Problem A. Anti-AK Problem

Input file:	`standard input`
Output file:	`standard output`
Time limit:	20 seconds
Memory limit:	512 mebibytes

Zhang3 and Aunt are playing with tomatoes. They throw tomatoes onto a sphere. The center of the sphere is at $(0, 0, 0)$, and the radius is R. Each time a tomato is thrown, it will burst on the surface of the sphere, then the tomato juice will cover a certain circle on the surface of the sphere, which is called a spherical circle for short. A spherical circle can be represented as (x, y, z, r), where (x, y, z) is the center, a point on the surface, and r is the radius, which is the spherical distance between center and border of the circle. Those points on the surface whose spherical distance to center (x, y, z) is not greater than r are in the circle. (As you know, the spherical distance between two points on the surface is the shortest distance to travel from one to another, passing through only points on the surface of the sphere.)

Zhang3 has already thrown n tomatoes, the i^{th} of which covered a spherical circle (x_i, y_i, z_i, r_i) with tomato juice.

Aunt decides to throw an extra tomato. She will randomly choose a point on the surface of the sphere, called (x, y, z), and then let tomato juice cover a spherical circle (x, y, z, r_0), where r_0 is given.

Zhang3 wants to know the expected area which is covered by tomato juice at least once. Please calculate and print the value.

Input

The first line of the input gives the number of test cases, T $(1 \le T \le 10)$. T test cases follow.

For each test case, the first line contains three numbers n, R, r_0 $(1 \le n \le 5000, 1 \le R \le 10)$, representing the number of tomatoes thrown by Zhang3, the radius of the sphere, and the radius of the random spherical circle, respectively.

Then n lines follow, the i^{th} of which contains four real numbers x_i, y_i, z_i, r_i, representing the i^{th} spherical circle.

The sum of n in all test cases doesn't exceed 10000.

For every real number in the input, the decimal part is not longer than 25 digits. The center of each spherical circle, i.e. (x_i, y_i, z_i), has a distance of at most 10^{-20} to the surface of the sphere. Formally, $|(x_i, y_i, z_i)| \in [R - 10^{-20}, R + 10^{-20}]$.

No spherical circle can be as large as half the sphere. Formally, $0.001\pi R \le r_0, r_i \le 0.499\pi R$.

Output

For each test case, print a line with a real number, representing the answer.

Your answers should have absolute or relative errors of at most 10^{-9}.

Example

standard input	standard output

`1`
`2 1 0.1`
`0 0 1 1.5`
`0 1 0 1.4`
	`8.65419601623420079589`

Problem B. Blow up the Enemy

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 seconds
Memory limit:	512 mebibytes

Zhang3 is playing a shooting game with Father. In the game there are two players trying to kill each other to win the game.

The game provides n weapons, each has two properties: Damage and Delay. The i^{th} weapon has Damage A_i and Delay D_i. When a player shoots with this weapon, his enemy's HP is reduced by A_i, then he must wait for D_i ms before he can shoot again.

The game processes as follows:

1. Before the game starts, Zhang3 and Father choose a weapon respectively. Father always randomly chooses one of the n weapons with equal probabilities. Each player can only use the chosen weapon during the game.
2. When the game starts, Zhang3 and Father have 100 HP each. They make their first shot at the same time.
3. They keep shooting as quickly as possible. That means, a player shoots instantly whenever he can shoot, until the game ends.
4. When a player's HP is reduced to 0 or lower, he dies and the game ends. If the other player is still alive (i.e. has HP higher than 0), then the living player wins the game; otherwise (if the two players die at the same time), each player has 50% probability to win the game.

Zhang3 wants to win the game. Please help her to choose a weapon so that the probability to win is maximized. Print the optimal probability.

Input

The first line of the input gives the number of test cases, T $(1 \le T \le 100)$. T test cases follow.

For each test case, the first line contains an integer n $(1 \le n \le 1000)$, the number of weapons in the game.

Then n lines follow, the i^{th} of which contains two integers A_i, D_i $(1 \le A_i \le 100,\ 1 \le D_i \le 10000)$, representing the Damage and the Delay of each weapon.

The sum of n in all test cases doesn't exceed 2000.

Output

For each test case, print a line with a real number p $(0 \le p \le 1)$, representing the optimal probability.

Your answers should have absolute or relative errors of at most 10^{-6}.

