BIUPC 2024 2.0 Online Preliminary Contest(Junior)

A. ChatGPT?

1 second, 256 megabytes

Using ChatGPT in programming contests can be unfair, as most contests do not allow outside help, which could lead to disqualification. Relying too much on AI can also stop you from improving your own problem-solving skills, like debugging and creating algorithms, which are important for becoming a better programmer. AI solutions might also be wrong or miss important details, making them unreliable in a contest. Ultimately, using AI takes away the chance to learn and practice on your own, which is the real benefit of participating in these contests.

In this problem, you just need to print a single line given in the sample output.

input
```
No Input
```
output
```
Je ChatGPT Use Korbe Shey Hawn Uncle.
```

B. Oi mama jhogra na pls

1 second, 256 megabytes

Nobita and Gian are two friends who love candies. They went to the store and bought a certain number of candies. They don't want to quarrel, they want to divide all the candies among themselves such that:

- Both Nobita and Gian receive the same number of candies.
- Each of them gets an integer number of candies (no breaking candies allowed).

Given the total number of candies n, determine if it is possible for them to divide the candies equally.

Input

The input consists of a single integer n $(1 \leq n \leq 10^9)$ — the total number of candies they have.

Output

Print "YES" if it is possible to divide the candies equally.

"NO" otherwise.

input
```
4
```
output
```
YES
```

input
```
7
```
output
```
NO
```

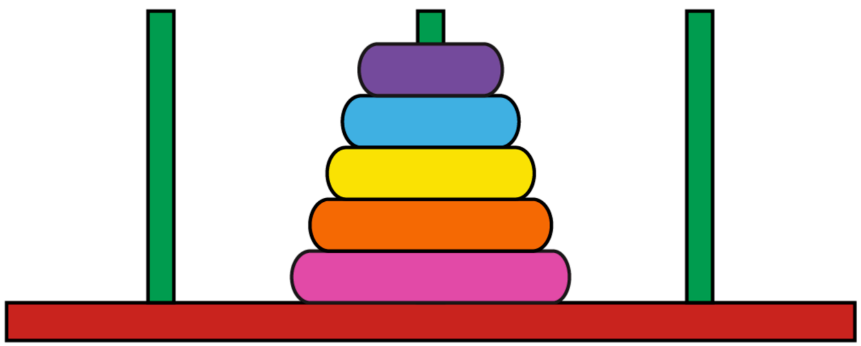
C. Tower of Harun

1 second, 256 megabytes

Harun Shaheb is an avid enthusiast of the Tower of Hanoi puzzle. He has a large collection of towers and rings in his 'Tower Room'. One day he went on a trip with his friends. After returning home, he found that his cats made a mess in his tower room. All the towers and rings were here and there. He can't rearrange everything alone, so he needs your help.

Harun Shaheb will give you n rings where the size of the i-th ring is m_i $(1 \le i \le n)$. You have to receive the rings one by one in their given order and put the rings in any tower which follows one of the conditions:

- the tower is empty
- the top most ring of the tower is not smaller than m_i



Note: There are more towers than the rings in the tower room.

As you are a good competitive programmer, Harun Shaheb asked you the minimum number of towers you need to use to put all the n rings. Can you solve it?

Input

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

The first line of each test case contains a single integer n $(1 \le n \le 5 \cdot 10^5)$ — number of rings.

The next line contains n integers $m_1, m_2, \ldots, m_n (1 \le m_i \le 10^9)$ — size of the rings.

It is guaranteed that the sum of n over all test cases does not exceed $5 \cdot 10^5$.

Output

For each test case, output a single integer — minimum number of towers you need to use to put all the rings.

input

```
2
3
10 11 12
5
8 6 3 6 6
```

output

```
3
2
```

D. Hallway doors

1 second, 256 megabytes

A knight stands at the entrance of a long, narrow hallway containing four doors, each a different color: red, green, blue, and black. The first three doors are positioned one after the other but can be arranged in any order, while the black door is fixed at the end of the hallway where the princess is waiting. To pass through each door, the knight must first open it. To unlock the black door, the knight must first open all the previous doors , which will trigger the automatic opening of the black door.

