Problem A. Scoring Board

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

In programming contests, the scoring board shows each team's judge results. Assume there is only one problem in the contest, the scoring board will show according to the rules below:

- When there is no submissions, the scoring board will show "`-`".
- When there is no "`WA`" before "`AC`", the scoring board will show "`+`".
- When the team failed to solve the problem, the scoring board will show "`-k`", where k is the number of "`WA`".
- When the team passed the problem with several tries, the scoring board will show "`+k`", where k is the number of "`WA`" before "`AC`".

Please write a program to show the scoring board.

Input

The first line of the input contains an integer $T(1 \leq T \leq 50)$, denoting the number of test cases.

In each test case, there is one integer $n(0 \leq n \leq 50)$ in the first line, denoting the number of submissions.

In the second line, there are n strings $s_1, s_2, \ldots, s_n(s_i \in \{$"`AC`", "`WA`"$\})$, denoting the judge results of each submission. The submissions have already been sorted by time.

Output

For each test case, print a single line, denoting what the scoring board shows.

Example

standard input	standard output
5 4 WA WA WA AC 5 WA WA WA AC WA 1 AC 2 WA WA 0	+3 +3 + -2 -

Problem B. Boxes

Input file:	standard input
Output file:	standard output
Time limit:	6 seconds
Memory limit:	512 megabytes

There are n boxes in 3D-space, labeled by $1, 2, \ldots, n$. They are all in same size with length d_x, width d_y and height d_z. The sides of the boxes are all paralleled to the axes. For the i-th box, one of its corners' coordinate is (x_i, y_i, z_i), and the corner opposite to it is at $(x_i + d_x, y_i + d_y, z_i + d_z)$.

For two different boxes i and j, they share common points if $|x_i - x_j| \le d_x, |y_i - y_j| \le d_y$ and $|z_i - z_j| \le d_z$.

Your task is to determine whether there exists two different boxes i and $j (i \ne j)$ that they share common points.

Input

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

In each test case, there are four integers $n, d_x, d_y, d_z (2 \le n \le 100000, 1 \le d_x, d_y, d_z \le 10^9)$ in the first line, denoting the number of boxes and the size of each box.

For the next n lines, each line contains three integers $x_i, y_i, z_i (1 \le x_i, y_i, z_i \le 10^9)$, denoting the coordinate of each box.

It is guaranteed that $\sum n \le 10^6$.

Output

For each test case, print a single line. If there exists two different boxes i and $j (i \ne j)$ that they share common points, output "Yes", otherwise output "No".

Example

standard input	standard output
2 2 2 3 4 1 1 1 3 4 5 2 2 3 4 1 1 1 3 4 6	Yes No

Problem C. Catering

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

Mr. Bread owns a catering company and business is booming. The company is planning to open some new restaurants. There are n possible locations to open restaurant, the i-th location will hire a_i staff, and will need b_i staff when things are busy.

The company wants to choose as many as possible locations to open new restaurants. The only constrait is that every new restaurant should have enough staff in busy. Fortunately, there will be at most one restaurant in busy each day, so a plan is valid if $\sum a_i \geq \max(b_i)$.

Please write a program to determine how many locations can be choosen.

Input

The first line of the input contains an integer $T(1 \leq T \leq 10000)$, denoting the number of test cases.

In each test case, there is one integer $n(1 \leq n \leq 100000)$ in the first line, denoting the number of possible locations.

For the next n lines, each line contains two integers $a_i, b_i(1 \leq a_i, b_i \leq 10^9)$, describing each location.

It is guaranteed that $\sum n \leq 10^6$.

Output

For each test case, print a single line containing an integer, denoting the maximum number of choosen locations.

Example

standard input	standard output
2 2 3 4 2 5 2 3 4 2 6	2 0

Problem D. Distance

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	512 megabytes

There are n holes and $n-1$ tunnels connecting them. The holes are labeled by $1, 2, \ldots, n$. For all $i > 1$, a hole number i is connected by a tunnel to the hole number $\lfloor \frac{i}{2} \rfloor$. Tunnels are bidirectional and each tunnel's length is 1.

There will be m events happened in the tunnel system. There are two kinds of events:

- `- u`, remove the tunnel between u and $\lfloor \frac{u}{2} \rfloor$. It is guaranteed that each tunnel will be removed at most once.
- `? k`, count the number of pairs $i, j (1 \le i < j \le n)$ that holes i, j are still connected and the shortest distance between i and j is not larger than k.