Example

standard input	standard output
2 1 100 100 4 50 50 40 20 30 10 20 100	0.5 0.875

Problem C. Contest of Rope Pulling

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	512 mebibytes

Rope Pulling, also known as Tug of War, is a classic game. Zhang3 organized a rope pulling contest between Class 1 and Class 2.

There are n students in Class 1 and m students in Class 2. The i^{th} student has strength w_i and beauty-value v_i. Zhang3 needs to choose some students from both classes, and let those chosen from Class 1 compete against those chosen from Class 2. It is also allowed to choose no students from a class or to choose all of them.

To be a fair contest, the total strength of both teams must be equal. To make the contest more beautiful, Zhang3 wants to choose such a set of students, that the total beauty-value of all participants is maximized. Please help her determine the optimal total beauty-value.

Input

The first line of the input gives the number of test cases, T $(1 \le T \le 30)$. T test cases follow.

For each test case, the first line contains two integers n, m $(1 \le n, m \le 1000)$, representing the number of students in Class 1 and Class 2.

Then $(n + m)$ lines follow, describing the students. The i^{th} line contains two integers w_i, v_i $(1 \le w_i \le 1000,\ -10^9 \le v_i \le 10^9)$, representing the strength and the beauty-value of the i^{th} student. The first n students come from Class 1, while the other m students come from Class 2.

The sum of $(n + m)$ in all test cases doesn't exceed 10^4.

Output

For each test case, print a line with an integer, representing the optimal total beauty-value.

Example

standard input	standard output
2 3 4 4 7 3 8 2 2 1 4 5 8 1 3 4 4 1 2 1000 -10000 200 3000 800 5000	30 0

Problem D. Deliver the Cake

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

It is Zhang3's birthday! Zhang3 has bought a birthday cake and now it's time to take it home.

There are n villages, labeled $1, 2, \ldots, n$. There are m bidirectional roads, the i^{th} of which connects village a_i, b_i and it is d_i meter(s) long.

The bakery locates at village s and Zhang3's home locates at village t. So Zhang3 wants to carry the cake from s to t. She can carry the cake either with her left hand or with her right hand. She can switch to the other hand during the trip, which takes extra x second(s) each time (when she's performing this action, she must stay in her place). Switching is allowed at any place, including the middle of the roads. She can do this as many times as she like, or don't do it at all.

Some villages are LEFT. When Zhang3 is at a LEFT village, she must carry the cake with her left hand at the moment. In the same way, some other villages are RIGHT, she must carry with her right hand when she's at these villages. The rest villages are called MIDDLE. There's no special rules at MIDDLE villages.

Zhang3 can start and finish with any hand carrying the cake. However, if s or t is not MIDDLE, their special rules must be followed.

Please help Zhang3 find a way to take the cake home, with the minimum amount of spent time.

Input

The first line of the input gives the number of test cases, T $(1 \le T \le 100)$. T test cases follow.

For each test case, the first line contains five integers n, m, s, t, x $(1 \le n \le 10^5,\ 1 \le m \le 2 \times 10^5,\ 1 \le x \le 10^9)$, representing the number of villages, the number of roads, the bakery's location, home's location, and the time spent for each switching.

The next line contains a string of length n, describing the type of each village. The i^{th} character is either `L` representing village i is LEFT, or `M` representing MIDDLE, or `R` representing RIGHT.

Finally, m lines follow, the i^{th} of which contains three integers a_i, b_i, d_i $(1 \le d_i \le 10^9)$, denoting a road connecting village a_i and b_i of length d_i.

It is guaranteed that t can be reached from s.

The sum of n in all test cases doesn't exceed 2×10^5. The sum of m doesn't exceed 4×10^5.

Output

For each test case, print a line with an integer, representing the minimum amount of spent time (in seconds).

Example

standard input	standard output
1 3 3 1 3 100 LRM 1 2 10 2 3 10 1 3 100	100

Problem E. Equal Sentences

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 seconds
Memory limit:	512 mebibytes

Sometimes, changing the order of the words in a sentence doesn't influence understanding. For example, if we change '`what time is it`', into '`what time it is`'; or change '`orz zhang three ak world final`', into '`zhang orz three world ak final`', the meaning of the whole sentence doesn't change a lot, and most people can also understand the changed sentences well.

