

# Mock CCC '24 Contest 1 S3 - Optimal Split Point

**Time limit:** 1.0s **Memory limit:** 512M

You are given an integer array  $a$  of length  $N$ . Your task is to find the *optimal split point* in the array such that the sum of the subarray to the left of the split point is as close as possible to the sum of the subarray to the right, without including the element indexed at the split point. You are allowed to move one element from the left subarray to the right or vice versa to achieve a better balance.

## Input Specification

The first line contains an integer  $N$ , representing the length of the array.

The second line contains  $N$  space-separated integers,  $a_1, \dots, a_N$ , representing the elements of the array.

The following table shows how the available 15 marks are distributed.

Marks Awarded	$N$	$a_i$
3 marks	$3 \leq N \leq 10^3$	$1 \leq a_i \leq 10^6$
9 marks	$3 \leq N \leq 10^5$	$1 \leq a_i \leq 10^6$
3 marks	$3 \leq N \leq 10^6$	$1 \leq a_i \leq 10^6$

## Output Specification

Output the index of the *optimal split point* and the index of the element that you moved between two subarrays separated by space. If you did not move any element between two subarrays, output `-1` for the second number. If multiple solutions exist, output the lexicographically smallest one. The index of the *optimal split point* takes precedence, and `-1` is given higher priority than the index of the moved element.

## Sample Input

```
7
1 2 1 3 3 2 1
```

## Sample Output

```
3 4
```

## Explanation for Sample

---

By splitting at index 3, we obtain the arrays of [1, 2] and [3, 3, 2, 1]. By moving the 4<sup>th</sup> element from the right array to the left array, we obtain arrays of [1, 2, 3] and [3, 2, 1]. The sum of both arrays is 6, meaning `3 4` is the most optimal split and move point. It can be shown that this is the lexicographically smallest solution.