[DivC] contest #3

A. Shaass and Lights

1 second, 256 megabytes

There are n lights aligned in a row. These lights are numbered 1 to n from left to right. Initially some of the lights are switched on. Shaass wants to switch all the lights on. At each step he can switch a light on (this light should be switched off at that moment) if there's at least one adjacent light which is already switched on.

He knows the initial state of lights and he's wondering how many different ways there exist to switch all the lights on. Please find the required number of ways modulo 1000000007 $(10^9 + 7)$.

Input

The first line of the input contains two integers n and m where n is the number of lights in the sequence and m is the number of lights which are initially switched on, $(1 \le n \le 1000, 1 \le m \le n)$. The second line contains m distinct integers, each between 1 to n inclusive, denoting the indices of lights which are initially switched on.

Output

In the only line of the output print the number of different possible ways to switch on all the lights modulo 1000000007 $(10^9 + 7)$.

input
3 1 1

output
1

input
4 2 1 4

output
2

input
11 2 4 8

output
6720

B. Feed Cats

3 seconds, 512 megabytes

There is a fun game where you need to feed cats that come and go. The level of the game consists of n steps. There are m cats; the cat i is present in steps from l_i to r_i, inclusive. In each step, you can feed all the cats that are currently present or do nothing.

If you feed the same cat more than once, it will overeat, and you will immediately lose the game. Your goal is to feed as many cats as possible without causing any cat to overeat.

Find the maximum number of cats you can feed.

Formally, you need to select several integer points from the segment from 1 to n in such a way that among given segments, none covers two or more of the selected points, and as many segments as possible cover one of the selected points.

Input

The first line of input contains a single integer t $(1 \le t \le 10^4)$ — the number of test cases. Then the descriptions of the test cases follow.

The first line of each test case contains two integers n and m ($1 \le n \le 10^6$, $1 \le m \le 2 \cdot 10^5$).

The i-th of the next m lines contains a pair of integers l_i and r_i ($1 \le l_i \le r_i \le n$).

The sum of n for all tests does not exceed 10^6, the sum of m for all tests does not exceed $2 \cdot 10^5$.

Output

For each test case, print a single integer, the maximum number of cats you can feed.

input
3 15 6 2 10 3 5 2 4 7 7 8 12 11 11 1000 1 1 1000 5 10 1 2 3 4 3 4 3 4 3 4 1 1 1 2 3 3 3 4 3 4

output
5 1 10

In the first example, one of the ways to feed five cats is to feed at steps 4 and 11.

- At step 4, cats 1, 2, and 3 will be fed.
- At step 11, cats 5 and 6 will be fed.

C. Increase/Decrease/Copy

2 seconds, 256 megabytes

You are given two integer arrays: array a of length n and array b of length $n + 1$.

You can perform the following operations any number of times in any order:

- choose any element of the array a and increase it by 1;
- choose any element of the array a and decrease it by 1;

- choose any element of the array a, copy it and append the copy to the end of the array a.

Your task is to calculate the minimum number of aforementioned operations (possibly zero) required to transform the array a into the array b. It can be shown that under the constraints of the problem, it is always possible.

Input

The first line contains a single integer t ($1 \le t \le 10^4$) — the number of test cases.

Each test case consists of three lines:

- the first line contains a single integer n ($1 \le n \le 2 \cdot 10^5$);
- the second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^9$);
- the third line contains $n+1$ integers $b_1, b_2, \ldots, b_{n+1}$ ($1 \le b_i \le 10^9$).

Additional constraint on the input: the sum of n over all test cases doesn't exceed $2 \cdot 10^5$.

Output

For each test case, print a single integer — the minimum number of operations (possibly zero) required to transform the array a into the array b.

input
3 1 2 1 3 2 3 3 3 3 3 4 4 2 1 2 2 1 5 2 3

output
3 1 8

In the first example, you can transform a into b as follows: $[2] \to [2,2] \to [1,2] \to [1,3]$.

D. Catch the Coin

2 seconds, 256 megabytes

Monocarp visited a retro arcade club with arcade cabinets. There got curious about the "Catch the Coin" cabinet.

