Problem A. Reversing

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	1024 mebibytes

Consider a rectangular grid of size $N \times M$. Each cell is colored black or white.

If you touch a cell C of the grid, you change the color of all the cells that belong to same-colored connected component of C, including C itself. For connected components, two cells are neighbors if they share a side.

You know the current state of the grid, but you may have touched some cells an arbitrary number of times. Calculate the number of possible initial states of the grid. As the answer may be very large, calculate it modulo 1 000 000 007.

Input

The first line contains two integers N and M, the dimensions of the grid ($1 \le N, M \le 2000$).

Each of the next N lines describes one row of the grid. Each of these lines contains M characters denoting the colors of cells in the row. Each character is either "`B`" for black or "`W`" for white.

Output

Print the number of possible initial states of the grid modulo 1 000 000 007.

Example

standard input	*standard output*
2 2 WW WB	2

Problem B. Lawyers

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	1024 mebibytes

There are N lawyers. Each lawyer has been charged with committing a fraudulent offense. These N lawyers try to defend each other and make sure they are acquitted.

Lawyer A can defend lawyer B if and only if lawyer B trusts lawyer A, and there are M such pairs (A, B). Note that, if lawyer B trusts lawyer A, it does not imply that lawyer A trusts lawyer B.

Each lawyer is very hard-working, so one lawyer can defend any number of others.

Each lawyer is very talented, so anyone who receives at least one defense is unconditionally acquitted. With one exception: if lawyer A defends lawyer B and lawyer B defends lawyer A, it seems very suspicious, and both are found guilty.

Determine whether it is possible or not for all lawyers to be acquitted together.

Input

The first line contains two integers N and M, the number of lawyers and the number of trust relationships ($1 \le N, M \le 200\,000$).

The next M lines describe trust relations. The i-th of these M lines contains two different integers A_i and B_i, which means lawyer B_i trusts lawyer A_i, and so lawyer A_i can defend lawyer B_i. There are no such i and j ($1 \le i, j \le M$, $i \ne j$) that $A_i = B_i$ and $A_j = B_j$.

Output

Print "`YES`" (without quotes) if it is possible for all lawyers to be acquitted together. Print "`NO`" (without quotes) otherwise.

Examples

standard input	*standard output*
3 3 1 2 2 3 3 1	YES
4 6 1 2 1 3 1 4 2 3 2 4 3 4	NO
4 4 1 2 2 1 3 4 4 3	NO

Problem C. Contour

Input file:	standard input
Output file:	standard output
Time limit:	2 second
Memory limit:	512 mebibytes

Mount Snowdon, the highest peak in Wales, draws mountain bikers from all over as a prominent attraction. To capitalize on the growing popularity of this exhilarating sport, a bold new business venture intends to establish multiple bike repair shops across the surrounding undulating foothills.

The cunning small business owner's financial success relates directly to the velocity of the average biker: the faster a biker is going at the foot of the hill the more likely they are to encounter a problem and have to walk — or sometimes limp — into the shop.

Snowdon, like most mountains, has a very angular sort of shape. In fact, the profile of the mountain can be represented as N connected line segments pointing downward at various angles, each starting where the last left off. Given this highly scientific representation of the landscape, we need to find the likely speeds of bikers given that they may start off from the top of any of the N segments.

As we all know, a biker on a θ-degree slope from the vertical will accelerate at a rate of precisely $g \times \cos(\theta)\ \ ms^{-2}$ along the slope.

Input

- One line containing a positive integer N ($1 \leq N \leq 4$), the number of line segments making up the mountain, followed by a space and then a real number g ($1 \leq g \leq 100$), the coefficient of acceleration due to gravity.
- N more lines each containing two integers D_i and then θ_i ($1 \leq D \leq 10^4; 1 \leq \theta \leq 89$): the sloped distance in metres and absolute angle in degrees of this line segment from the vertical respectively. The segments are ordered from the top of the hill to its bottom.

Output

Each of the N lines of output should contain one real number: the velocity of a biker starting at the i^{th}-most segment from the top and finishing at the foot of the mountain.

Answers will be judged as correct if they are printed to within an absolute or relative difference of 10^{-6} from their exact values.

Examples

standard input	standard output
2 9 69 60 100 60	39.0 30.0
3 77 500 65 20 5 1000 80	249.70323 172.65601 163.52926

Problem D. Lonely King

Input file:	*standard input*
Output file:	*standard output*
Time limit:	3 seconds
Memory limit:	1024 mebibytes

You are given a rooted tree with N vertices. Vertex 1 is the root, and each of the other $N-1$ vertices has exactly one incoming edge. There are C_i people living in i-th vertex.

