

Problem A. Increasing Array

Time limit 1000 ms

Mem limit 524288 kB

You are given an array of n integers. You want to modify the array so that it is increasing, i.e., every element is at least as large as the previous element.

On each move, you may increase the value of any element by one. What is the minimum number of moves required?

Input

The first input line contains an integer n : the size of the array.

Then, the second line contains n integers x_1, x_2, \dots, x_n : the contents of the array.

Output

Print the minimum number of moves.

Constraints

- $1 \leq n \leq 2 \cdot 10^5$
- $1 \leq x_i \leq 10^9$

Example

Input	Output
5 3 2 5 1 7	5

Problem B. Missing Number

Time limit 1000 ms

Mem limit 524288 kB

You are given all numbers between $1, 2, \dots, n$ except one. Your task is to find the missing number.

Input

The first input line contains an integer n .

The second line contains $n - 1$ numbers. Each number is distinct and between 1 and n (inclusive).

Output

Print the missing number.

Constraints

- $2 \leq n \leq 2 \cdot 10^5$

Example

Input	Output
5 2 3 1 5	4

Problem C. Chessboard and Queens

Time limit 1000 ms

Mem limit 524288 kB

Your task is to place eight queens on a chessboard so that no two queens are attacking each other. As an additional challenge, each square is either free or reserved, and you can only place queens on the free squares. However, the reserved squares do not prevent queens from attacking each other.

How many possible ways are there to place the queens?

Input

The input has eight lines, and each of them has eight characters. Each square is either free (`.`) or reserved (`*`).

Output

Print one integer: the number of ways you can place the queens.

Example

Input	Output
<pre>*....** ...*... </pre>	65

Problem D. Left Rotate the Array

OS Linux

A *left rotation* operation on an array of size n shifts each of the array's elements **1** unit to the left. For example, if **2** left rotations are performed on array **[1, 2, 3, 4, 5]**, then the array would become **[3, 4, 5, 1, 2]**.

Given an array of n integers and a number, d , perform d left rotations on the array. Then print the updated array as a single line of space-separated integers.

Input Format

The first line contains two space-separated integers denoting the respective values of n (the number of integers) and d (the number of left rotations you must perform).
The second line contains n space-separated integers describing the respective elements of the array's initial state.

Constraints

- $1 \leq n \leq 10^5$
- $1 \leq d \leq n$
- $1 \leq a_i \leq 10^6$

Output Format

Print a single line of n space-separated integers denoting the final state of the array after performing d left rotations.

Input	Output
5 4 1 2 3 4 5	5 1 2 3 4

Explanation

When we perform $d = 4$ left rotations, the array undergoes the following sequence of changes:

$$[1, 2, 3, 4, 5] \rightarrow [2, 3, 4, 5, 1] \rightarrow [3, 4, 5, 1, 2] \rightarrow [4, 5, 1, 2, 3] \rightarrow [5, 1, 2, 3, 4]$$

Thus, we print the array's final state as a single line of space-separated values, which is **5 1 2 3 4**.

Problem E. Mr. Zero

Time limit 1000 ms

Mem limit 524288 kB

Mr. Zero has a matrix Z consisting of m rows and n columns. The rows are numbered from 1 to m , columns are numbered from 1 to n . Element at row i ($1 \leq i \leq m$) and column j ($1 \leq j \leq n$) is denoted as Z_{ij} . All elements of Z are either 0 or 1 .

Mr. Zero gives you a task. You need to calculate how many positions are in the matrix Z where all elements of i th row and j th column contain 0 .

Input

The first line contains two integer m and n ($1 \leq m, n \leq 1000$), number of rows and number of columns of matrices respectively.

The next m lines each contain n integers separated by spaces describing rows of matrix Z (each element of Z is either 0 or 1).

Output

In the first line, print the desired output.

Sample

Input	Output
3 3 1 0 0 0 0 0 1 0 0	2

In the sample case:

All elements of **second row** and **second column** contain **0**.

All elements of **second row** and **third column** contain **0**.

Problem F. The Universe Loves Minimum Steps!

Time limit 1000 ms

Mem limit 524288 kB

Given an array A of N integers. Your task is to make all the integers 0 by performing the following two operations.

- Operation 1: You can select K integers arbitrarily from the array where $1 \leq K \leq N$.
- Operation 2: From the K selected integers, you must have to arbitrarily select A and B integers. You have to arbitrarily select A integers and increase all integers by 1 where $0 \leq A \leq K$ and you have to arbitrarily select B integers and decrease all integers by 1 where $0 \leq B \leq K$.