The game is pretty simple. The screen represents a coordinate grid such that:

- the X-axis is directed from left to right;
- the Y-axis is directed from bottom to top;
- the center of the screen has coordinates $(0,0)$.

At the beginning of the game, the character is located in the center, and n coins appear on the screen — the i-th coin is at coordinates (x_i, y_i). The coordinates of all coins are different and not equal to $(0,0)$.

In one second, Monocarp can move the character in one of eight directions. If the character is at coordinates (x,y), then it can end up at any of the coordinates $(x,y+1)$, $(x+1,y+1)$, $(x+1,y)$, $(x+1,y-1)$, $(x,y-1)$, $(x-1,y-1)$, $(x-1,y)$, $(x-1,y+1)$.

If the character ends up at the coordinates with a coin, then Monocarp collects that coin.

After Monocarp makes a move, all coins fall down by 1, that is, they move from (x,y) to $(x,y-1)$. You can assume that the game field is infinite in all directions.

Monocarp wants to collect at least one coin, but cannot decide which coin to go for. Help him determine, for each coin, whether he can collect it.

Input

The first line contains a single integer n ($1 \le n \le 500$) — the number of coins.

In the i-th of the next n lines, two integers x_i and y_i ($-50 \le x_i, y_i \le 50$) are written — the coordinates of the i-th coin. The coordinates of all coins are different. No coin is located at $(0,0)$.

Output

For each coin, print "YES" if Monocarp can collect it. Otherwise, print "NO".

input
5 24 42 -2 -1 -1 -2 0 -50 15 0

output
YES YES NO NO YES

Pay attention to the second coin in the example. Monocarp can first move from $(0,0)$ to $(-1,-1)$. Then the coin falls 1 down and ends up at $(-2,-2)$. Finally, Monocarp moves to $(-2,-2)$ and collects the coin.

E. Choosing Cubes

1 second, 256 megabytes

Dmitry has n cubes, numbered from left to right from 1 to n. The cube with index f is his favorite.

Dmitry threw all the cubes on the table, and the i-th cube showed the value a_i ($1 \le a_i \le 100$). After that, he arranged the cubes in non-increasing order of their values, from largest to smallest. If two cubes show the same value, they can go in any order.

After sorting, Dmitry removed the first k cubes. Then he became interested in whether he removed his favorite cube (note that its position could have changed after sorting).

For example, if $n = 5$, $f = 2$, $a = [4,3,3,2,3]$ (the favorite cube is highlighted in green), and $k = 2$, the following could have happened:

- After sorting $a = [4,3,3,3,2]$, since the favorite cube ended up in the second position, it will be removed.
- After sorting $a = [4,3,3,3,2]$, since the favorite cube ended up in the third position, it will not be removed.

Input

The first line contains an integer t ($1 \le t \le 1000$) — the number of test cases. Then follow the descriptions of the test cases.

The first line of each test case description contains three integers n, f, and k ($1 \le f, k \le n \le 100$) — the number of cubes, the index of Dmitry's favorite cube, and the number of removed cubes, respectively.

The second line of each test case description contains n integers a_i ($1 \le a_i \le 100$) — the values shown on the cubes.

Output

For each test case, output one line — "YES" if the cube will be removed in all cases, "NO" if it will not be removed in any case, "MAYBE" if it may be either removed or left.

You can output the answer in any case. For example, the strings "YES", "nO", "mAyBe" will be accepted as answers.

input

```
12
5 2 2
4 3 3 2 3
5 5 3
4 2 1 3 5
5 5 2
5 2 4 1 3
5 5 5
1 2 5 4 3
5 5 4
3 1 2 4 5
5 5 5
4 3 2 1 5
6 5 3
1 2 3 1 2 3
10 1 1
1 1 1 1 1 1 1 1 1 1
1 1 1
42
5 2 3
2 2 1 1 2
2 1 1
2 1
5 3 1
3 3 2 3 2
```