Please write a program to support these events efficiently.

Input

The first line of the input contains an integer $T (1 \le T \le 10)$, denoting the number of test cases.

In each test case, there are two integers $n, m (2 \le n \le 100000, 1 \le m \le 100000)$ in the first line, denoting the number of holes and events.

For the next m lines, each line describes an event, and it is guaranteed that $2 \le u \le n$ and $1 \le k \le 40$.

Output

For each count event, print a single line containing an integer, denoting the answer.

Example

standard input	standard output
1 5 5 ? 2 - 2 ? 2 - 5 ? 2	8 4 2

Problem E. Toy Cars

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

Hamster is a little boy - he is only three years old and enjoys playing with toy cars very much. Hamster has n different cars, labeled by $1, 2, \ldots, n$. They are kept on a long straight trail from left to right. The i-th car occupies the interval $[l_i, r_i]$ of the trail. Two adjacent cars can touch, but can't overlap, so $l_i \geq r_{i-1}$ always holds for all $i = 2, 3, \ldots, n$.

You are a guest visiting Hamster's house. Hamster invites you to play toy cars together. He will give you m commands. There are three kinds of commands:

- `L x k`, move the x-th toy car to the left by k. That is, assume the x-th car is at $[l_x, r_x]$, move it to $[l_x - k, r_x - k]$. Negative coordinates are allowed.
- `R x k`, move the x-th toy car to the right by k. That is, assume the x-th car is at $[l_x, r_x]$, move it to $[l_x + k, r_x + k]$.
- `? x`, you should tell Hamster l_x.

Note that the car being moved might "bump into" another car. The moving car can push other cars if touched.

Please write a program to help Hamster play toy cars.

Input

The first line of the input contains an integer $T (1 \leq T \leq 10)$, denoting the number of test cases.

In each test case, there are two integers $n, m (1 \leq n, m \leq 2000)$ in the first line, denoting the number of toy cars and commands.

For the next n lines, each line contains two integers $l_i, r_i (1 \leq l_i \leq r_i \leq 10^8)$, describing each toy car. It is guaranteed that $l_i \geq r_{i-1}$ for all $i = 2, 3, \ldots, n$.

For the next m lines, each line describes an event, and it is guaranteed that $1 \leq x \leq n$ and $1 \leq k \leq 10^5$.

Output

For each query command, print a single line containing an integer, denoting the answer.

Example

standard input	standard output
1 4 5 1 2 4 6 6 9 9 10 L 2 2 ? 2 L 2 2 ? 2 ? 1	2 0 -1

Problem F. Factorization

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

You are given a positive integer $n (n > 1)$. Consider all the different prime divisors of n. Each of them is included in the expansion n into prime factors in some degree.

Assume $n = p_1^{e_1} p_2^{e_2} \ldots p_m^{e_m}$, where p_i are different prime divisors of n and $e_i \geq 1$. Your task is to find the greatest common divisor of $e_1, e_2, \ldots, e_m$.

Input

The first line of the input contains an integer $T (1 \leq T \leq 10000)$, denoting the number of test cases.

In each test case, there is only one integer $n (2 \leq n \leq 10^{18})$.

Output

For each test case, print a single line containing an integer, denoting the answer.

Example

standard input	standard output
3 4 6 36	2 1 2

Problem G. Guess the Number

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	512 megabytes

Professor Elephant has n numbers $f_1, f_2, \ldots, f_n$ but he doesn't want to tell you the numbers directly. Instead, you should give him some money to buy some useful imformation.

You can buy two types of information:

- `? x`, ask the value of f_x, each information will cost you c_x dollars.
- `? a[i] b[i]`, ask the value of $f_{a[i]} - f_{b[i]}$, each information will cost you $w[i]$ dollars.

You can buy information for many times. Please pay as few as possible dollars to make sure that you can know all the n numbers.

Input

The first line of the input contains an integer $T (1 \leq T \leq 10)$, denoting the number of test cases.

In each test case, there are two integers $n, m (1 \leq n \leq 100000, 0 \leq m \leq 200000)$ in the first line, where m denotes the number of information in the form of `? a[i] b[i]`.

In the second line, there are n integers $c_1, c_2, \ldots, c_n (1 \leq c_i \leq 10^9)$, denoting the cost of type 1.

