Problem A. Maomao's candy

Input file:	stdin
Output file:	stdout
Time limit:	2 seconds
Memory limit:	256 megabytes

Maomao's candies were taken away by Dudu. Now Maomao wants to grab Dudu and bring the candies back.

Now they are in a rectangular area with n rows and m columns. The coordinates of the top left corner of the area are $(1, 1)$, and the co-ordinates of the bottom right corner of the grid are (n, m). Give the coordinates of the location where Maomao and Dudu were origi-nally located, they take turns to act and Dudu acted first. When it is Dudu's turn, Dudu will move to the adjacent grid. At the same time, he will eat some candy. The number of candy Dudu eats satisfies the following formula:

$$a_i = \frac{[(\frac{1+\sqrt{5}}{2})^i - (\frac{1-\sqrt{5}}{2})^i]}{\sqrt{5}}$$

means that if it is his i-th actions, he will eat a_i candies. When it is Maomao's turn, if Dudu's location is adjacent to or coincide with hers, she will catch Dudu and take back all her candies. Otherwise she will move to adjacent grid. Both Dudu and Maomao can not stand still in his(her) turn, which means they must move in their turns, and the adjacent grids of grid (i, j) are grid $(i - 1, j)$, $(i + 1, j)$, $(i, j + 1)$ and $(i, j - 1)$. They can not move outside the rectangular area. To eat candies as many as possible, Dudu wants to take more actions. While Maomao wants to get her candies back and get the least loss, so she will take actions with the strategy which can catch Dudu as soon as possible. Please calculate how many candies will Dudu eat before he is caught by Maomao.

Input

The first line an integer T, means T cases. $(1 \le T \le 10^5)$

For each case, input one line with 6 numbers n, m, r_1, c_1, r_2, c_2, means that the rectangle has n rows and m columns, and Maomao's initial location is (r_1, c_1), Dudu's initial location is (r_2, c_2). $(1 \le r_1, r_2 \le n \le 10^9$, $1 \le c_1, c_2 \le m \le 10^9)$

Their initial locations are distinct.

Output

For each data, if it's impossible for Maomao to catch Dudu, please output "countless". Otherwise output the number of candies Dudu will eat before he is caught by Maomao. Since the number will be very large, you should output the answer mod 1025436931.

Example

stdin	stdout
2 1 2 1 1 1 2 3 3 1 1 3 3	1 4

Problem B. Dudu's maze

Input file:	`stdin`
Output file:	`stdout`
Time limit:	1 seconds
Memory limit:	256 megabytes

To seek candies for Maomao, Dudu comes to a maze. There are n rooms numbered from 1 to n and m roads.

There are two kinds of rooms in the maze – candy room and monster room. There is one candy in each candy room, and candy room is safe. Dudu can take the only candy away when entering the room. After he took the candy, this candy room will be empty. A empty room is also safe. If Dudu is in safe, he can choose any one of adjacent rooms to go, whatever it is. Two rooms are adjacent means that at least one road connects the two rooms.

In another kind of rooms, there are fierce monsters. Dudu can't beat these monsters, but he has a magic portal. The portal can show him a randomly chosen road which connects the current room and the other room.

The chosen road is in the map so Dudu know where it leads to. Dudu can leave along the way to the other room, and those monsters will not follow him. He can only use the portal once because the magic energy is not enough.

Dudu can leave the maze whenever he wants. That's to say, if he enters a monster room but he doesn't have enough energy to use the magic portal, he will choose to leave the maze immediately so that he can save the candies he have. If he leave the maze, the maze will never let him in again. If he try to fight with the monsters, he will be thrown out of the maze (never let in, of course). He remembers the map of the maze, and he is a clever guy who can move wisely to maximum the expection of candies he collected.

Maomao want to know the expection of candies Dudu will bring back. Please tell her the answer. He will start his adventure in room 1, and the room 1 is always a candy room. Since there may be more than one road connect the current room and the room he wants to go to, he can choose any of the roads.

Input

First line a integer t, means t cases. $1 \leq t \leq 5$

For each case:

First line 3 integer n, m, k, n means the number of rooms, m means the number of roads, k means the number of monster rooms. $1 \leq n \leq 100000$, $n-1 \leq m \leq 2*n$, $0 \leq k \leq n$

Next m lines, for each line there are two integer a and b, separated by a space, means there is a road between a and b. There may be repeated edges, but won't be self loop. $1 \leq a, b \leq n$

In the last line there are k distinct numbers, the i-th number x_i means the number of the i-th monster room is x_i, and room 1 won't be monster room. $1 \leq i \leq k$

Output

For each case output a real number. The absolute error of the answer should not exceed 10^{-6}.