output

```
MAYBE
YES
NO
YES
YES
YES
MAYBE
MAYBE
YES
YES
YES
NO
```

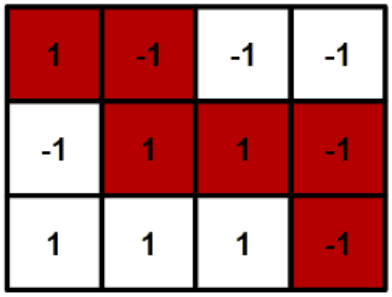
F. Zero Path

2 seconds, 256 megabytes

You are given a grid with n rows and m columns. We denote the square on the i-th ($1 \le i \le n$) row and j-th ($1 \le j \le m$) column by (i, j) and the number there by a_{ij}. All numbers are equal to 1 or to -1.

You start from the square $(1, 1)$ and can move one square down or one square to the right at a time. In the end, you want to end up at the square (n, m).

Is it possible to move in such a way so that the sum of the values written in all the visited cells (including a_{11} and a_{nm}) is 0?



Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 10^4$). Description of the test cases follows.

The first line of each test case contains two integers n and m ($1 \le n, m \le 1000$) — the size of the grid.

Each of the following n lines contains m integers. The j-th integer on the i-th line is a_{ij} ($a_{ij} = 1$ or -1) — the element in the cell (i, j).

It is guaranteed that the sum of $n \cdot m$ over all test cases does not exceed 10^6.

Output

For each test case, print "YES" if there exists a path from the top left to the bottom right that adds up to 0, and "NO" otherwise. You can output each letter in any case.

input

```
5
1 1
1
1 2
1 -1
1 4
1 -1 1 -1
3 4
1 -1 -1 -1
-1 1 1 -1
1 1 1 -1
3 4
1 -1 1 1
-1 1 -1 1
1 -1 1 1
```

output

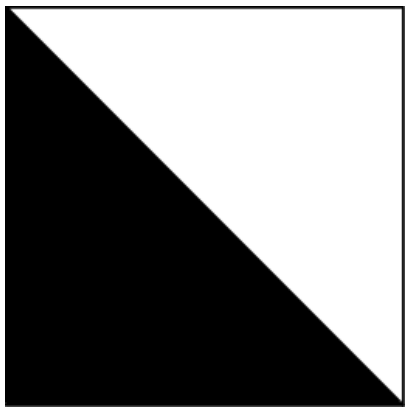
```
NO
YES
YES
YES
NO
```

One possible path for the fourth test case is given in the picture in the statement.

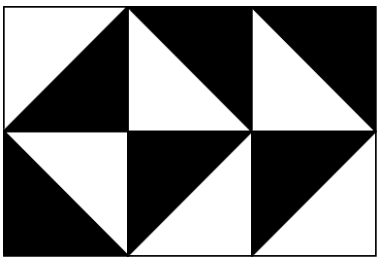
G. Tiles

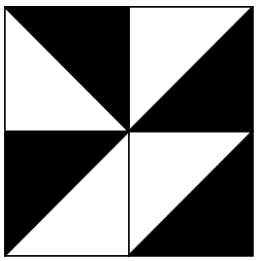
1 second, 256 megabytes

Bob is decorating his kitchen, more precisely, the floor. He has found a prime candidate for the tiles he will use. They come in a simple form factor — a square tile that is diagonally split into white and black part as depicted in the figure below.



The dimension of this tile is perfect for this kitchen, as he will need exactly $w \times h$ tiles without any scraps. That is, the width of the kitchen is w tiles, and the height is h tiles. As each tile can be rotated in one of four ways, he still needs to decide on how exactly he will tile the floor. There is a single aesthetic criterion that he wants to fulfil: two adjacent tiles must not share a colour on the edge — i.e. one of the tiles must have a white colour on the shared border, and the second one must be black.





The picture on the left shows one valid tiling of a 3×2 kitchen. The picture on the right shows an invalid arrangement, as the bottom two tiles touch with their white parts.

Find the number of possible tilings. As this number may be large, output its remainder when divided by 998244353 (a prime number).

Input

The only line contains two space separated integers w, h ($1 \le w, h \le 1\,000$) — the width and height of the kitchen, measured in tiles.

