'Hush!' said Frodo. 'I think I hear hoofs again.'

They stopped suddenly and stood as silent as tree-shadows, listening. There was a sound of hoofs in the lane, some way behind, but coming slow and clear down the wind. Quickly and quietly they slipped off the path, and ran into the deeper shade under the oak-trees.

The hoofs drew nearer. They had no time to find any hiding-place better than the general darkness under the trees.

- Frodo, Sam and Pippin, when they encounter a Black Rider.

Indeed, the Black Riders are in the Shire, and they are looking for the One Ring. There are N hobbits out in their fields, but when they hear the Riders approaching, or feel the fear cast by their presence, they immediately wish to run and hide in M holes located nearby.

Now, each hole has space for just 1 hobbit; however, once a hobbit reaches a hole, he is able to make room for one more hobbit by digging away at the earth. The time required to make enough space for a second hobbit is C units. Also, each hole CANNOT hold more than 2 hobbits, even after digging. Also note that a hobbit can begin making space for the next hobbit only after he reaches the hole.

You are given the time required to travel from each hobbit's current location to each hole. Find the minimum amount of time it will take before at least K of the hobbits are hiding safely.

Input

The first line contains T, the number of test cases.

The first line of each test case contains 4 integers – N (no of hobbits), M (no of holes), K (minimum number of hobbits to hide) and C (time taken to expand a hole). The next N lines contain M integers each, denoting the time taken for each hobbit to each hole.

Output

Output one line per test case which contains the minimum time.

Constraints:

$1 \le T \le 6$

$1 \le N, M \le 100$

$1 \le K \le \min(N, 2 * M)$

$0 < C < 10,000,000$

$0 <$ Time taken by the hobbits to the holes $<$ 10,000,000

Notes/Explanation of Sample Input:

For the first test case, there are 3 hobbits and 3 holes, and we need to get atleast 2 of them to safety. We can send the first hobbit to the first hole, and the second hobbit to the second hole, thereby taking 10 time units.

For the second test case, we can make hobbit #1 reach hole 1 at time 1, hobbit #2 reach hole 1 at time 9 (by when hobbit #1 would have finished digging the hole), and hobbit #3 reach hole 3 at time 6.

Sample Input

```
2
3 3 2 10
9 11 13
2 10 14
12 15 12
4 3 3 8
1 10 100
1 10 100
100 100 6
12 10 10
```

Sample Output

```
10
9
```

> *'We fought far under the living earth, where time is not counted. Ever he clutched me, and ever I hewed him, till at last he fled into dark tunnels. Ever up now we went, until we came to the Endless Stair. Out he sprang, and even as I came behind, he burst into new flame. Those that looked up from afar thought that the mountain was crowned with storm. Thunder they heard, and lightning, they said, smote upon Celebdil, and leaped back broken into tongues of fire.'*
>
> - Gandalf, describing his fight against the Balrog.

Although Gandalf would not go into the details of his battle, they can be summarized into the following simplified form: both Gandalf and the Balrog have a set of N attacks they can use (spells, swords, brute-force strength etc.). These attacks are numbered from 1 to N in increasing order of Power. When each has chosen an attack, in general, the one with the higher power wins. However, there are a few (M) anomalous pairs of attacks, in which the lesser-powered attack wins.

Initially, Gandalf picks an attack. Then the Balrog counters it with one of the remaining attacks. If the Balrog's counter does not defeat Gandalf's, then we say Gandalf receives a score of 2. If however it does, then Gandalf has exactly one more opportunity to pick an attack that will defeat the Balrog's. If he manages to defeat him now, his score will be 1, whereas if he is still unable to defeat him, his score will be 0.

Your task is to determine, given N and the M anomalous pairs of attacks, what will be Gandalf's score, given that both play optimally. Further, in case Gandalf gets a score of 2, you must also determine which attack he could have chosen as his first choice.

Note 1: The Balrog can choose only one attack, whereas Gandalf can choose upto two.
Note 2: The relation A defeats B is not transitive within the attacks. For example, attack A can defeat attack B, attack B can defeat attack C, and attack C can defeat attack A.
Note 3: Between any two attacks A and B, either attack A defeats attack B or attack B defeats attack A.

Input

The first line will consist of the integer T, the number of test-cases.

Each test case begins with a single line containing two integers N and M. This is followed by M lines consisting of 2 integers each x and y, denoting that x and y are an anomalous pair.

Output

For each test-case, output a single line either

`2` A, if Gandalf can defeat any attack the Balrog chooses if he picks attack A,
`1`, if Gandalf can choose an attack such that even if the Balrog chooses an attack to defeat him, he can choose an attack to defeat the Balrog's chosen card,
`0`, if none of the above two options are possible for all possible choices of Gandalf's attack(s). Between successive test cases, there should not be any blank lines in the output.

Constraints:
$1 \leq T \leq 15$
$3 \leq N \leq 1,000,000$
$0 \leq M \leq \min(N(N-1)/2, 300,000)$
$1 \leq x < y \leq N$ for all the anomalous pairs (x, y)
The sum of M over all test-cases will not exceed 300,000.

Notes/Explanation of Sample Input:

In the first case, attack 3 can beat both attacks 1 and 2. So Gandalf just chooses attack 3.

In the second case, attack 1 beats 3 which beats 2 which beats 1. No matter which attack Gandalf chooses, the Balrog can pick the one which defeats his, but then he can pick the remaining attack and defeat the Balrog's.

Sample Input

```
2
3 0
3 1
1 3
```

Sample Output

```
2 3
1
```

'Hoo, ho! Good morning, Merry and Pippin!' he boomed, when he saw them. 'You sleep long. I have been many a hundred strides already today. Now we will have a drink, and go to Entmoot.'

'Where is Entmoot?' Pippin ventured to ask.

'Hoo, eh? Entmoot?' said Treebeard, turning round. 'It is not a place, it is a gathering of Ents - which does not often happen nowadays. But I have managed to make a fair number promise to come.'

Indeed, Entmoot cannot be thought of as any particular place, since where it occurs changes from time to time. The choice of the location however, follows a basic principle: Although the Ents (walking and talking Trees – "shepherds of Fangorn Forest") are by and large not hasty, when it comes to gathering for Entmoot, they would like to choose a location that ensures all the Ents can reach as soon as possible.

Note however, that the speed they can travel varies from Ent to Ent. Also, although Fangorn Forest has dense overgrowth, with regard to the Ents, the forest poses no obstacles.

You are given the locations of N Ents who have agreed to join in for Entmoot, as well as their speeds. You need to find out where Entmoot will occur. Formally, given the (x_i, y_i) along with speed s_i for each Ent, find the point (X, Y) such that the maximum time taken by any of the Ents to reach (X, Y) is minimized. Output the earliest time when all the Ents can meet.

Input

The first line contains T, the number of test cases.

The first line of each test case contains N, the number of Ents. The next N lines contain three space-separated integers each. The ith of these lines contains :x_i, y_i, s_i.

Output

Output one line per test case, containing the earliest time when the Ents can meet. Relative and absolute error of 10^{-4} are acceptable.

Constraints:
$1 \leq T \leq 10$
$2 \leq N \leq 50$
$-1,000,000 \leq x_i, y_i \leq 1,000,000$
$1 \leq s_i \leq 1,000,000$

Notes/Explanation of Sample Input:

In the first test case, all the ents can meet at origin in 1 unit of time.

In the second test case, the first and the third ent reach (25,0) after 7.5 units of ime, whereas the second ent reaches there after 2.5 units of time and waits for the remaining ents to arrive.

In the third test case, all the ents can meet at origin in 100/15 units of time.

Sample Input

```
4
3
0 3 3
4 0 4
-3 -4 5
3
0 10 2
0 20 2
0 40 2
3
0 100 15
0 -100 15
8 0 7
3
0 0 1
10000 0 1
5000 8661 1
```

Sample Output

```
1.000000
7.500000
6.666667
5773.751357
```

> *With water from the stream Galadriel filled the basin to the brim, and breathed on it, and when the water was still again she spoke. 'Here is the Mirror of Galadriel,' she said. 'I have brought you here so that you may look in it, if you will. For this is what your folk would call magic, I believe; though I do not understand clearly what they mean; and they seem also to use the same word of the deceits of the Enemy. But this, if you will, is the magic of Galadriel. Did you not say that you wished to see Elf-magic?'*
>
> – Galadriel to Frodo and Sam, describing her Mirror.

We call a string S magical if every substring of S appears in Galadriel's Mirror (under lateral inversion). In other words, a magical string is a string where every substring has its reverse in the string.

Given a string S, determine if it is magical or not.

Input

The first line contains T, the number of test cases. The next T lines contain a string each.

Output

For each test case, output '`YES`' if the string is magical, and '`NO`' otherwise.

Constraints:
$1 \leq T \leq 100$
$1 \leq |S| \leq 10$
S contains only lower-case characters.

Notes/Explanation of Sample Input:

For the first test case, the list of substrings are : `a`, `b`, `ab`, `ba`, `aba`. The reverse of each of these strings is present as a substring of S too.

For the second test case, the list of substring are : `a`, `b`, `ab`. The reverse of '`ab`', which is '`ba`' is not present as a substring of the string.

Sample Input

```
2
aba
ab
```

Sample Output

```
YES
NO
```

> *'Light, light of Sun and Moon, he still feared and hated, and he always will, I think; but he was cunning. He found he could hide from daylight and moonshine, and make his way swiftly and softly by dead of night with his pale cold eyes, and catch small frightened or unwary things. He grew stronger and bolder with new food and new air. He found his way into Mirkwood, as one would expect.'*
>
> – Gandalf, describing Gollum after he ventured forth from Moria.

Gollum has spent half a millennium in the long darkness of Moria, where his eyes grew used to the dark, and without caring for reading or writing, he became dyslexic. Indeed, as much as he hates the Moon and the Sun, he also hates strings with long palindromes in them.

Gollum has a tolerance level of K, which means that he can read a word so long as it does not contain any palindromic substring of length K or more. Given the values N and K, return how many BINARY strings of length N can Gollum tolerate reading.

Input

The first line contains T, the number of test cases.

Each test case consists of one line containing 2 integers, N and K.

Output

For each test case, output the answer modulo 1,000,000,007.

Constraints:

$1 \leq T \leq 100$
$1 \leq N \leq 400$
$1 \leq K \leq 10$

Notes/Explanation of Sample Input:

For the first test case, 01 and 10 are the valid binary strings, while 00 and 11 are invalid.

For the second test case, 001, 011, 100, 110 are the valid binary strings.

For the third test case, all possible binary strings of length 3 are valid.

Sample Input

```
3
2 2
3 3
3 4
```

Sample Output

```
2
4
8
```

'Though the Stewards deemed that it was a secret kept only by themselves, long ago I guessed that here in the White Tower, one at least of the Seven Seeing Stones was preserved. In the days of his wisdom Denethor did not presume to use it, nor to challenge Sauron, knowing the limits of his own strength. But his wisdom failed; and I fear that as the peril of his realm grew he looked in the Stone and was deceived: far too often, I guess, since Boromir departed. He was too great to be subdued to the will of the Dark Power, he saw nonetheless only those things which that Power permitted him to see.'

– Gandalf.

Sauron and Saruman have been communicating from large distances using the Seeing Stones. Denethor, with great difficulty has been able to break into their channel of communication using his strength of will. Despite this however, it seems that their communication has been encrypted. Gondor's spies in Isengard have found out the encryption algorithm they use and have reported back to Denethor.

The algorithm is as follows :

We refer to a dequeue as a double-ended queue. We define a "dequeue permutation of N" as a permutation of 1 to N that can be got by starting from a dequeue having elements 1, 2, 3, ..., N (in that order with 1 at the front and N at the back) and performing any sequence of N `pop_front()` or `pop_back()` operations.

Note that not all permutations of 1 to N are dequeue permutations. For example, with $N = 3$, you have 3-1-2, 1-3-2 etc. as dequeue permutations whereas 2-1-3, 2-3-1 aren't (you can't have 2 right at the beginning since its not at any end of the dequeue).

If Sauron wants to encrypt the number K and send it to Saruman, he would instead send the K'th lexicographically smallest (0-based indexed) dequeue permutation of N. That is, if Sauron wanted to send '0' to Saruman, he would just send 1-2-3- $\ldots$-N (since this is clearly the smallest lexicographic dequeue permutation of N).

Sauron is transmitting the size of his army to Saruman, so that they can coordinate an attack on the Men of Rohan and Gondor. Since Sauron's will is so powerful, Denethor is able to get only vague glimpses of the numbers, and he is able to remember only the first half ($floor(N/2)$ elements) of the permutation. Further, since these images are so vague, his understanding of the numbers happens out of order. For example, Denethor may understand that the 5th number of the permutation is 4, and later on only understand that the 3rd number was 3.

Help him estimate the minimum and maximum possible size of Sauron's army (value of K), given the number N, and incremental understanding of the first half of the permutation, not necessarily in order.