For the next m lines, each line contains three integers $a[i], b[i], w[i] (1 \leq a[i] < b[i] \leq n, 1 \leq w[i] \leq 10^9)$, describing each information of type 2.

Output

For each test case, print a single line containing an integer, denoting the minimum cost.

Example

standard input	standard output
1 2 1 5 7 1 2 4	9

Problem H. Treasure Hunting

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	512 megabytes

You've got a treasure map. The map shows there are n pirate chests buried in Byteland, labeled by $1, 2, \ldots, n$. The i-th chest's location is (i, p_i), and it contains v_i golden coins.

You are at $(0, 0)$, every time you can move from (x, y) to $(x + 1, y)$ or $(x, y + 1)$. When you touch any chest, you will get all the golden coins inside it.

Unfortunately, the map is not perfect - some chests described on the map do not exist! There exists two integers $l, r(1 \leq l \leq r \leq n)$ that the chests labeled in $[l, r]$ are fake while other chests are real.

You don't know l or r, but you know $r - l + 1 = k$. Please write a program to determine how many coins you can get in all possible situations.

Input

The first line of the input contains an integer $T(1 \leq T \leq 10000)$, denoting the number of test cases.

In each test case, there are two integers $n, k(1 \leq k < n \leq 100000)$ in the first line, denoting the number of pirate chests and the value of $r - l + 1$.

For the next n lines, each line contains two integers $p_i, v_i(1 \leq p_i \leq n, 1 \leq v_i \leq 10^9)$, describing each chest.

It is guaranteed that $\sum n \leq 10^6$.

Output

For each test case, print $n - k + 1$ lines. The i-th line contains an integer, denoting the maximum number of coins you can get when chests labeled in $[i, i + k - 1]$ are fake.

Example

standard input	standard output
1 5 2 1 2 1 3 2 1 1 4 3 5	9 11 10 6

Problem I. Permutation

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

In mathematics, a permutation of n is a set of integers $p_1, p_2, \ldots, p_n$ where $1 \le p_i \le n$ and $p_i \ne p_j$ for all pairs of $(i,j)(i \ne j)$.

It is not hard to figure out that the number of permutations of n is $n!$. In this task, you are given n integers $a_1, a_2, \ldots, a_n$, please figure out how many permutations of n are good.

A permutation is good if and only if $p_i \le a_i$ for all $i \in [1, n]$.

Input

The first line of the input contains an integer $T(1 \le T \le 10000)$, denoting the number of test cases.

In each test case, there is one integer $n(1 \le n \le 100000)$ in the first line.

In the second line, there are n integers $a_1, a_2, ..., a_n(1 \le a_i \le n)$.

It is guaranteed that $\sum n \le 10^6$.

Output

For each test case, print a single line containing an integer, denoting the number of good permutations. As the answer can be very large, output it modulo $10^9 + 7$.

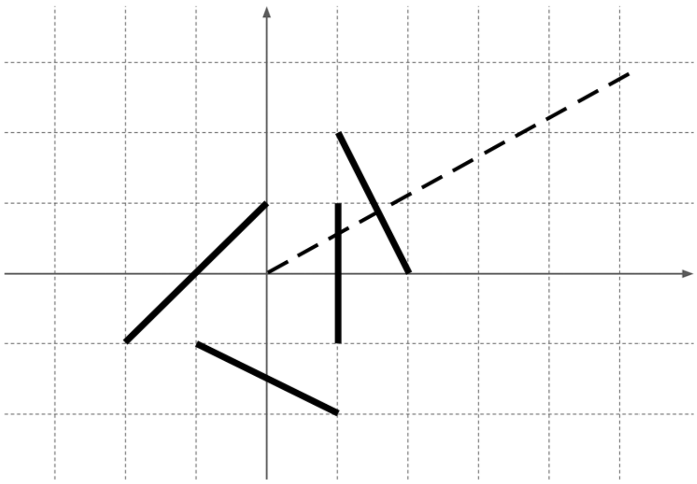
Example

standard input	standard output
2 3 3 3 3 3 1 3 3	6 2

Problem J. Shooting Game

Input file:	`standard input`
Output file:	`standard output`
Time limit:	6 seconds
Memory limit:	512 megabytes

Meow is a gifted shooter. One day, Meow is playing a shooting game. In this game, there are n targets standing on the ground, labeled by $1, 2, \ldots, n$. Each target can be regarded as a segment in 2D coordinate system. Meow is at $(0, 0)$, he can't move to another place. What he should do is to choose a best firing angle, and then shoots a bullet from his gun. The bullet will fly along the chosen angle, it may pass through some targets. The score of the game is the sum of all the targets' scores passed through by the bullet. Note that if the bullet only touches the endpoint of the target, the target's score is also counted.