To open a door, the knight needs the corresponding key: a red key for the red door, a green key for the green door, and a blue key for the blue door. The keys are also scattered throughout the hallway in specific positions.

The knight has a map that shows the exact layout of the hallway as a string of six characters, each representing either a door or a key:

- R, G, B — denoting red, green and blue doors, respectively;
- r, g, b — denoting red, green and blue keys, respectively.

Each character appears exactly once in the string.

The knight starts at the far left of the hallway. Your task is to determine if the knight can collect the keys in the correct order to open all the doors and reach the princess at the end of the hallway.

Input

The first line contains a single integer $t(1 \le t \le 720)$ — the number of testcases.

Each testcase consists of a single string. Each character is one of R, G, B (for the doors), r, g, b (for the keys), and each of them appears exactly once..

Output

For each testcase, print YES if the knight can reach the princess . Otherwise, print NO.

input

```
4
rgbBRG
RgbrBG
bBrRgG
rgRGBb
```

output

```
YES
NO
YES
NO
```

In the first testcase, the knight first collects all keys, then opens all doors with them.

In the second testcase, there is a red door right in front of the knight, but he doesn't have a key for it.

In the third testcase, the key to each door is in front of each respective door, so the knight collects the key and uses it immediately three times.

In the fourth testcase, the knight can't open the blue door.

E. Fight of the Century

1 second, 256 megabytes

Mike Tyson and Jake Paul are engaged in a fight to the death. Each fighter has health points and attack power. During the fight, they deal damage to their opponent every second based on their attack power.

Both fighters also have a special move that doubles their attack power for one second. However, it can only be used again once every c seconds, where c is the cooldown period.

The fight ends when either fighter's health points reach zero or below. The outcome of the fight is determined as follows:

If Mike Tyson's health reaches zero before Jake Paul's, **Jake Paul wins**. If Jake Paul's health reaches zero before Mike Tyson's, **Mike Tyson wins**. If both fighters' health reaches zero in the same second, the result is a **draw**. Your task is to determine the winner if both fighters play optimally.

Input

The input consists of two lines:

The first line contains three integers h_1, b_1, and c_1 $(1 \le h_1, b_1, c_1 \le 10^9)$, representing Mike Tyson's health points, attack power, and special move cooldown, respectively.

The second line contains three integers h_2, b_2, and c_2 $(1 \le h_2, b_2, c_2 \le 10^9)$, representing Jake Paul's health points, attack power, and special move cooldown, respectively.

Output

Print a single line:

"MIKE TYSON" if Mike Tyson wins.

"JAKE PAUL" if Jake Paul wins.

"DRAW" if both lose at the same second.

input

```
20 3 2
30 4 3
```

output

```
JAKE PAUL
```

input

```
10 3 5
10 3 5
```

output

```
DRAW
```

In the first test case, Jake Paul can use his special move at the first and fourth seconds to win the fight.

"Be careful with spacing and spelling."

F. XOR Convolution

1 second, 256 megabytes

The problem is straightforward. You are given an integer n, and your task is to calculate $func(n)$.

The function $func(n)$ is defined as the sum of the results of the function $get(k)$ for all k from 1 to n.

$$func(n) = get(1) + get(2) + get(3) + \cdots + get(n)$$

The function $get(k)$ for any integer k is defined as the bitwise XOR of all integers from 1 to k.

$$get(k) = 1 \oplus 2 \oplus 3 \oplus \cdots \oplus k$$

Where $\oplus$ denotes the bitwise XOR operation

Input

The first line of input contains an integer $t (1 \leq t \leq 2 \cdot 10^5)$, denoting the number of test cases.

The next t lines will contain a single integer $n (1 \leq n \leq 10^9)$.

Output

For each test case output the value of $func(n)$ in a single line.

input

```
1
4
```

output

```
8
```

input

```
2
2
5
```

output

```
4
9
```

G. Painting Wrong!

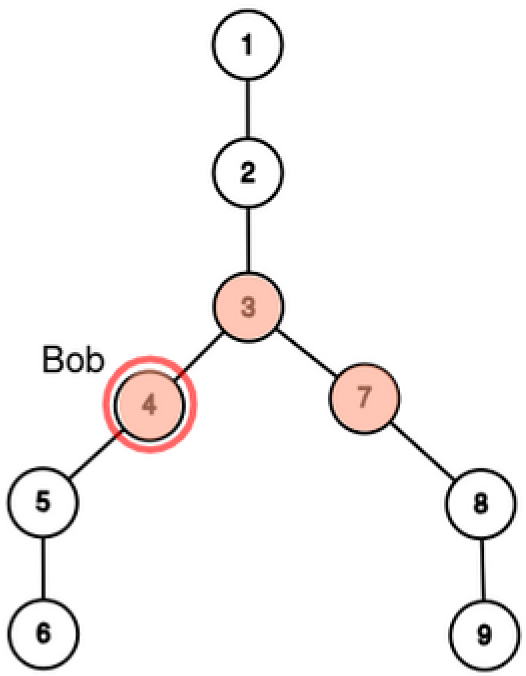
1 second, 256 megabytes

Alice and Bob are here! And guess what? They started playing a game again! :3

There is a tree with n nodes and $n - 1$ bidirectional edges. Initially, all the nodes are White. At the beginning of the game, Alice and Bob are in different nodes. Alice has Blue color paintbrush and Bob has Green color paintbrush. They color their node with their paintbrush. Then in every second, they both do one of the following independently:

- Stay in the current node.
- Move to any White node which shares an edge with their respected color's node. Then color that node with their paint-brush.

For more clarity, Alice can move to any White node that shares an edge with at least one Blue node. Bob can move to any White node that shares an edge with at least one Green node. If both of them move to the same node at a time, they can not color that node. The game ends when none of them can make any move.



In the picture above, node $[3, 4, 7]$ are Red node and Bob is in node 4. Bob can only go to node $[2, 5, 8]$ as they share an edge with Red node.

You are given q queries. In each query, you are given the starting node of Alice. You have to determine if it is possible for Bob to choose his starting node optimally so that after finishing the game, number of Green nodes is greater than Blue nodes.

Input

The first input line contains an integer n $(2 \le n \le 2 \cdot 10^5)$ — the number of nodes. The nodes are numbered $1, 2, \ldots, n$.

Then there are $n - 1$ lines describing the edges. Each line contains two integers a and b: there is an edge between nodes a and b.

Then there is an integer q $(1 \le q \le 2 \cdot 10^5)$— the number of queries.

Then next q lines follows a single integer s_i $(1 \le s_i \le n)$ — starting node of Alice.

Output

For each query, print "YES" if it is possible for Bob to choose his starting node optimally so that after finishing the game, number of Green nodes is greater than Blue nodes.

Otherwise print "NO".

input

```
6
1 2
2 3
2 4
1 5
5 6
2
5
2
```

output

```
YES
NO
```

H. FE!N

1 second, 256 megabytes

A grand concert is being held in the city named HAUKAU Land, attracting massive crowds of people. The crowd, overwhelmed by excitement, is loudly and crazily shouting "FE!N FE!N", causing the land to shake. The intensity of their screams is so powerful that any building located within d distance from the concert area collapses.

You are given an array b of length n, where $b[i]$, represents the distance of the $i'th$ building from the concert area.

Your task is to determine which buildings remain standing after the concert.

Input

The input consists of multiple test cases:

The first line contains an integer $t(1 \leq t \leq 100)$, the number of test cases.

For each test case: The first line contains two integers $n(1 \leq n \leq 10^5)$ and $d(1 \leq d \leq 10^9)$: the number of buildings and the maximum distance their screams can be heard. The second line contains an array b contains n integers $b[1], b[2], \ldots, b[n] (1 \leq b[i] \leq 10^9)$: the distances of the buildings from the concert area.

It is guaranteed that the sum of all n across all test cases does not exceed 10^5.

Output

For each test case:

If all buildings collapse, then print "FE!N".

