Problem D. D

Time limit	1000 ms
Mem limit	131072 kB
OS	Linux

Quick sort is based on the Divide-and-conquer approach. In QuickSort(A, p, r), first, a procedure Partition(A, p, r) divides an array A[p..r] into two subarrays A[p..q-1] and A[q+1..r] such that each element of A[p..q-1] is less than or equal to A[q], which is, inturn, less than or equal to each element of A[q+1..r]. It also computes the index q.

In the conquer processes, the two subarrays A[p..q-1] and A[q+1..r] are sorted by recursive calls of QuickSort(A, p, q-1) and QuickSort(A, q+1, r).

Your task is to read a sequence A and perform the Partition based on the following pseudocode:

```
Partition(A, p, r)
1 x = A[r]
2 i = p-1
3 for j = p to r-1
4     do if A[j] <= x
5        then i = i+1
6             exchange A[i] and A[j]
7 exchange A[i+1] and A[r]
8 return i+1
```

Note that, in this algorithm, Partition always selects an element A[r] as a pivot element around which to partition the array A[p..r].

Input

The first line of the input includes an integer n, the number of elements in the sequence A.

In the second line, A_i (i = 1,2,...,n), elements of the sequence are given separated by space characters.

Output

Print the sorted sequence. Two contiguous elements of the sequence should be separated by a space character. The element which is selected as the pivot of the partition should be indicated by [ ].

Constraints

- $1 \le n \le 100{,}000$
- $0 \le A_i \le 100{,}000$

Sample Input 1

```
12
13 19 9 5 12 8 7 4 21 2 6 11
```

Sample Output 1

```
9 5 8 7 4 2 6 [11] 21 13 19 12
```