Output

Output a single integer n — the remainder of the number of tilings when divided by 998244353.

input
2 2
output
16

input
2 4
output
64

H. Nauuo and Circle

2 seconds, 256 megabytes

Nauuo is a girl who loves drawing circles.

One day she has drawn a circle and wanted to draw a tree on it.

The tree is a connected undirected graph consisting of n nodes and $n-1$ edges. The nodes are numbered from 1 to n.

Nauuo wants to draw a tree on the circle, the nodes of the tree should be in n **distinct** points on the circle, and the edges should be straight without crossing each other.

"Without crossing each other" means that every two edges have no common point or the only common point is an endpoint of both edges.

Nauuo wants to draw the tree using a permutation of n elements. A permutation of n elements is a sequence of integers $p_1, p_2, \ldots, p_n$ in which every integer from 1 to n appears exactly once.

After a permutation is chosen Nauuo draws the i-th node in the p_i-th point on the circle, then draws the edges connecting the nodes.

The tree is given, Nauuo wants to know how many permutations are there so that the tree drawn satisfies the rule (the edges are straight without crossing each other). She only wants to know the answer modulo 998244353, can you help her?

It is obvious that whether a permutation is valid or not does not depend on which n points on the circle are chosen.

Input

The first line contains a single integer n ($2 \le n \le 2 \cdot 10^5$) — the number of nodes in the tree.

Each of the next $n-1$ lines contains two integers u and v ($1 \le u, v \le n$), denoting there is an edge between u and v.

It is guaranteed that the given edges form a tree.

Output

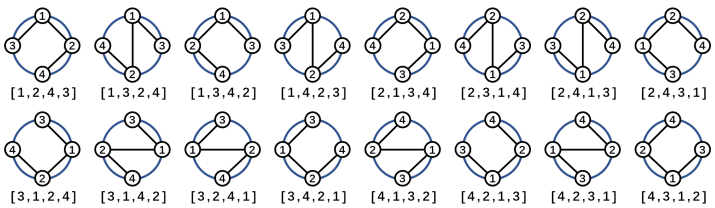
The output contains a single integer — the number of permutations suitable to draw the given tree on a circle satisfying the rule, modulo 998244353.

input
4 1 2 1 3 2 4
output
16

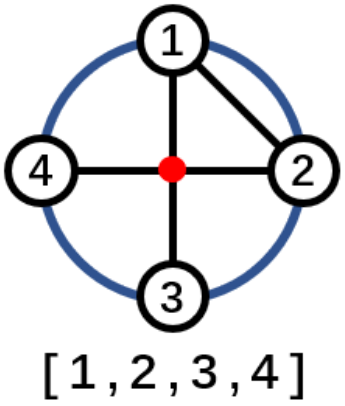
input
4 1 2 1 3 1 4
output
24

Example 1

All valid permutations and their spanning trees are as follows.



Here is an example of invalid permutation: the edges $(1, 3)$ and $(2, 4)$ are crossed.



Example 2

Every permutation leads to a valid tree, so the answer is $4! = 24$.

I. Fox And Names

2 seconds, 256 megabytes

Fox Ciel is going to publish a paper on FOCS (Foxes Operated Computer Systems, pronounce: "Fox"). She heard a rumor: the authors list on the paper is always sorted in the lexicographical order.

After checking some examples, she found out that sometimes it wasn't true. On some papers authors' names weren't sorted in lexicographical order in normal sense. But it was always true that after some modification of the order of letters in alphabet, the order of authors becomes lexicographical!

She wants to know, if there exists an order of letters in Latin alphabet such that the names on the paper she is submitting are following in the lexicographical order. If so, you should find out any such order.

Lexicographical order is defined in following way. When we compare s and t, first we find the leftmost position with differing characters: $s_i \neq t_i$. If there is no such position (i. e. s is a prefix of t or vice versa) the shortest string is less. Otherwise, we compare characters s_i and t_i according to their order in alphabet.

Input

The first line contains an integer n ($1 \leq n \leq 100$): number of names.

Each of the following n lines contain one string $name_i$ ($1 \leq |name_i| \leq 100$), the i-th name. Each name contains only lowercase Latin letters. All names are different.