Example

stdin	stdout
2 7 6 2 1 2 1 3 2 5 2 4 3 6 4 7 2 3 7 7 2 1 2 1 3 2 5 2 4 3 6 4 7 2 4 2 3	2.000000 2.250000

Problem C. Dawn-K's water

Input file:	stdin
Output file:	stdout
Time limit:	1 seconds
Memory limit:	256 megabytes

Dawn-K recently discovered a very magical phenomenon in the supermarket of Northeastern University: The large package is not necessarily more expensive than the small package.

On this day, Dawn-K came to the supermarket to buy mineral water, he found that **there are n types of mineral water**, and he already knew **the price** p and **the weight** c **(kg)** of each type of mineral water. Now Dawn-K wants to know **the least money** a he needs to buy **no less than** m **kilograms** of mineral water and **the actual weight** b **of mineral water he will get**. Please help him to calculate them.

Input

The input consists of multiple test cases, each test case starts with a number n ($1 \leq n \leq 10^3$) – the number of types, and m ($1 \leq m \leq 10^4$) – the least kilograms of water he needs to buy.

Then followed n lines with each line two integers p ($1 \leq p \leq 10^9$) – the price of this type, and c ($1 \leq c \leq 10^4$) – the weight of water this type contains.

Output

For each test case, you should output one line contains the minimum cost a and the weight of water Dawn-K will get b. If this minimum cost corresponds different solution, output the maximum weight he can get.

(The answer a is between 1 and $1e9$, and the answer b is between 1 and $1e4$)

Example

stdin	stdout
3 3 2 1 3 1 1 1 3 5 2 3 1 2 3 3	3 3 3 6

Problem D. Fish eating fruit

Input file:	stdin
Output file:	stdout
Time limit:	1 seconds
Memory limit:	256 megabytes

State Z is a underwater kingdom of the Atlantic Ocean. This country is amazing. There are n cities in the country and $n-1$ undirected underwater roads which connect all cities.

In order to save energy and avoid traffic congestion, the king promulgated a series of traffic regulations:

1. Residents have to travel on fish!
2. Residents need to feed the fish before you start your trip!The amount of food you feed the fish should be exactly the total distance of your journey.
3. What kind of food to feed depends on the total distance of your journey! Total distance is a multiple of three. You should feed the fish with Durian. Total distance modulus 3 equaling 1. It should be Papaya.Total distance modulus 3 equaling 2. It should be Avocado!!!

Sure, fish like to eat these fruits very much.

Today is the first day of the king's decree. Because the residents here are not good at mathematics, they don't know how much fruit they need to give fish to go to other cities. So the king give an order to the energy minister Ynaonlctrm **From all cities to all cities directly**, which means that he will make $n*(n-1)$ trips.

For example, A - (5 *mile*) - B - (5 *mile*) - C, he needs to run the fish, starting at A, passing B, finally arriving C (papaya 10 *kg*), also he needs to start at C and end at A (papaya 10 *kg*). Indirect passage is useless. "I've passed City B, my dear emperor." "Oh! It's no use! Not directly! People in cities will never know how much the fish need to eat! The fish will surely die!!! You also need to take several trips which start at B or end with B." The Emperor said.

It's really a tough task. Can you help him figure out how much fruit he needs to prepare for the emperor's mission?

Input

Multiple input!

Fist line is N. next $N-1$ line has three integer a, b and c. It represent city a and city b is connected by a road of c nautical miles. ($1 < n \leq 10^4$, $0 \leq a, b < n$, $1 \leq c < 10^5$, $\sum n \leq 10^5$)

Output

For each data, output three number, the amount of Durian, Papaya and Avocado he need. (the result could be very large, please output the result mod 10^9+7)

Example

stdin	stdout
5 0 1 2 0 2 3 0 3 7 0 4 6	54 60 30

stdin	stdout
8 4 7 1 7 5 1 4 6 1 6 3 1 5 2 1 2 1 1 7 0 1	48 54 48

Problem E. Gugugu's upgrade schemes

Input file:	stdin
Output file:	stdout
Time limit:	2 seconds
Memory limit:	256 megabytes

Recently, Gugugu is addicted with a kind of auto chess. In this auto chess, there are many different kinds of powerful weapon. And all weapons can be divided into two different kinds, basic weapon and upgraded weapon. Basic weapon is the most basic weapon, it means you can't split them. Upgraded weapon can be made by using other two weapons (after that these two weapons will be consumed), And the upgraded weapon is distinguished from other upgraded weapon only by the basic weapons which made it up. It means if two upgraded weapons have same composition, they will be regard as one weapon.

