C1. Sorted subarrays (easy version)

Difficulty: Easy **Time**: 1.5 s **Memory**: 1024 MB

by shcal

Note: the only difference between the easy and hard versions of this problem is that in the easy version, $n \leq 10^4$, while in the hard version, $n \leq 10^5$. The hard version is easiest to complete in C++.

Given an array $a_1, ..., a_n$ of length n, find all $1 \leq k \leq n$ for which the sums of consecutive length k subarrays are nondecreasing.

The i-th $(1 \leq i \leq n - k + 1)$ consecutive length k subarray of $a_1, ..., a_n$ is $a_i, ..., a_{i+k-1}$, and its sum is $a_i + ... + a_{i+k-1}$.

Input

The first line contains an integer n $(1 \leq n \leq 10^4)$, the length of the array.

The second line contains n space-separated integers $a_1, ..., a_n$ $(0 \leq a_i \leq 10^9)$.

Output

Output a space-separated, **sorted** list of all $1 \leq k \leq n$ with nondecreasing subarray sums.

Sample 1

Input

```
5
1 9 6 9 8
```

Output

```
2 4 5
```

Explanation

The subarrays of length 2 are $[1, 9], [9, 6], [6, 9], [9, 8]$. Respectively, their sums are $1 + 9 = 10, 9 + 6 = 15, 6 + 9 = 15, 9 + 8 = 17$. Since $[10, 15, 15, 17]$ is nondecreasing, $k = 2$ has the property that the consecutive length k subarrays of a have nondecreasing sums, so it is in the output list.

Similarly, one can see that $k = 4$ and $k = 5$ are valid choices and that no others exist.

Sample 2

Input

```
20
21 13 34 1 21 7 36 24 44 31 19 46 42 45 35
21 46 16 43 45
```

Output

```
12 14 16 18 19 20
```

Explanation

The second line is wrapped here, but not in the actual input.