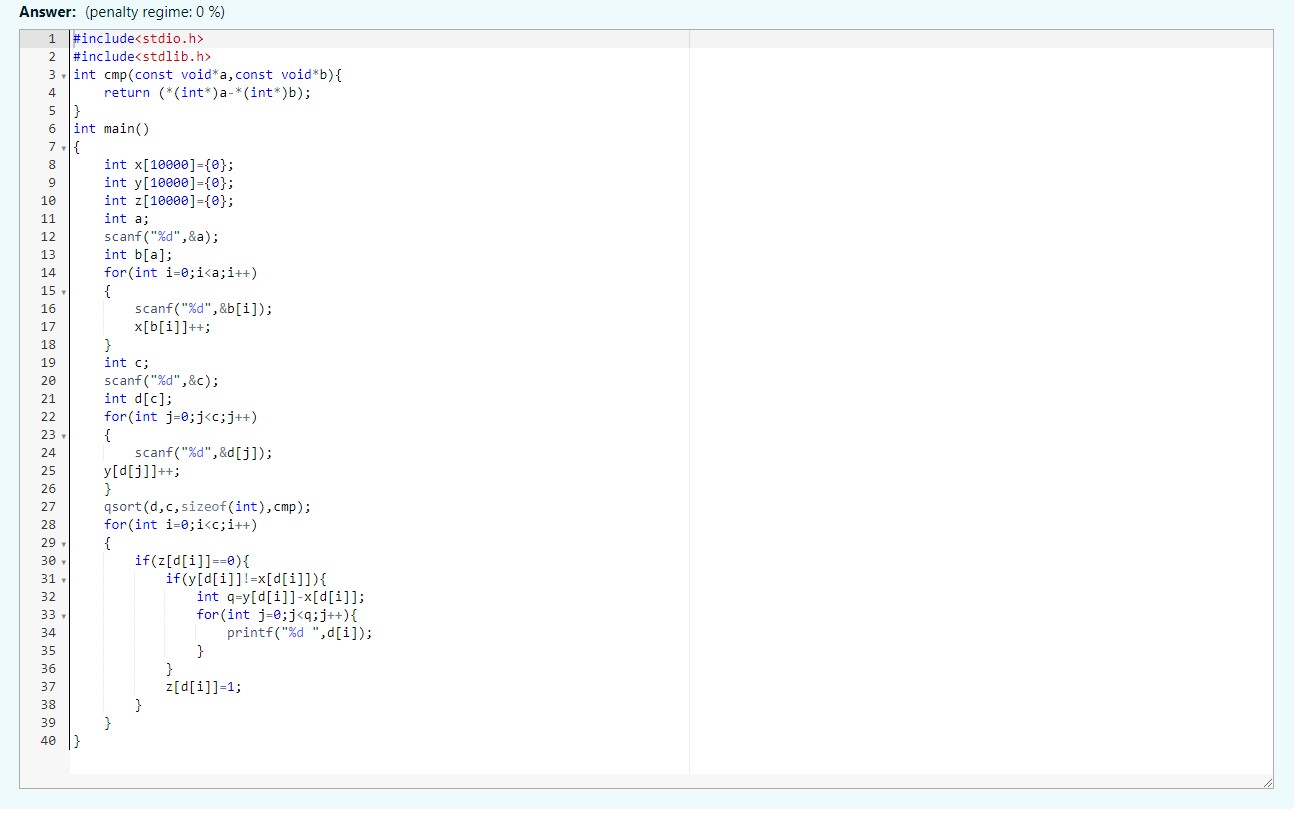