Initially, all edges are blue. You can change a "blue path" into a "red edge". Formally, when there are k blue edges, (a_1, a_2), $(a_2, a_3), \ldots, (a_k, a_{k+1})$, you can replace them with one red edge, (a_1, a_{k+1}). You can execute this operation any number of times.

Because of the COVID-19, your purpose is to prevent contacts between people, so you want to minimize the total number of contacts.

The total number of contacts is the number of pairs of people (A, B) such that A and B live in different vertices and A can visit B via edges (of any color). Note that the edges are directed.

Find the minimum total number of contacts that can be achieved after some (possibly zero) operations on the tree.

Input

The first line contains an integer N, the number of vertices ($1 \le N \le 200\,000$).

The next line contains $N-1$ integers, $P_2, P_3, \ldots, P_N$ ($1 \le P_i \le N$). It means that vertex i has one incoming edge from vertex P_i. These numbers describe a rooted tree with vertex 1 as the root. Keep in mind that the edges are directed.

The next line contains N integers, $C_1, C_2, \ldots, C_N$, which denote the number of people in each vertex ($1 \le C_i \le 10^6$).

Output

Print one integer, the minimum total number of contacts.

Example

standard input	*standard output*
4 1 1 2 2 1 3 2	10

Problem E. Treasure Box

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	1024 mebibytes

You are playing a VR game about adventures in an ancient world. You were on a journey to find the legendary treasure, and after breaking through several gates, you found a box that was supposed to contain the treasure. But the box had a lock on it with the following message: "To open the box, change the letters under the box well, so that it reads the same forward and backward."

After looking under the box, you found a string consisting of N characters in a row. The i-th character is located at position $(i, 0)$, so distances between adjacent characters are 1.

It seems that the box can be opened by replacing the characters under the box and making it a palindrome. To do that, you start at some position, and you can repeatedly move to another position and replace the character at that position by any other character, until the string becomes a palindrome.

Since your HP is limited, you want to gain the treasure using minimum HP. Each position requires a different amount of HP to replace the respective character. Also, it consumes C HP to move a unit distance. That is, if you were at position $(i, 0)$, and you want to move to position $(j, 0)$ to replace the j-th character, the movement will decrease your HP by $C \cdot |j - i|$.

For each integer i such that $1 \le i \le N$, find the minimum HP consumed to obtain the treasure if you start at position $(i, 0)$.

Input

The first line contains an integer T, the number of test cases ($1 \le T \le 100\,000$). The test cases follow.

The first line of each test case contains two integers: N, the number of characters under the box ($1 \le N \le 1\,000\,000$), and C, the amount of HP consumed when moving a unit distance ($1 \le C \le 10^9$).

The second line consists of N characters. It represents the string under the box. Each letter is an uppercase English letter.

The third line contains N integers. The i-th integer represents the HP consumed to replace the i-th character. Each of these integers is between 1 and 10^9.

The sum of N over all test cases does not exceed $1\,000\,000$.

Output

For each test case, print N integers on a single line. The i-th integer must be the minimum HP consumed when starting at position $(i, 0)$.

Example

standard input	*standard output*
2 5 1 ABCDE 7 1 4 5 1 5 1 ABCDA 7 1 4 5 1	6 5 6 6 5 2 1 2 3 4

Note

For the first test case, when the starting position is $(1, 0)$, one of the optimal ways is to first move to position $(2, 0)$ and change "B" to "D", then move to position $(5, 0)$ and change "E" to "A".

Problem F. First Counter

Input file:	standard input
Output file:	standard output
Time limit:	2 second
Memory limit:	512 mebibytes

While driving through the Irish countryside, one may notice small grey stones placed along the roadside approximately every one and a half kilometers. These stones, known as milestones, were originally intended to mark this historic unit of measurement. However, due to their age and attractiveness to birds and other mischievous creatures, not all of these stones remain today.

If you happen to come across some of these worn-out markers while traveling at an unknown constant speed, you may still be able to infer some information from their placements. For instance, if you have counted M remaining stones since you began, what could have been your driving speed?

Input

- One line containing two positive integers, M and N ($2 \le M \le N \le 10^3$): the number of consecutive stones you noticed and the total number of stones along the road respectively.
- One line containing M distinct non-negative integers $T_{1..M}$ in ascending order — the times at which you passed stones in hours ($0 \le T_i \le 10^{15}$).
- One line containing N distinct non-negative integers $X_{1..N}$ in ascending order — the distances along the road of each milestone ($0 \le X_i \le 10^{15}$) in miles.