Output

If there exists such order of letters that the given names are sorted lexicographically, output any such order as a permutation of characters 'a'–'z' (i. e. first output the first letter of the modified alphabet, then the second, and so on).

Otherwise output a single word "Impossible" (without quotes).

input
3 rivest shamir adleman
output
bcdefghijklmnopqrsatuvwxyz

input
10 tourist petr wjmzbmr yeputons vepifanov scottwu oooooooooooooooo subscriber rowdark tankengineer
output
Impossible

input
10 petr egor endagorion feferivan ilovetanyaromanova kostka dmitriyh maratsnowbear bredorjaguarturnik cgyforever
output
aghjlnopefikdmbcqrstuvwxyz

input
7 car care careful carefully becarefuldontforgetsomething otherwiseyouwillbehacked goodluck
output
acbdefhijklmnogpqrstuvwxyz

J. Nikita and string

2 seconds, 256 megabytes

One day Nikita found the string containing letters "a" and "b" only.

Nikita thinks that string is beautiful if it can be cut into 3 strings (possibly empty) without changing the order of the letters, where the 1-st and the 3-rd one contain only letters "a" and the 2-nd contains only letters "b".

Nikita wants to make the string beautiful by removing some (possibly none) of its characters, but without changing their order. What is the maximum length of the string he can get?

Input

The first line contains a non-empty string of length not greater than $5\,000$ containing only lowercase English letters "a" and "b".

Output

Print a single integer — the maximum possible size of beautiful string Nikita can get.

input
abba

output
4

input
bab
output
2

It the first sample the string is already beautiful.

In the second sample he needs to delete one of "b" to make it beautiful.

K. Count the Arrays

2 seconds, 512 megabytes

Your task is to calculate the number of arrays such that:

- each array contains n elements;
- each element is an integer from 1 to m;
- for each array, there is **exactly** one pair of equal elements;
- for each array a, there exists an index i such that the array is **strictly ascending** before the i-th element and **strictly descending** after it (formally, it means that $a_j < a_{j+1}$, if $j < i$, and $a_j > a_{j+1}$, if $j \geq i$).

Input

The first line contains two integers n and m ($2 \leq n \leq m \leq 2 \cdot 10^5$).

Output

Print one integer — the number of arrays that meet all of the aforementioned conditions, taken modulo 998244353.

input
3 4
output
6

input
3 5
output
10

input
42 1337
output
806066790

input
100000 200000
output
707899035

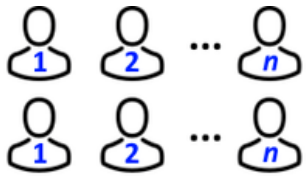
The arrays in the first example are:

- $[1, 2, 1]$;
- $[1, 3, 1]$;
- $[1, 4, 1]$;
- $[2, 3, 2]$;
- $[2, 4, 2]$;
- $[3, 4, 3]$.

L. Basketball Exercise

2 seconds, 256 megabytes

Finally, a basketball court has been opened in SIS, so Demid has decided to hold a basketball exercise session. $2 \cdot n$ students have come to Demid's exercise session, and he lined up them into two rows of the same size (there are exactly n people in each row). Students are numbered from 1 to n in each row in order from left to right.



Now Demid wants to choose a team to play basketball. He will choose players from left to right, and the index of each chosen player (excluding the first one **taken**) will be strictly greater than the index of the previously chosen player. To avoid giving preference to one of the rows, Demid chooses students in such a way that no consecutive chosen students belong to the same row. The first student can be chosen among all $2n$ students (there are no additional constraints), and a team can consist of any number of students.

Demid thinks, that in order to compose a perfect team, he should choose students in such a way, that the total height of all chosen students is maximum possible. Help Demid to find the maximum possible total height of players in a team he can choose.

Input

The first line of the input contains a single integer n ($1 \leq n \leq 10^5$) — the number of students in each row.

The second line of the input contains n integers $h_{1,1}, h_{1,2}, \ldots, h_{1,n}$ ($1 \leq h_{1,i} \leq 10^9$), where $h_{1,i}$ is the height of the i-th student in the first row.