Meow always chooses the angle that will maximize the final score. His goal is to get at least m score, which may be very easy for him to achieve. He can adjust the difficulty of the game by setting the parameter $k(1 \le k \le n)$ - the game will only show the first k targets i.e. the 1-th, 2-th, $\ldots$, k-th target.

Please write a program to help him find the minimum value of parameter k that he can still reach at least m score, or determine it is impossible to achieve the goal.

Input

The first line of the input contains an integer $T(1 \le T \le 10000)$, denoting the number of test cases.

In each test case, there are two integers $n, m(1 \le n \le 100000, 1 \le m \le 10^{18})$ in the first line, denoting the number of targets and the goal.

For the next n lines, each line contains five integers $xa_i, ya_i, xb_i, yb_i, w_i(|xa_i|, |ya_i|, |xb_i|, |yb_i| \le 10^9$, $1 \le w_i \le 10^9)$. It means the endpoints of the i-th target are (xa_i, ya_i) and (xb_i, yb_i), the score of it is w_i. (xa_i, ya_i) will never be coincided with (xb_i, yb_i). No target will touch $(0, 0)$, even $(xa_i, ya_i), (xb_i, yb_i), (0, 0)$ are not collinear.

It is guaranteed that $\sum n \le 10^6$.

Output

For each test case, print a single line containing an integer, denoting the minimum value of k. If there is no solution, please output "`-1`" instead.

Example

standard input	standard output
1 4 7 1 -1 1 1 3 1 2 2 0 5 -1 -1 1 -2 1 0 1 -2 -1 2	2

Problem K. Math Expression

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	512 megabytes

Given a sequence of integers $a_1, a_2, \ldots, a_n$. You should insert exactly one operation between a_i and a_{i+1} for all $i \in [1, n-1]$. The only operations you can choose to insert are "$+$" and "$\times$", so there are 2^{n-1} ways in total.

Your task is to count how many possible ways of insertion that the value of the final math expression is a multiple of integer k.

For example, assume the sequence is $2, 1, 2$ and $k = 2$, there are 4 possible ways of insertion:

- $2 + 1 + 2 = 5$, is not a multiple of 2.
- $2 + 1 \times 2 = 4$, is a multiple of 2.
- $2 \times 1 + 2 = 4$, is a multiple of 2.
- $2 \times 1 \times 2 = 4$, is a multiple of 2.

Input

The first line of the input contains an integer $T (1 \le T \le 10)$, denoting the number of test cases.

In each test case, there are two integers $n, k (2 \le n, k \le 300)$ in the first line.

In the second line, there are n integers $a_1, a_2, \ldots, a_n (1 \le a_i \le 10^9)$, denoting the sequence.

Output

For each test case, print a single line containing an integer, denoting the number of possible ways of insertion. As the answer can be very large, output it modulo $10^9 + 7$.

Example

<table>
<tr><th>standard input</th><th>standard output</th></tr>
<tr><td>1
3 2
2 1 2</td><td>3</td></tr>
</table>

Problem L. Map Tiles

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

You are a developer of a 2D computer game. Your job is to draw maps in the game. The designer has already designed the map of the game, so what you need to do is to cut the picture into very small square tiles.

Assume the size of the picture is $w \times h$, the picture can be saved as a matrix $g[0 \ldots w-1, 0 \ldots h-1]$. You need to find the smallest integer k that $g[i,j] = g[i \bmod k, j \bmod k]$ always holds for all pairs of $(i,j)(0 \le i < w, 0 \le j < h)$.

Input

The first line of the input contains an integer $T(1 \le T \le 10)$, denoting the number of test cases.

In each test case, there are two integers $w, h(1 \le w, h \le 200)$ in the first line.

For the next w lines, each line contains a string of length h, which only contains lower-case letters, denoting the picture.

Output

For each test case, print a single line containing an integer, denoting the smallest value of k.

Example

standard input	standard output
3 3 6 ababab cdcdcd ababab 2 2 ab cd 3 3 aaa aaa aaa	2 2 1