Output

Output two lines:

- First, the number of distinct possible speeds at which the car could have been travelling.
- Second, a space-separated list of all of the possible distances between the first milestone you saw and the second milestone you saw, in increasing order.

Examples

standard input	standard output
4 12 1 2 4 5 6 8 12 18 26 28 30 34 36 37 39 40	2 1 2
5 10 1 2 3 4 5 0 1 2 3 4 5 6 7 8 9	1 1
3 6 1 2 4 11 12 15 19 24 30	0

Problem G. Make Everything White

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	1024 mebibytes

Consider a rectangular grid of size $N \times M$. Each cell is colored black or white.

For each cell, you have to execute exactly one of the three operations given below.

1. Do not change anything.
2. Change the color of all neighbor cells (black to white, white to black).
3. Change the color of all neighbor cells and itself.

Two cells are neighbors if they share a side.

Your goal is make all cells white. Find a way to achieve it, or determine that it is impossible.

Input

The first line contains two integers N and M, the dimensions of the grid ($1 \le N, M \le 2000$).

Each of the next N lines describes one row of the grid. Each of these lines contains M characters denoting the colors of cells in the row. Each character is either "`B`" for black or "`W`" for white.

Output

Print "`-1`" on the first line if it is impossible to make all cells white.

Otherwise, print "`1`" on the first line, followed by N more lines. Each of these lines must contain M characters which describe the operations you chose for the respective row of the grid. Each of the characters must be either "`1`", "`2`", or "`3`", corresponding to the three operations in the statement. If there are several solutions, print any one of them.

Examples

standard input	*standard output*
2 3 WBW BWB	1 111 121
1 1 B	1 3

Problem H. Hungover

Input file:	standard input
Output file:	standard output
Time limit:	2 second
Memory limit:	512 mebibytes

The University of Lagado is planning events for the upcoming Fresher's week and has received an unexpected request from some undergraduates who are interested in a beer tasting. To ensure the safety of the students, the university has decided to enforce a strict limit on the amount of alcohol consumption by each student. The drinks will be served in varying amounts of a liter, half a liter, or a third of a liter, each containing different strengths of alcohol. To limit possible intoxication, the university has established that 1% of a liter at 100% strength equals one unit of alcohol. While the students are excited about the event, they want to make the most of their money and remain sober for the upcoming morning lectures. Therefore, they need to determine how to spend their money wisely and consume their self-imposed alcohol limit for the night without getting too drunk.

Input

- One line containing three numbers:
  - m ($0.00 \leq m \leq 10.00$), the amount of money they can spend to two decimal places;
  - u ($0.0 \leq u \leq 20.0$), the number of units they aim to drink to one decimal place;
  - d ($1 \leq d \leq 8$), the number of different drinks available.
- Another d lines, each containing:
  - up to 20 lowercase latin letters (the name of the drink);
  - an integer between 0 and 100 (its strength as a percentage);
  - its size (either '1/1' for a litre, '1/2' for a half or '1/3' for a third);
  - a real number to two decimal places; its cost.

Name, strength, price and cost will be separated by spaces.

Output

If the students' aims are possible, write one or more lines, each listing the name of the drink purchased followed by the positive integer count of how many drinks of that type were bought. If there are several solutions, output any.

Otherwise, output a single line containing the word **IMPOSSIBLE**.

Examples

standard input	standard output
10.00 9.0 2 fire 2 1/1 4.00 water 10 1/2 2.00	fire 2 water 1
2.00 3.0 3 firewater 1 1/1 1.00 windwater 1 1/1 1.00 earthwater 1 1/1 1.00	IMPOSSIBLE

Problem I. Visiting Friend

Input file:	*standard input*
Output file:	*standard output*
Time limit:	3 seconds
Memory limit:	1024 mebibytes

The village where Sunghyeon lives consists of N intersections and M two-way roads. The intersections are numbered by integers from 1 to N.

Each two-way road connects two different intersections, there is a path along the roads between any two intersections. Also, for any two intersections, there is at most one road directly connecting them.

On Sunday, Sunghyeon left her house and went to play at the house of her friend Changki, who lives in the same village. Since Sunghyeon has just moved in, she doesn't know anything about the village's road network, and she can only distinguish her house and Changki's house: all other houses look the same to her. After wandering for a long time, she finally arrived at Changki's house.

Sunghyeon remembers that, after departure, she did not visit the intersection where her house was located anymore, and she immediately entered Changki's house when she reached the intersection where Changki's house was located.