You must perform the above two operations to make all the elements of the array to be 0.

In one step, you can perform the above two operations.

Now your job is to find the minimum number of steps needed to make all the elements of the array to be 0.

Input

Input starts with an integer T ($T \leq 100$), denoting the number of test cases.

Each case contains an integer N ($1 \leq N \leq 100$) and then next line N integers follow. Each integer is between -100 to 100 inclusive.

Output

For each case, print the case number and minimum number of steps needed.

Sample

Input	Output
2 3 7 7 7 4 1 -2 1 2	Case 1: 7 Case 2: 2

This problem was authored for CodeMask Championship 2016 and is being hosted on Toph per organizer's request.

Problem G. Decent Arrays

Time limit 1000 ms

Mem limit 524288 kB

Given N numbers determine if they are in ascending order.

Input

The first line of the input will contain N ($0 < N < 100$).

The following line will contain N integers, each between 1 and 1000.

Output

Print “Yes” if the numbers were in ascending order, otherwise “No”.

Sample

Input	Output
3 1 2 3	Yes

Problem H. 2D Array - DS

OS Linux

Given a 6×6 2D array, *arr*, an hourglass is a subset of values with indices falling in the following pattern:

```
a b c
  d
e f g
```

There are **16** hourglasses in a 6×6 array. The *hourglass sum* is the sum of the values in an hourglass. Calculate the hourglass sum for every hourglass in *arr*, then print the *maximum* hourglass sum.

Example

arr =

1	-9	-9	-9	1	1	1
2	0	-9	0	4	3	2
3	-9	-9	-9	1	2	3
4	0	0	8	6	6	0
5	0	0	0	-2	0	0
6	0	0	1	2	4	0

The **16** hourglass sums are:

1	-63,	-34,	-9,	12,
2	-10,	0,	28,	23,
3	-27,	-11,	-2,	10,
4	9,	17,	25,	18

The highest hourglass sum is **28** from the hourglass beginning at row **1**, column **2**:

1	0	4	3
2		1	
3	8	6	6

Note: If you have already solved the Java domain's *Java 2D Array* challenge, you may wish to skip this challenge.

Function Description

Complete the function *hourglassSum* with the following parameter(s):

- *int arr[6][6]*: a 2-D array of integers

Returns

- *int*: the maximum hourglass sum

Input Format

Each of the 6 lines of inputs *arr[i]* contains 6 space-separated integers *arr[i][j]*.

Constraints

- $-9 \leq arr[i][j] \leq 9$
- $0 \leq i, j \leq 5$

Input	Output
1 1 1 0 0 0 0 1 0 0 0 0 1 1 1 0 0 0 0 0 2 4 4 0 0 0 0 2 0 0 0 0 1 2 4 0	19

Explanation

arr contains the following hourglasses:

1 1 1 1 1 0 1 0 0 0 0 0
1 0 0 0 0
1 1 1 1 1 0 1 0 0 0 0 0

0 1 0 1 0 0 0 0 0 0 0 0
1 1 0 0
0 0 2 0 2 4 2 4 4 4 4 0

1 1 1 1 1 0 1 0 0 0 0 0
0 2 4 4
0 0 0 0 0 2 0 2 0 2 0 0

0 0 2 0 2 4 2 4 4 4 4 0
0 0 2 0
0 0 1 0 1 2 1 2 4 2 4 0

The hourglass with the maximum sum (**19**) is:

1 | 2 4 4
2 | 2
3 |

1 2 4

Problem I. Building a List

OS Linux

Chan has decided to make a list of all possible combinations of letters of a given string S. If there are two strings with the same set of characters, print the lexicographically smallest arrangement of the two strings.

abc acb cab bac bca

all the above strings' lexicographically smallest string is abc.

Each character in the string S is unique. Your task is to print the entire list of Chan's in lexicographic order.

for string *abc*, the list in lexicographic order is given below

a ab abc ac b bc c

Input Format

The first line contains the number of test cases T. T testcases follow.

Each testcase has 2 lines. The first line is an integer N (the length of the string).

The second line contains the string S.

Output Format

For each testcase, print the entire list of combinations of string S, with each combination of letters in a newline.

Constraints

$0 < T < 50$

$1 < N < 16$

string S contains only small alphabets(a-z)

Input	Output
2 2 ab 3 xyz	a ab b x xy xyz xz y yz z

Explanation

In the first case we have ab , the possibilities are a , ab and b . Similarly, all combination of characters of xyz .