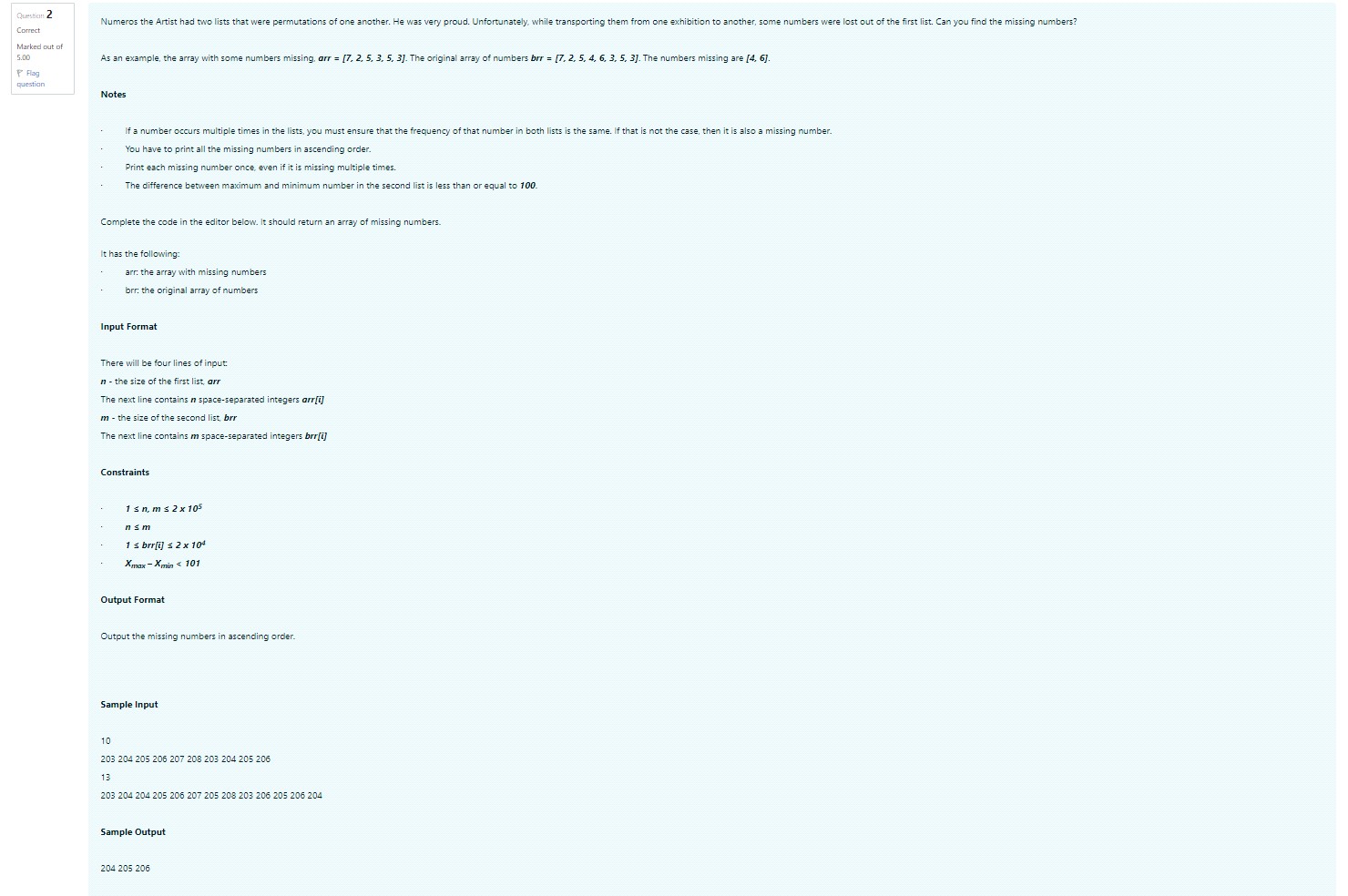
10