Formally, we define a sentence as a sequence of words. Two sentences S and T are almost-equal if the two conditions holds:

1. The multiset of the words in S is the same as the multiset of the words in T.
2. For a word α, its i^{th} occurrence in S and its i^{th} occurrence in T have indexes differing no more than 1. (The k^{th} word in the sentence has index k.) This holds for all α and i, as long as the word α appears at least i times in both sentences.

Please notice that almost-equal is not a equivalence relation, unlike its name. That is, if sentences A and B are almost-equal, B and C are almost-equal, it is possible that A and C are not almost-equal.

Zhang3 has a sentence S consisting of n words. She wants to know how many different sentences there are, which are almost-equal to S, including S itself. Two sentences are considered different, if and only if there is a number i such that the i^{th} word in the two sentences are different. As the answer can be very large, please help her calculate the answer modulo $10^9 + 7$.

Input

The first line of the input gives the number of test cases, T ($1 \le T \le 100$). T test cases follow.

For each test case, the first line contains an integer n ($1 \le n \le 10^5$), the number of words in the sentence.

The second line contains the sentence S consisting of n words separated by spaces. Each word consists of no more than 10 lowercase English letters.

The sum of n in all test cases doesn't exceed 2×10^5.

Output

For each test case, print a line with an integer, representing the answer, modulo $10^9 + 7$.

Example

standard input	standard output
2 6 he he zhou is watching you 13 yi yi si wu yi si yi jiu yi jiu ba yao ling	8 233

Problem F. Fake Photo

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	512 mebibytes

Zhang3 has made a fake photo by Photoshop, and published it on the Internet.

However, there are n watches in the photo, showing different times. We assume that every watch has two hands: the hour hand and the minute hand, both moves at a uniform speed.

Someone pointed out the problem of the watches. Then everyone began to doubt that it's a fake photo. To respond, Zhang3 will announce that the photo is taken at time x, which is a time in a day. By choosing x perfectly, she can make all of the watches look like showing time x, with a small error. For example, if there are two watches showing `12:30:00` and `02:40:00` respectively, she can choose `01:35:00` as if both watches are showing time x.

Specifically, we define the error of a hand of a watch as the angle between its actual location and its ideal location, where the ideal location is the correct location to show time x. In the example above, the first watch shows `12:30:00` and $x =$ `01:35:00`, so the minute hand has an error of $30°$ while the error of the hour hand is a little bit larger.

Please help Zhang3 choose such x that the maximum error among all of the $2n$ hands is minimized. x doesn't need to be an integer in seconds. Print the optimal error in degrees.

Input

The first line of the input gives the number of test cases T $(1 \le T \le 100)$. T test cases follow.

For each test case, the first line contains an integer n $(1 \le n \le 5 \times 10^4)$, the number of watches.

Then n lines follow, the i^{th} of which contains a string of format $\texttt{HH:MM:SS}$ $(0 \le \texttt{HH} \le 23, 0 \le \texttt{MM}, \texttt{SS} \le 59)$, describing the time the i^{th} watch is showing.

The sum of n in all test cases doesn't exceed 10^5.

Output

For each test case, print a line with a real number α $(0 \le \alpha \le 180)$, representing the answer is $\alpha°$.

Your answers should have absolute or relative errors of at most 10^{-6}.

Example

standard input	standard output
2 2 12:30:00 02:40:00 3 00:00:00 23:59:59 06:30:30	32.5 91.55

Problem G. Go Running

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	512 mebibytes

Zhang3 is the class leader. Recently she's implementing a policy about long-distance running. This forces every student in her class to take a run.

There is a main road in the school from west to east, which can be regarded as an infinite axis, and its positive direction is east. Positions on the road are represented with numbers. In order to complete the task, each student must run along the main road. Each student can decide the following things:

- The time to start running.
- The time to finish running.
- The position to start running.
- The direction of running, which is either west or east.

Once these things are decided, the student will appear at the starting position on the road at the start time, then start running in the chosen direction at a speed of 1 `m/s`. The student disappears at the finish time. Each student will only run once.

Zhang3 knows that some students are not following her policy, so she set up some monitors. According to technical issues, the monitors can only report that there's at least one student at a certain place at a certain time. Finally Zhang3 received n reports.

Help Zhang3 determine the minimum possible number of students who have run.

Input

The first line of the input gives the number of test cases, T ($1 \le T \le 100$). T test cases follow.

For each test case, the first line contains an integer n ($1 \le n \le 10^5$), the number of reports.

Then n lines follow, the i^{th} of which contains two integers t_i, x_i ($1 \le t_i, x_i \le 10^9$), representing that there's at least one student at position x_i (m) at time t_i (s).

The sum of n in all test cases doesn't exceed 5×10^5.

Output

For each test case, print a line with an integer, representing the minimum number of students who have run.

Example

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

Problem H. Head Maker

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

Zhang3's favorite toy, Steve, has been hit and his head was broken. Zhang3 wants to make a cube as a new head for him.

Zhang3 got a piece of grid paper as material, satisfying the conditions below:

- The grid paper consists of $n \times m$ square units, forming n rows by m columns.
- Some units of the grid paper are broken, the others are usable.
- All of the usable units are 4-connected (two units are directly connected by sharing an edge).

By using some tools, she can do the following things for arbitrary times:

- Cut along an edge on the grid paper. After doing this, the paper mustn't be split into two pieces, i.e., all the usable units are still 4-connected.
- Fold along an edge, so that the two adjacent units form a right angle.

A cube has six faces. Every unit should overlap with one of the six faces. The paper is so thin that it's possible to overlap more than one units at the same face. But it's not allowed to leave a face without any units.

Help Zhang3 determine whether it is possible to make a cube with the grid paper.

Input

The first line of the input gives the number of test cases, T ($1 \le T \le 100$). T test cases follow.

For each test case, the first line contains two integers n, m ($1 \le n, m \le 6$), the size of the grid paper.

Then n lines follow, each contains a string of length m, describing the grid. Each character is 0 or 1, where 0 means the unit is broken, and 1 means the unit is usable.

Output

For each test case, print a line with a string `Yes` or `No`, representing the answer.

Example

standard input	standard output
2 2 5 11111 11000 2 5 11111 01100	No Yes

Problem I. Imperative Meeting

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

Some rumors about Zhang3 have been spread on the Internet. Zhang3 needs to dispel the rumors, but she's busy preparing her birthday party, so she asks her m classmates for help. Her classmates decide to have a meeting for discussion.

The m classmates live in the same community. There are n houses in the community, labeled $1, 2, \ldots, n$. There are $(n-1)$ roads connecting the houses, the i^{th} of which connects house $(i+1)$ and f_{i+1}, forming a tree. Each road is 1 km long.

The m classmates live in m different houses. They always choose such a house to have the meeting, that the total distance to travel for the m classmates is minimized. The optimal total distance (in km) to travel is called the cost of the meeting.

Zhang3 doesn't know which houses her classmates live in, so there are $\binom{n}{m}$ different cases of that. Zhang3 wants to know the sum of the cost in all cases. As the answer can be very large, please help her calculate the answer modulo $10^9 + 7$.

Input

The first line of the input gives the number of test cases, T $(1 \le T \le 1000)$. T test cases follow.

For each test case, the first line contains two integers n, m $(1 \le n \le 10^6,\ 1 \le m \le n)$, the number of houses in the community and the number of classmates.

The second line contains $(n-1)$ integers $f_2, \ldots, f_n$ $(1 \le f_i < i)$, separated by spaces, describing the roads.

The sum of n in all test cases doesn't exceed 2×10^6.

Output

For each test case, print a line with an integer, representing the answer modulo $10^9 + 7$.

Example

standard input	standard output
2 4 3 1 1 1 5 3 1 2 3 3	9 27

Problem J. Joyful Party

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	512 mebibytes

It is Zhang3's birthday! Zhang3 is going to hold a birthday party.

Zhang3 prepared n puzzles for the party, labeled $1, 2, \ldots, n$. She invited m friends to solve the puzzles. Each puzzle should be solved by exactly one friend, and each friend should solve at least one puzzle. The puzzles must be solved in some order, which means no two puzzles can be solved simultaneously.

As some puzzles are similar, it may be possible to solve a puzzle by imitating another one. This makes the puzzle solver very happy and produces some joy-value for the party. Specifically, there are some methods to solve a puzzle by imitating. Each of the methods is of the form (X, L, R, C), which means when someone is going to solve puzzle X, he can choose another puzzle $Y \in [L, R]$, as long as he has solved puzzle Y before, then use the method to solve puzzle X by imitating puzzle Y, producing C joy-value. Only one imitation can be done while solving a certain puzzle, i.e. only one method and one solved puzzle can be chosen to perform the imitation, even if there are multiple methods and solved puzzles meeting the requirements. Imitating nothing is also allowed, producing no joy-value.