Now Gugugu only have n different basic weapon, she wants to know how many upgrade schemes she can use. Two scheme regard different if and only if it exists at least one weapon one scheme has, but the other doesn't. Additional, because Gugugu have poor memory, she can only count number smaller then p, so you also should tell she the answer module p.

Input

The first line contains a integer T ($T \leq 50$).

Then follow T line, each line contains a integer n, p ($0 < n < 10^6$, $1 < p < 1000$), meaning the number of basic weapons Gugugu has and the number Gugugu can count smaller than. It guarantee that p is a prime number, and $n/p < 1000$.

Output

For each test case, output a line contain only one integer denoting the answer.

Example

stdin	stdout
3 1 997 5 997 990320 997	1 52 363

Problem F. Honk's pool

Input file:	stdin
Output file:	stdout
Time limit:	1 seconds
Memory limit:	256 megabytes

As we all know, Honk has n pools, numbered as 1 ~n . There is a_i liters water in the i-th pool. Every day, Honk will perform the following operations in sequence.

1. Find the pool with the most water (If there are more than one, choose one at random) and take one liter of water.
2. Find the pool with the least water (If there are more than one, choose one at random) and pour one liter of water into the pool.
3. Go home and rest (Waiting for the next day).

Please calculate the difference between the amount of water in the pool with the most water and the amount of water in the pool with the least water after the k days.

Input

The input consists of multiple test cases. The input is terminated by the end of file.

The first line of each test case contains two integers n and k, which indicate the number of pools and the number of days to operate the pool.

The second line of each test case contains n integers, and the i-th number represent a_i indicating the initial amount of water in the i-th pool.

$1 \leq n \leq 500000$, $1 \leq k \leq 10^9$, $1 \leq a_i \leq 10^9$.

Output

For each test case, print one line containing the answer described above.

Example

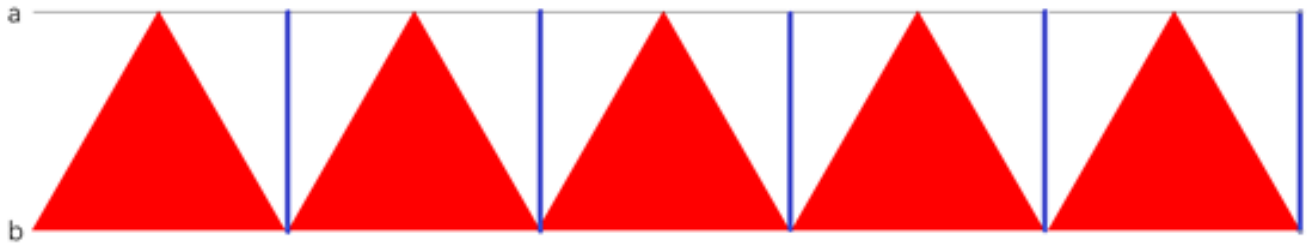
stdin	stdout
4 100 1 1 10 10	1

stdin	stdout
4 3 2 2 2 2	0

Problem G. Special necklace

Input file:	stdin
Output file:	stdout
Time limit:	1 seconds
Memory limit:	256 megabytes

The SPLENDID XZG want to give his GIRLFRIEND a SPLENDID gift so that he can surprise her! He carefully selected diamonds and finally he decides to use one kind of ruby of regular triangle shape. He then adds some broken drill bars to decorate the necklace so that the necklace can be much more exquisite.



In order to let his girlfriend be moved by his special romance of science students, XZG wants to tell the resistance value between point a and point b of the necklace to his girlfriend. When we only consider one single regular triangle ruby, he measures the resistance of any two endpoints of it, the resistance values are all $2a$. Then he measured the resistance of the broken drill bar, and the resistance value is a. Moreover, the top black line is a wire whose resistance is 0.

So, What's the resistance value of one unfolded necklace with n regular triangle rubies and n broken drill bars?

Input

The first line contains a single integer T, which indicates the number of tests. $T \leq 100$.

Then the next T lines contain two numbers n and a, representing the number of rubies and the resistance of one broken drill bar, respectively.

$1 \leq n \leq 10^{1000000}$, $0.1 \leq a \leq 10^6$.

Output

T lines in total. Each line output a number of resistance values. Keep 10 decimal places. The absolute error of the answer should not exceed 10^{-6}.

Example

stdin	stdout
1 1 1	1.6666666667

Problem H. Texas hold'em Poker

Input file:	`stdin`
Output file:	`stdout`
Time limit:	1 seconds
Memory limit:	256 megabytes

Recently, Yang was addicted to Texas hold'em Poker. So he found a lot of people to play with him. Due to the large number of people, Yang had to change the rules of the game:

1. All poker cards are counted by number without suit.
2. Each card has a value which is one of 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13 (denoted A, 2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K)
3. Each player drew five poker cards from a complete deck of poker (without jokers), and the possible hand values are ranked as follows from lowest to highest:
   - **High Card.** Hands which do not fit any higher category are ranked by the sum of all the cards.
   - **Pair.** 2 of the 5 cards in the hand have the same value. Hands which both contain a pair are ranked by the value of the cards forming the pair. If these values are the same, the hands are ranked by the sum of the rest cards.
   - **Two Pairs.** The hand contains 2 different pairs. Hands which both contain 2 pairs are ranked by the value of their highest pair. Hands with the same highest pair are ranked by the value of their other pair. If these values are the same the hands are ranked by the value of the remaining card.
   - **Three of a Kind.** Three of the cards in the hand have the same value. Hands which both contain three of a kind are ranked by the value of the 3 cards. Hands with the same 3 cards are ranked by the sum of the rest two cards.
   - **Full House.** 3 cards of the same value, with the remaining 2 cards forming a pair. Ranked by the value of the 3 cards. Hands with the same 3 cards are ranked by the value of the cards forming the pair.
   - **Four of a kind.** 4 cards with the same value. Ranked by the value of the 4 cards. Hands with the same 4 cards are ranked by the remaining card.

     item **Straight.** Hand contains 5 cards with consecutive values. Hands which both contain a straight are ranked by their highest card.
   - **Royal Straight.** Straight from 10 to A (10-J-Q-K-A). The largest hand!

Now, Yang has known everyone's cards, he wants a ranking list of all poker hands. If the value of players are equal, output their names in lexicographical order. It's guaranteed that no one's name repeats. Can you just help him?

Input

The input consists of multiple test cases, Each test case starts with a number n ($1 \le n \le 10^5$) represents the number of players, Then followed n lines with each line two string m ($1 \le |m| \le 10$) – the name of the player s ($1 \le |s| \le 10$) the poker cards of the player.

Output

For each test case, you should output n lines represent the ranking list.

Example

stdin	stdout
3 Alice AAA109 Bob 678910 Boa 678910	Boa Bob Alice

Problem I. Self-game

Input file:	stdin
Output file:	stdout
Time limit:	12 seconds
Memory limit:	256 megabytes

After a long programming training, ghh feels a little tired. So he comes up with a game that he can entertain himself. He calls this game a self-game. Thus, ghh and his inner personality gfh's game begin. The rules of the self-game are as follows: First, there is a board with a size of $N * M$, and there are some obstacle points on the board. At the beginning, there are two pieces placed on the board. Attention that there is no difference between the two pieces. During the game, ghh and gfh takes turns to move, and ghh first. When it is someone's turn to move, he can choose anyone of the two pieces and move it only one step in any direction (up, down, left and right). Of course, the pieces cannot be moved outside the board and cannot be moved to the obstacle points. The positions of the two pieces cannot overlap, too. If someone can not move the piece, then he loses.

However, when ghh playes this self-game with his inner personality gfh using the rule above for a while, he discovers that his rules are a bit problematic. He realizes that this game will never have a winner. So, he adds a new rule: assuming that the current game is at a certain moment, it happens to be someone's action turn. If he chooses one piece and moves the piece one step. After this movement, the situation appearing on the board has already appeared before this moment. Then this step of movement is not allowed in this case.

After adding this rule, ghh can finally play the self-game with gfh. Now ghh wants to ask you, for a board that gives the initial positions of the two pieces, who will be the winner?