203 204 205 206 207 208 203 204 205 206

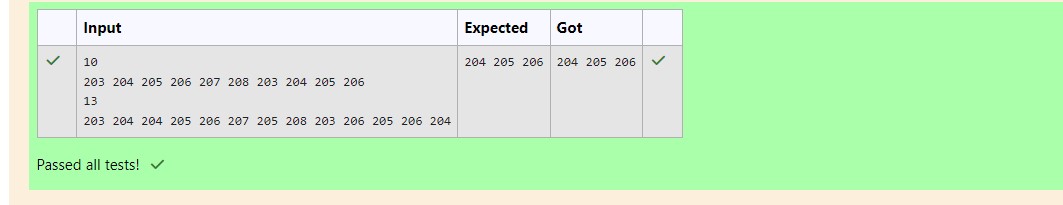
13

1. 204 204 205 206 207 205 208 203 206205 206 204 **SampleOutput**:

1. 205 206



Output:



**Q3) ProblemStatement**:

Watsongives Sherlock an array of integers. His challenge is to find an element of the array such that the sum of all elements to the left is equal to the sum of all elements to the right For instance, given the arrayarr 5 6 8 11 8 is between two subarrays that sum to 11 If your starting array is 1 that element satisfies the rule as left and right sums to 0 You will be givenarrays of integers and must determine whether there is an elementthat meets the criterion.

Complete the code in the editor below. It should return a string, either YES if there is an element meetingthe criterion or NO otherwise. It has the following: arr an array of integers. **InputFormat**:

Thefirst line contains T the number of test cases.

Thenext T pairs of lines each represent a test case.

-The first line contains n the number of elements in the array arr.

- The second line contains n space-separated integersarr i where 0

Constraints:

1

≤

T

≤ 10, 1 ≤

n

≤ 105, 1 ≤

arr[i]

≤ 2

x 104, 0

≤

i

≤

n

Output Format:

For each test case print YES if there exists an element in the array, such that the sum of the

elements on its left is equal to the sum of the elements on its right; otherwise print NO.

Sample Input 0:

2

3

2

3

1

4

1

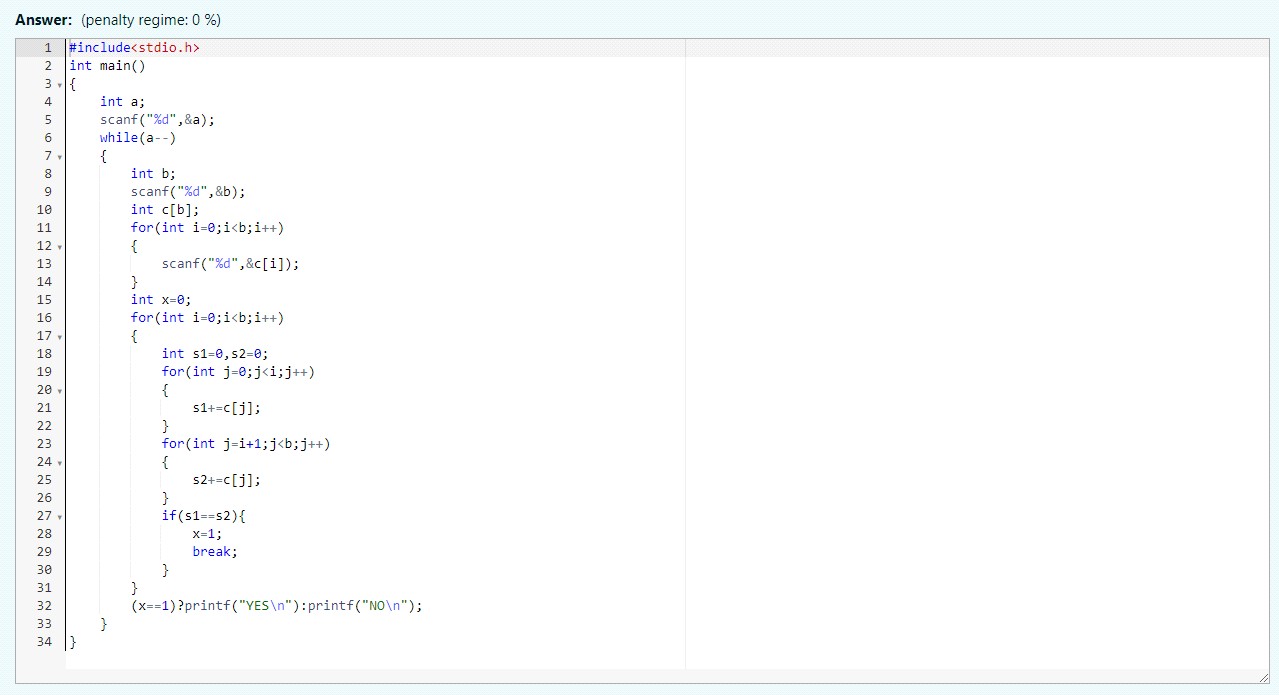
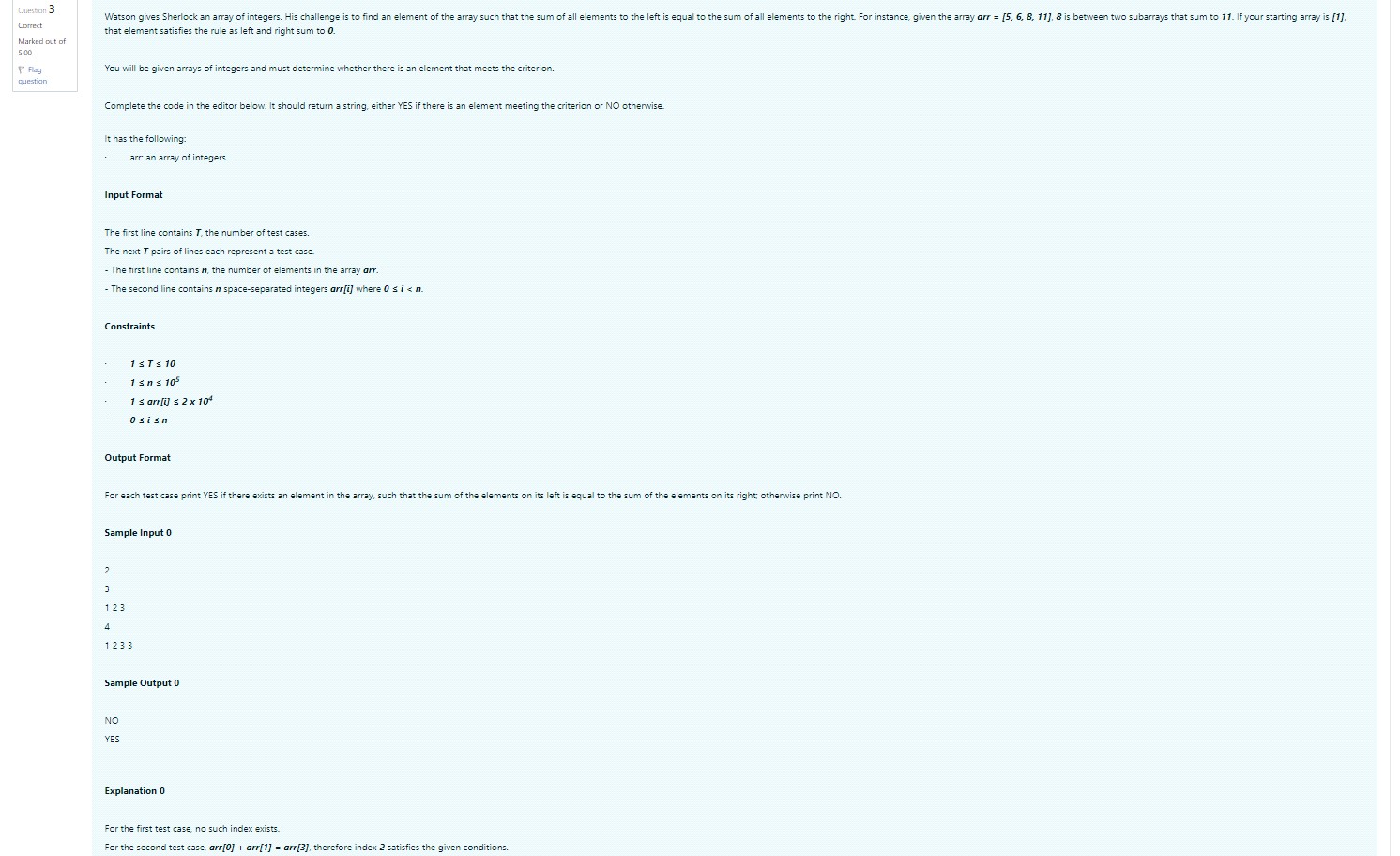
2 3

3

Sample Output 0:

NO

YES



Output:

