# Problem A. Sorter

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 1 second |
| Memory limit: | 256 megabytes |

Almas loves to sort numbers with specific rules that he comes up with himself. Today he wants to sort rows, each row contain m integer elements. He decided to sort rows in decreasing order of their sum, i.e. the row with larger sum will be the first. In case the sums are equal, then he order rows lexicographically, i.e. The row with least first element will be the first, if first elements are equal, then compare second element, third element, etc. He successfully finished sorting. He wanted to put the result in the closet, but accidentally put it in the shredder and divided everything into rows again. Help him sort it again.

## Input

The first line contains two numbers, n and m the number of rows and number of elements in a row. The next n lines contain m numbers, elements of each row.

- $1 \leq n, m \leq 500$
- $0 \leq row[i] \leq 5000$

## Output

Print rows in desired sorted order.

## Examples

| standard input | standard output |
|---|---|
| 5 3<br>1 2 3<br>1 2 4<br>1 3 2<br>6 0 0<br>2 3 2 | 1 2 4<br>2 3 2<br>1 2 3<br>1 3 2<br>6 0 0 |
| 2 3<br>16 3 18<br>5 10 8 | 16 3 18<br>5 10 8 |
| 5 5<br>17 16 21 10 48<br>1 34 5 41 42<br>29 8 50 21 43<br>21 39 9 26 23<br>45 44 29 8 24 | 29 8 50 21 43<br>45 44 29 8 24<br>1 34 5 41 42<br>21 39 9 26 23<br>17 16 21 10 48 |

# Problem B. Optimizing Program

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 2 seconds |
| Memory limit: | 256 megabytes |

The ICPC finals will be held soon, so Yergeldi and his team needs your help. While they were preparing for the competition, they faced an interesting task. You have a list of length N which consists of arrays of different lengths. You have one single operation, you can take any two arrays and merge them into one, the cost of the operation is equal to the sum of their lengths. As a result, you will have a list of N - 1 arrays. The process repeats until there is only one final array left. Find out for what minimum cost it is possible to combine all arrays.

## Input

The first line contains an integer N ($1 \leq N \leq 2 * 10^5$), the size of the list A. The next line contains n positive integers A1,A2,...,An ($1 \leq Ai \leq 2 * 10^5$), representing the sizes of arrays in the list A.

## Output

Print a single integer, the minimum cost of operations.

## Examples

| standard input | standard output |
|---|---|
| 4<br>6 5 3 9 | 45 |
| 10<br>42 18 63 26 19 15 11 29 26 24 | 869 |

## Note

Explanation for the first test case:

[6, 5, 3, 9] –> First, merge arrays of lengths 5 and 3 that will cost 8.

[6, 8, 9] –> Next, merge arrys of lengths 6 and 8 that will cost 14.

[14, 9] –> Finally, merge the remaining two arrays that will cost 23.

Therefore, the total cost for merging all arrays is $8 + 14 + 23 = 45$.

# Problem C. Nugman and Stack

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 2 seconds |
| Memory limit: | 256 megabytes |

One day Nugman was solving problems from LAB1 and he almost solved every problem in the laboratory work. Just one problem left that Nugman couldn't solve. Nugman asks for your help.

You are given N people in the queue, $i$-th person has age $a_i$. Each person wants to know if there is a younger person before him in a queue, in particular, the age of the closest person that is younger before him, otherwise print $-1$.

## Input

The first line of input consists of a single integer $1 \le N \le 10^5$ that describes the number of people in the queue. The next line contains N integers $1 \le a_i \le 10^9$, which describes the age of people in the i-th position.

## Output

If your function is implemented correctly, the program will print N numbers where i-th number is the answer for the ith person.

## Examples

| standard input | standard output |
|---|---|
| 5<br>2 1 5 8 3 | -1 -1 1 5 1 |
| 5<br>1 2 3 4 5 | -1 1 2 3 4 |

## Note

Go to the link

https://pastebin.com/RJfTeRK4, if you use C++

https://pastebin.com/E1k2AXJr, if you use Java

https://pastebin.com/kVZVLSBK, if you use Python

and take already written code from there.

Your only task is to implement function *push(int x)*

Solutions not involving template will be ignored.

# Problem D. Snake

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 1 second |
| Memory limit: | 256 megabytes |

Write a program that outputs the coordinates of elements from a array of size n$\times m$, which is filled like snake. Snake array - which is filled in like this:

- For all $j$ and $k$ $(j < k)$: $a_{ij} > a_{(i+1)j}$ .
- If $i$ is even then, for all $j$ and $k$ $(j < k)$ : $a_{ij} > a_{ik}$ .
- If $i$ is odd then, for all $j$ and $k$ $(j < k)$ : $a_{ij} < a_{ik}$.

