A - Problem A

Source file name: `bully.py`
Time limit: 1 second

Eloy is a hard worker man, however, he is constantly bullied by his superiors, molested by this, one day he was wondering in what "rank" you are, so you can bully the people with lower ranks, also to discover who can really bully Eloy!.

Now, given the number of employees and the number of relations between them, Eloy need you to output the "rank" which employee is in, being 1 the "boss" (not bullied by anybody) and the employee who are in these ranks.

Input

There will be an integer T denoting the number of test cases, then, T test cases will follow. Each test case starts with two integers N ($1 \leq N \leq 1000$) and R ($1 \leq R \leq 10000$), the number of employees and the number of relations between them. The next R lines consists of two integers R_1 and R_2, meaning that "employee R_1 is lower than employee R_2's rank".

The input must be read from standard input.

Output

You will output for each test case the string `''Scenario #i:''` where `i` is the test case you are analyzing, after that, you will print N lines, for each line you will output the rank of the employee and the employee itself, if there is the same rank for several employees, then output them lexicographically ordered (the first is the lower.)

The output must be written to standard output.

Sample Input	**Sample Output**
2 5 6 2 0 2 4 1 4 1 2 3 2 4 0 5 4 1 0 2 0 3 2 4 2	Scenario #1: 1 0 2 4 3 2 4 1 4 3 Scenario #2: 1 0 2 1 2 2 3 3 3 4

B - Problem B

Source file name: `gems.py`
Time limit: 1 second

Zlatan is building a retro basic computer game. It's a very simple text-based adventure where a warrior roams around trying to find treasure (adamantium gems) while avoiding hidden pitfalls. The game takes place on a rectangular grid, and the player has very limited information about their surroundings.

The game involves the warrior moving around on the grid for as long as she likes (or until she falls into a pitfall). The warrior can move up, down, left, and right (but not diagonally). She will pick up gems if she steps into the same square as the gem. If the warrior stands next to one or more pitfalls (i.e., immediately up, down, left, or right of them), she will "sense a draft" but will not know from which direction the draft is coming or how many traps are nearby. If she tries to move into a square containing a wall, she will notice the wall in that direction and remain in her current position.

For scoring purposes, Zlatan wants to show the warrior how many gems she could have collected safely. In other words, how many gems can a player collect using an optimal strategy while always ensuring that the square she steps into is safe. The player does not have access to the map, and the maps are randomly generated for each game, so she has no prior knowledge of the layout.

Input

The input contains several test cases, each of them as follows. The first line of input contains two positive integers W and H, neither of them smaller than 3 or larger than 101, giving the width and the height of the map, respectively. The next H lines contain W characters each, giving the map. The symbols that may occur in a map are as follows:

`P` the warrior's starting position.

`G` a piece of gold.

`T` a pitfall.

`#` a wall.

`.` normal floor.

There will be exactly one '`P`'in the map, and the border of the map will always contain walls.

The input must be read from standard input.

Output

For each test case, write to the output the number of gems the warrior can obtain without risking falling into a pitfall, on a line by itself.

The output must be written to standard output.

Sample Input	**Sample Output**
7 4 ####### #P.GTG# #..TGG# ####### 8 6 ######## #...GTG# #..PG.G# #...G#G# #..TG.G# ########	1 4

C - Problem C

Source file name: `slowcalm.py`
Time limit: 1 second

Zlatan studies and works, but his salary does not allow him to save enough money to buy his favorite car, so every time he wants to go out with someone he is in love with, he has to use his mother's car. This car is slow, and it has been many years since his mother bought it, so the radio and air conditioning no longer work, but this is not an impediment for Zlatan to have fun in it. Zlatan is very reckless and on his last outing with Booleana he decided to go at top speed to impress her, but the vibrations of the old engine caused the speedometer to come off. Zlatan glued it back together, but he's not a handyman, and now the speedometer reads wrong. So, when the speedometer says it is going x, in reality its speed is $x + y$, where y is an unknown constant, which may be negative.

Zlatan carefully recorded one of his last trips with Booleana and wants to use this information to know the y value, as he likes to show off his programming skills, he will possibly use this information to further impress his dates. The trip consisted of M segments. On the i-th segment he traveled a distance d_i and the speedometer was reading s_i for the entire segment. The entire trip took a T amount of time.

Zlatan's computer is not very good and cannot do very large calculations, so he asks you to help him find the value of y using the information he recorded on his last trip. Zlatan makes it clear to you that the speedometer can read negative speeds, so his true speed was greater than zero for each segment of his trip.

Input

The input file contains several test cases, each of them as described below. The first line of input contains two integers n ($1 \leq n \leq 1000$), the number of segments in Zlatan's trip, and T ($1 \leq T \leq 10^6$), the total time. This is followed by n lines, each describing one segment of Zlatan's journey. The i-th of these lines contains two integers d_i ($1 \leq d_i \leq 1000$) and s_i ($|s_i| \leq 1000$), the distance and speedometer reading for the i-th segment of the trip. Time is specified in hours, distance in kilometres, and speed in kilometrres per hour.

The input must be read from standard input.

Output

For each test case, on a line by itself, display the constant y in kilometres per hour. Your answer should be given to six decimal places.

The output must be written to standard output.

Sample Input	Sample Output
3 5 4 -1 4 0 10 3 4 10 5 3 2 2 3 6 3 1	3.000000 -0.508653