Zhang3 wants to maximize the total joy-value. Please help her schedule the party (i.e. decide who to solve each puzzle, the order of solving, and the imitations) so that the sum of joy-value produced by imitations is maximized.

Input

The first line of the input gives the number of test cases, T $(1 \leq T \leq 2)$. T test cases follow.

For each test case, the first line contains three integers n, m, k $(1 \leq m \leq n \leq 10^5,\ k \leq 2 \times 10^5)$, the number of puzzles, the number of friends and the number of methods.

Then k lines follow, the i^{th} of which contains four integers X_i, L_i, R_i, C_i $(1 \leq L_i \leq R_i \leq n,\ 0 \leq C_i \leq 10^9)$, representing the i^{th} method is (X_i, L_i, R_i, C_i).

Output

For each test case, print a line with an integer, representing the maximized total joy-value.

Example

standard input	standard output
1 3 1 3 1 1 3 600000 1 3 3 666666 3 1 1 173768	773768

Problem K. Kindergarten Physics

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 seconds
Memory limit:	512 mebibytes

Zhang3 a participant of IPhO (Immortal Physics Olympiad). The 0^{th} problem in the contest is as follows.

There are two balls that weigh a kg and b kg respectively. They can be regarded as particles in this problem, as they are small enough. At the very beginning (i.e. $t = 0$), the distance between two balls is d km, and both of them are not moving.

Assuming that only gravitation works in this system (no other objects or other forces considered). The two balls has started moving since $t = 0$. Your task is to calculate the distance between them when $t = t_0$ (s).

Help Zhang3 solve the problem!

The following information might help when solving the problem.

- Universal gravitation formula: $F = G \cdot m_1 \cdot m_2 / r^2$
- Gravitational constant: $G = 6.67430 \times 10^{-11}\ \mathrm{m^3/(kg \cdot s^2)}$

Input

The first line of the input gives the number of test cases T ($1 \le T \le 100$). T test cases follow.

For each test case, the only line contains four integers a, b, d, t_0 ($1 \le a, b, d, t_0 \le 100$), representing the mass of the two balls, the initial distance between them, and how much time the balls move.

It is guaranteed that two balls will not collide within $(t_0 + 1)$ seconds.

Output

For each test case, print a line with a real number x, representing that the distance is x km.

Your answers should have absolute or relative errors of at most 10^{-6}.

Example

standard input	standard output
3 1 2 3 4 7 73 7 68 100 100 1 100	2.99999999999999999982 6.99999999999999974807 0.99999999999993325700

Problem L. Last Problem

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 seconds
Memory limit:	512 mebibytes

It's the night before Zhang3's birthday, and she's preparing for her birthday party in the classroom. She has brought a huge birthday cake and some powerful tomatoes, and has decorated almost every corner of the classroom. However, the blackboard is still empty. The last thing to do is to draw a beautiful pattern on it.

The blackboard is regarded as an infinite plane, each integer point (x, y) has an integer value as its color. At the very beginning, the color of every point is 0.

Zhang3 has n crayons, labeled $1, 2, \ldots, n$. Painting with the i^{th} crayon, you can replace the color of some chosen integer point (x, y) with color i. It is called a step, and she will draw the pattern step by step.

According to Zhang3's judgement of beautiful patterns, there's a restriction: Just before you paint (x, y) into some color i, the last four colors of i must appear among the adjacent points of (x, y). The last four colors of i means colors from $(i - 4)$ to $(i - 1)$, ignoring those non-positive ones. Two integer points are adjacent if their Euclid distance is exactly one. (Note that a point is not adjacent to itself.) If the condition above is not satisfied, the step is not allowed.

Zhang3 doesn't want to waste crayons, so the final pattern should contain at least one point with color n. Please help her find a way to draw such a beautiful pattern.

Input

The only line of the input contains an integer n ($1 \le n \le 100$), the number of crayons.

Output

Print the steps in chronological order, each in a separate line. Notice that you should not print the number of steps.

In the i^{th} line, print three integers x_i, y_i, c_i, separated by spaces, indicating the i^{th} step is to paint (x_i, y_i) into color c_i.

Your answer should satisfy $|x_i|, |y_i| \le 10^9$, $1 \le c_i \le n$. The number of steps should not exceed 10^5. The output file should not be larger than 5 MB.

It can be proved that there is always a solution. Any solution that meets all of the requirements will be accepted.

Example

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