Note that both ghh and gfh will choose the optimal strategy during the game.

Input

The input contains several test cases, and the number of test cases does not exceed 10.

For each test case, the first line contains three positive integers N, M, K, where N, M indicate the size of the board, and K indicates there are K different initial positions of the two pieces. $N, M \leq 10, K \leq 20$.

Then the following N lines and M columns indicate the board, where '.' indicates an empty point and '#' indicates an obstacle point.

After that there are K lines, each line contains four positive integers x_1, y_1, x_2, y_2, which indicate a different initial position of the two pieces.

Output

For each test cases, output K lines, each line contains only 'ghh wins.' or 'gfh wins.'

Example

stdin	stdout
1 10 3 .....#.... 1 1 1 5 1 7 1 10 1 1 1 10	ghh wins. gfh wins. ghh wins.

Problem J. Ghh Matin

Input file:	stdin
Output file:	stdout
Time limit:	1 seconds
Memory limit:	256 megabytes

Similar to the strange ability of Martin (the hero of Martin Martin), Ghh will random occurrence in one of N cities every morning and the money will change to X RMB (No matter how much money had yesterday).

Ghh finds that every N cities have only one way to enter and go out (for example, city A's out road is connected with city B's enter road, or city A's out road may also be connected with city A's enter road). After waking up one day, Ghh suddenly wants to do an experiment.

In the following days, Ghh will judge whether he has experimented in the city every time he wakes up. If not, he will go out to see if he can start from the city and finally return to itself. If he can't or has already done it, he will stay until the evening and go to the next random city by sleeping. This experiment lasts until every N cities have been done.

Because the distances between cities are very far, Ghh decides to take bus. The cost of bus between any two cities is 1 RMB (it also costs 1 RMB for the city that the enter road is connected with the out road of itself) and the speed is very fast. What is the probability that Ghh can return to the city itself from every city when Ghh has finished the experiment?

Note: When $n = 2$, there are two cases in city A and B:

1. A connected with A, B connected with B
2. A connected with B

When X is 1, only in the first case every city can return to itself, so the probability is $1/2$. When X is 2, any case is ok.

Input

The first line, input a integer T represents the number of test cases ($T \leq 10^4$). Next 2 ˜T+1 line, input two integers N, X represents the total number of cities and initial money ($2 \leq N \leq 10^6$, $N \leq 2X \leq 2 \times 10^9$). $\sum N \leq 10^7$.

Output

Each test case output one line, the probabilities may be a/b, please output $a * inv(b) \mod (10^9 + 7)$. $inv(b)$ is the inverse of b.

Example

stdin	stdout
3 2 1 2 2 145 134	500000004 1 772215686

Problem K. Guanguan's Happy water

Input file:	stdin
Output file:	stdout
Time limit:	4 seconds
Memory limit:	256 megabytes

Rather than drinking happy water, Guanguan loves storing happy water. So he bought a refrigerator and stored a_i bottles of cola into it every day. When the storage is finished on the k-th day, the refrigerator is full, but he still wants to store happy water every day. Here comes the solution: He first constructs a p-sequence: p_1, p_2, ..., p_k, where $p_1 + p_2 + ... + p_k = 1$. Then he chooses an number i among 1 to k, where number i has the probability p_i to be chosen. After that, he drinks the happy water stored on the i-th day before the current day and stores the same amount of happy water back into the refrigerator again. Let the amount of happy water stored on the i-th day be f_i. Given the amount of happy water stored in the first k days and the expected amount of the next k days(which means, from the $k+1$-th day to the $2k$-th day), could you help Guanguan figure out the sum of the expected amount of happy water stored during the first n days) (Be aware that every element of f has moded $1e9+7$ when input datas, and your output should mod $1e9+7$ as well)

Input

The first line is T ($1 \le T \le 20$), indicating the number of input sets. For each set of inputs, the first line is k and n ($1 \le k \le 70$, $1 \le n \le 10^{18}$), and the second line is $2k$ numbers, respectively representing a_1, a_2, ..., a_k, f_{k+1}, f_{k+2}, ..., f_{2k}.

Output

For each data, output a non-negative integer indicating $(\sum_{i=1}^{n} f_i) \mod 10^9 + 7$.

Example

stdin	stdout
2 1 9 2 2 2 8 6 5 5 5	18 41