Here is an example of 3×4 Snake array

25 23 20 19

13 15 17 18

12 10 9 8

## Input

The first line of input contains a single number t - the number of elements which you must find. $1 \leq t \leq 10000$

The next line contains t integers - the values of the elements that you need print their coordinates.

The next line of input contains 2 space-separated integers, n and m, the number of rows and the columns. $1 \leq n$,$m \leq 800$

The next n lines contain m integers. Snake array n $\times$ m, $10^7 \leq a_{ij} \leq 10^7$ for each $0 \leq i \leq n$, $0 \leq j \leq m$

## Output

Print k lines the answer with coordinates for each case. If the given element is not in the snake array, then print -1.

## Examples

| standard input | standard output |
|---|---|
| 5<br>10 15 13 8 23<br>3 4<br>25 23 20 19<br>13 15 17 18<br>12 10 9 8 | 2 1<br>1 1<br>1 0<br>2 3<br>0 1 |
| 8<br>1 7 17 12 6 15 18 20<br>5 5<br>25 24 23 22 21<br>16 17 18 19 20<br>15 14 13 12 11<br>6 7 8 9 10<br>5 4 3 2 1 | 4 4<br>3 1<br>1 1<br>2 3<br>3 0<br>2 0<br>1 2<br>1 4 |
| 4<br>-2 7 8 4<br>2 3<br>9 8 5<br>-1 3 4 | -1<br>-1<br>0 1<br>1 2 |

## Note

In the third example, the elements -2 and 7 is do not exist. Therefore, you should print -1.

# Problem E. Triangles

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 1 second |
| Memory limit: | 256 megabytes |

You are given N integers in order of their insertion to Binary Search Tree. Your task is to find the number of triangles with length k. Triangle of length $k$ in Binary Search Tree is a triplet of nodes $p, l, r$, such that $p$ is ancestor of $l$ and $r$ and distance from $p$ to $l$ and from $p$ to $r$ are both equal to $k$. Distance between nodes is number of edges on the path between these nodes. See the explanation below.

## Input

The first line consists of an integer $1 \leq N \leq 1000$ - number of nodes in the Binary Search Tree. The second line contains N integers $1 \leq a[i] \leq 10^6$ the value of each node in the Binary Search Tree in order of their insertion. It is guaranteed that there are no duplicates.

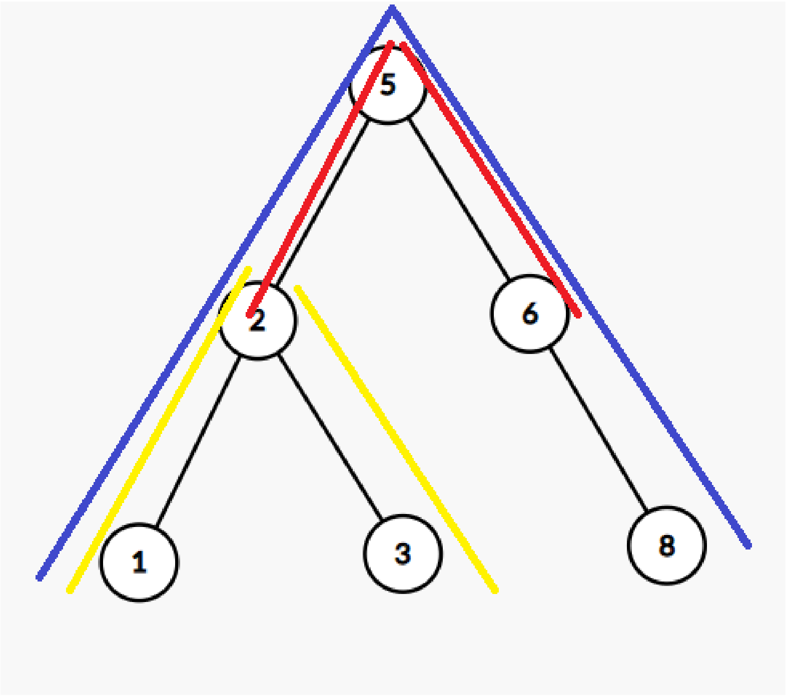
## Output

Print N - 1 number, the number of triangles of length $2 \leq k \leq N$ in the resulting Binary Search Tree.

## Examples

| standard input | standard output |
|---|---|
| `5`<br>`5 2 6 8 1` | `1 1 0 0` |
| `6`<br>`5 6 2 3 1 8` | `2 1 0 0 0` |

## Note



**Explanation** : As you can see from this picture there are two triangles with length 2 (yellow and red), and one with length 3(blue).