Sunghyeon wondered how many intersections she might have visited when she went to play at Changki's house. In other words, out of N intersections, she wants to count such intersections V that there exists a possible path for her which visited V.

Curious Sunghyeon went a step further here, and she wondered how many different intersections she might have visited if her house was at intersection A and Changki's house was at intersection B. She wants the answer for many different pairs (A, B).

Write a program to help Sunghyeon and Changki.

Input

The first line contains an integer T, the number of test cases ($1 \le T \le 1000$). The test cases follow.

The first line of each test case contains two integers N and M, the number of intersections and the number of roads ($2 \le N \le 200\,000$, $1 \le M \le 500\,000$).

Each of the next M lines contains two integers U_i and V_i which mean there is a two-way road between these intersections ($1 \le U_i, V_i \le N$, $U_i \ne V_i$). There is a path along the roads between any two intersections. Also, for any two intersections, there is at most one road directly connecting them.

The next line contains an integer Q, the number of questions ($1 \le Q \le 500\,000$).

Each of the next Q lines contains two integers A_j and B_j which mean you have to find the number of different intersections Sunghyeon might have visited if her house was at intersection A_i and Changki's house was at intersection B_i ($1 \le A_j, B_j \le N$, $A_j \ne B_j$).

The sum of N over all test cases does not exceed 200 000. The sum of M over all test cases does not exceed 500 000. The sum of Q over all test cases does not exceed 500 000.

Output

For each test case, print Q lines. The i-th of these Q lines must contain the number of intersections Sunghyeon might have visited if her house was at intersection A_i and Changki's house was at intersection B_i.

Example

standard input	*standard output*
1 5 5 1 2 1 3 2 4 4 5 2 5 5 1 2 1 4 2 3 2 5 3 5	2 4 3 3 5

Problem J. Cooperation Game

Input file:	*standard input*
Output file:	*standard output*
Time limit:	3 seconds
Memory limit:	1024 mebibytes

Athletic contests in schools sometimes get too heated. To promote cooperation, the teachers have devised a game where cooperation leads to better scores.

This cooperation game starts with N students standing in a line. To get a score, two students from the same class will get out of the line. If they were i-th and j-th students along the line, just before they went out, they add $|i - j|$ to the total score. The empty places of the two students that just came out will be compacted. The game stops when there are no more pairs of students from the same class.

For example, consider six students initially standing in a line:

$$1, 2, 3, 3, 2, 1$$

The numbers are class numbers of the students. If the students come out from the line in the order of class number 1, then class number 2, then class number 3, the total score is $5 + 3 + 1 = 9$. However, in the same initial situation, if the students come out from the line in the order of 3, 2, 1, then the total score is $1 + 1 + 1 = 3$.

Given the class numbers in the initial line, write a program to calculate the maximum possible score.

Input

The first line contains an integer T, the number of test cases ($1 \le T \le 48$). The test cases follow.

The first line of each test case contains a single integer N, the number of students ($1 \le N \le 300\,000$).

In the next line, the sequence of class numbers are given. The class numbers are integers from 1 to N.

The sum of N over all test cases does not exceed $7\,000\,000$.

Output

For each test case, print a line with a single integer: the maximum possible score.

Example

standard input	*standard output*
2 7 1 2 1 1 2 1 2 12 1 2 3 1 2 3 1 2 3 1 2 3	10 30

Problem K. KeyWord Log

Input file:	standard input
Output file:	standard output
Time limit:	2 second
Memory limit:	512 mebibytes

Popular social networking site Key+Word is struggling to manage its many popular forums. Recent regulation requires the site to report users engaged in conversations about certain topics. The sheer number of users means manual monitoring is too costly and so the site has asked its many interns to come up with a solution.

One intern has theorized that conversations about any given topic will see the same key words used over and over. If the most used words can be identified perhaps manual investigation can be directed towards appropriate forums.

Input

- One line containing a single integer M ($1 \leq M \leq 10^4$), the number of messages.
- M more lines each beginning with a user's name of no more than 20 characters and continuing with the content of that user's message all in lower case. The total number of characters across all messages, including spaces, will not exceed 2×10^6.

Output

Several words, one per line, listing the words used by every single user on the forum, ordered from most to least used and in case of a tie in alphabetical order. If there are no such words, output **ALL CLEAR**.

Examples

standard input	standard output
8 Jepson no no no no nobody never Ashley why ever not Marcus no not never nobody Bazza no never know nobody Hatty why no nobody Hatty nobody never know why nobody Jepson never no nobody Ashley never never nobody no	no nobody never
2 Villain avast Scoundrel ahoy	ALL CLEAR