Otherwise, print:

1. An integer k, the number of buildings that remain standing after the concert.
2. A list of k integers: the $1-based$ indices of the remaining buildings.

input

```
2
5 10
5 15 20 10 25
4 15
10 15 12 14
```

output

```
3
2 3 5
FE!N
```

Test Case 1: Buildings at distances 5 and 10 collapse (≤ 10). 3 buildings remain standing, located at indices 2, 3, and 5 (distances 15, 20, and 25). Therefore, the output is 3 and [2, 3, 5].

Test Case 2: All buildings are within the range of screams (≤ 15), so all collapse. Output is "FE!N"

I. Eagle Vision

1.5 seconds, 256 megabytes

Author :Hamza Miraz (CSE BUBT,intake-48)

In a distant realm, Altar the Guardian, stood atop the tallest tower overlooking an enchanted land divided into a vast grid of **n** rows and **m** columns. Each point on the grid was occupied either by a human or a vampire. The humans lived peacefully, while the vampires, hidden in plain sight, awaited their chance to attack. The problem was that, to the naked eye, there was no way to tell whether a figure was a human or a vampire.

However, Altar had a unique ability called Eagle Vision, which allowed him to pierce the illusion and distinguish the vampires from the humans.

After using Eagle Vision, Altar could uncover the true nature of each grid cell:

- If a grid cell containing 1 represents the presence of a vampire.
- If a grid cell containing 0 represents the presence of a human.

Altar's job was to count how many square submatrices in this grid contained only vampires. He needed to know the total number of such submatrices to assess the strength of the vampire forces.

Your task is to help Altar calculate the number of square submatrices within entire the grid that contain only vampires.

A square submatrix is defined as a subgrid that has equal numbers of rows and columns.

Input

The first line of input contains an integer $t(1 \leq t \leq 1000)$, the number of test cases.

Each test case starts with a line containing two integers n and m $(1 \leq n, m \leq 500)$, the dimensions of the grid.

The next n lines each contain m integers, either 0 or 1, representing the grid. 1 indicates a vampire, while 0 indicates a human.

Output

For each test case, output a single line in the format $Case\ i$: x, where i is the test case number and x is the total number of square submatrices that contain only vampires.

input

```
2
2 2
1 0
0 1
3 4
0 1 1 0
1 1 1 0
0 0 1 0
```

output

```
Case 1: 2
Case 2: 7
```

In the first testcase:

1x1 size square submatrices :

- The cell at (1, 1) contains a vampire represents 1 square submatrix.
- The cell at (2, 2) contains a vampire represents 1 square submatrix.

Total: There are 2 square submatrices of size 1x1 containing only vampires.

In the second testcase:

1x1 size square submatrices :

- The cell at (1, 2) represents 1 square submatrix.
- The cell at (1, 3) represents 1 square submatrix.
- The cell at (2, 1) represents 1 square submatrix.
- The cell at (2, 2) represents 1 square submatrix.
- The cell at (2, 3) represents 1 square submatrix.
- The cell at (3, 3) represents 1 square submatrix.

2x2 size square submatrices :

- The subgrid {(2, 2), (2, 3), (1, 2),(1, 3)} represents 1 square submatrix.

Total: There are 6 square submatrices of size 1x1 and 1 square submatrix of size 2x2 that contain only vampires.

J. Encrypted Message

1 second, 256 megabytes

Bob is a spy tasked with sending a confidential message to his headquarters via email. To ensure the message remains hidden from unauthorized parties, Bob encrypts it using a special technique:

- If the letter in the message is a **vowel** (a, e, i, o, u), it is shifted **one place** forward in the English alphabet.
- If the letter is a **consonant**, it is shifted **two places** forward.
- If the shift moves past the letter z, it starts over from the beginning of the alphabet (e.g., z shifted by 2 becomes b).

Your task is to encrypt Bob's message using this technique.

Input

The input consists of one lines:

- The first line contains a string , consisting only of lowercase English letters, representing the message to be encrypted.

Output

Print the encrypted message as a single string.

input

```
abcdez
```

output

```
bdeffb
```

Shifting means moving the letter to the next 1 or 2 positions in the alphabet. For example, shifting by 1 moves a to b, and shifting by 2 moves a to c. If the shift exceeds z, it starts over from the beginning of the alphabet.(e.g., z shifted by 2 becomes b).