The third line of the input contains n integers $h_{2,1}, h_{2,2}, \ldots, h_{2,n}$ ($1 \leq h_{2,i} \leq 10^9$), where $h_{2,i}$ is the height of the i-th student in the second row.

Output

Print a single integer — the maximum possible total height of players in a team Demid can choose.

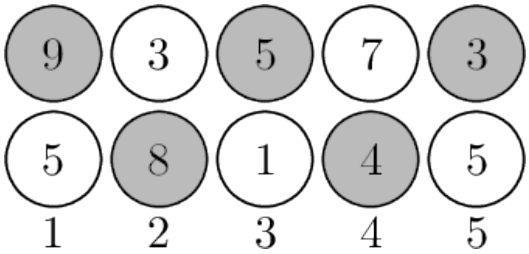
input
5 9 3 5 7 3 5 8 1 4 5
output
29

input
3 1 2 9 10 1 1
output
19

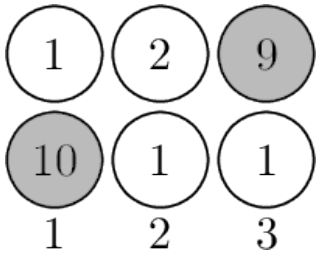
input
1 7 4

output
7

In the first example Demid can choose the following team as follows:



In the second example Demid can choose the following team as follows:



M. Mahmoud and a Message

2 seconds, 256 megabytes

Mahmoud wrote a message s of length n. He wants to send it as a birthday present to his friend Moaz who likes strings. He wrote it on a magical paper but he was surprised because some characters disappeared while writing the string. That's because this magical paper doesn't allow character number i in the English alphabet to be written on it in a string of length more than a_i. For example, if $a_1 = 2$ he can't write character 'a' on this paper in a string of length 3 or more. String "aa" is allowed while string "aaa" is not.

Mahmoud decided to split the message into some non-empty substrings so that he can write every substring on an independent magical paper and fulfill the condition. The sum of their lengths should be n and they shouldn't overlap. For example, if $a_1 = 2$ and he wants to send string "aaa", he can split it into "a" and "aa" and use 2 magical papers, or into "a", "a" and "a" and use 3 magical papers. He can't split it into "aa" and "aa" because the sum of their lengths is greater than n. He can split the message into single string if it fulfills the conditions.

A substring of string s is a string that consists of some consecutive characters from string s, strings "ab", "abc" and "b" are substrings of string "abc", while strings "acb" and "ac" are not. Any string is a substring of itself.

While Mahmoud was thinking of how to split the message, Ehab told him that there are many ways to split it. After that Mahmoud asked you three questions:

- How many ways are there to split the string into substrings such that every substring fulfills the condition of the magical paper, the sum of their lengths is n and they don't overlap? Compute the answer modulo $10^9 + 7$.
- What is the maximum length of a substring that can appear in some valid splitting?
- What is the minimum number of substrings the message can be spit in?

Two ways are considered different, if the sets of split positions differ. For example, splitting "aa|a" and "a|aa" are considered different splittings of message "aaa".

Input

The first line contains an integer n ($1 \le n \le 10^3$) denoting the length of the message.

The second line contains the message s of length n that consists of lowercase English letters.

The third line contains 26 integers $a_1, a_2, ..., a_{26}$ ($1 \le a_x \le 10^3$) — the maximum lengths of substring each letter can appear in.

Output

Print three lines.

In the first line print the number of ways to split the message into substrings and fulfill the conditions mentioned in the problem modulo $10^9 + 7$.

In the second line print the length of the longest substring over all the ways.

In the third line print the minimum number of substrings over all the ways.

input
3 aab 2 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
output
3 2 2

input
10 abcdeabcde 5 5 5 5 4 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
output
401 4 3

In the first example the three ways to split the message are:

- a|a|b
- aa|b
- a|ab

The longest substrings are "aa" and "ab" of length 2.

The minimum number of substrings is 2 in "a|ab" or "aa|b".

Notice that "aab" is not a possible splitting because the letter 'a' appears in a substring of length 3, while $a_1 = 2$.