

# Problem A. Sort the Array

**Time limit** 1000 ms  
**Mem limit** 262144 kB

Being a programmer, you like arrays a lot. For your birthday, your friends have given you an array  $a$  consisting of  $n$  **distinct** integers.

Unfortunately, the size of  $a$  is too small. You want a bigger array! Your friends agree to give you a bigger array, but only if you are able to answer the following question correctly: is it possible to sort the array  $a$  (in increasing order) by reversing **exactly one** segment of  $a$ ? See definitions of segment and reversing in the notes.

## Input

The first line of the input contains an integer  $n$  ( $1 \leq n \leq 10^5$ ) — the size of array  $a$ .

The second line contains  $n$  distinct space-separated integers:  $a[1], a[2], \dots, a[n]$  ( $1 \leq a[i] \leq 10^9$ ).

## Output

Print "yes" or "no" (without quotes), depending on the answer.

If your answer is "yes", then also print two space-separated integers denoting start and end (start must not be greater than end) indices of the segment to be reversed. If there are multiple ways of selecting these indices, print any of them.

### Sample 1

Input	Output
3 3 2 1	yes 1 3

### Sample 2

Input	Output
4 2 1 3 4	yes 1 2

**Sample 3**

Input	Output
4 3 1 2 4	no

**Sample 4**

Input	Output
2 1 2	yes 1 1

**Note**

Sample 1. You can reverse the entire array to get  $[1, 2, 3]$ , which is sorted.

Sample 3. No segment can be reversed such that the array will be sorted.

*Definitions*

A segment  $[l, r]$  of array  $a$  is the sequence  $a[l], a[l + 1], \dots, a[r]$ .

If you have an array  $a$  of size  $n$  and you reverse its segment  $[l, r]$ , the array will become:

$a[1], a[2], \dots, a[l - 2], a[l - 1], a[r], a[r - 1], \dots, a[l + 1], a[l], a[r + 1], a[r + 2], \dots, a[n - 1], a[n]$ .