NOTE 1: Since the values to be output can be rather large, output the values modulo 1,000,000,007.
NOTE 2: It may be the case that Denethor's understanding of the permutation is flawed. If it is not possible to have a dequeue permutation satisfying all the conditions seen so far, output '`-1`'.
NOTE 3: A permutation $p1$ is lexicographically smaller than $p2$ if at the first position where they differ, $p1$'s value is smaller than $p2$'s.

Input

The first line consists of the integer T, the number of test cases.

Each test case begins with a single integer N. This is followed by exactly $floor(N/2)$ lines containing 2 integers each: i and j, denoting that Denethor has understood that the i'th element (1-based) of the supposed permutation is j.

Output

For each testcase, output exactly $floor(N/2)$ lines, one for each (i, j) pair. If the recollections till now cannot all be feasible, output '`-1`' on a line. Else output two space-separated integers: the minimum and the maximum possible value of Sauron's army, K, that are consistent with all the observations seen so far.

Between successive test cases, there should not be any blank lines in the output.

Constraints:
$1 \le T < 3$
$2 \le N \le 100,000$
$1 \le i \le floor(N/2)$
$1 \le j \le N$
All (i, j) pairs are distinct. And for two pairs (i_1, j_1) and (i_2, j_2), you have that $i_1 \ne i_2$ and $j_1 \ne j_2$.

Notes/Explanation of Sample Input:

For the first test case, for $N = 3$, there are 4 dequeue permutations, lexicographically ordered as 1-2-3, 1-3-2, 3-1-2, and 3-2-1. Denethor sees the first number of the dequeue permutation as 3, and concludes that the permutation can be either 3-1-2, or 3-2-1.

In the second test case, we see that
(a) the 1st number is 1.
(b) the 3rd number is 2 ... this means that the 2nd number has to be 32.
(c) the 2nd number is 32 ... this does not add any new information.
(d) the 4th number is 4. But this is not possible, since the 4th number can now either be 3 or 31.

Hence it is inconsistent (and none of the further observations can make it consistent). Also notice that in the 2nd test-case, the values have been output modulo 1,000,000,007.

Sample Input

```
2
3
1 3
32
1 1
3 2
2 32
4 4
5 5
6 6
7 7
8 8
9 9
10 10
11 11
12 12
13 13
14 14
15 15
16 3
```

Sample Output

```
2 3
0 73741816
536870912 805306367
536870912 805306367
-1
-1
-1
-1
-1
-1
-1
-1
-1
-1
-1
-1
-1
```

'Strange are the ways of Men, Legolas! Here they have one of the marvels of the Northern World, and what do they say of it? Caves, they say! Caves! Holes to fly to in time of war, to store fodder in! My good Legolas, do you know that the caverns of Helm's Deep are vast and beautiful? There would be an endless pilgrimage of Dwarves, merely to gaze at them, if such things were known to be. Aye indeed, they would pay pure gold for a brief glance!

'And, Legolas, when the torches are kindled and men walk on the sandy floors under the echoing domes, ah! then, Legolas, gems and crystals and veins of precious ore glint in the polished walls; and the light glows through folded marbles, shell-like, translucent as the living hands of Queen Galadriel.'

– Gimli, describing to Legolas the Glittering Caves of Aglarond.

While these caves are by and large natural, there is one place where the Men of Rohan have chiseled into the rock to create a magnificent exhibit. You have a wall of the cave consisting of 'lighted diamonds' arranged in a N by M grid (basically, you have a light behind each diamond which can be turned on or off). Further, you have a switch corresponding to each row of this diamond-grid. When you operate a switch, it will toggle (flip) the lights corresponding to that row.

You are given the current configuration of the lighted diamonds. Gimli challenges Legolas to turn on as many diamonds as possible using EXACTLY K on/off operations of the switches. Since Legolas is an Elf of the Wood and doesn't care much for things that glitter, he instead asks for your help. Note that the same switch (i.e. row) can be chosen multiple times.

Input

The first line contains the number of test cases T.

Each test case contains N, M and K on the first line followed by N lines containing M characters each. The i-th line denotes the state of the diamonds in the i-th row, where '`*`' denotes a diamond which is on and '`.`' denotes a diamond which is off.

Output

Output T lines containing the answer for the corresponding case.

Between successive test cases, there should not be any blank lines in the output.

Constraints:
$1 \leq T \leq 100$
$1 \leq N, M \leq 50$
$1 \leq K \leq 100$

Notes/Explanation of Sample Input:

In the first test case, row 1 can be toggled hence leaving all 4 lights to be in the ON state.

In the second test case, row 1 (or row 2) can be toggled twice, hence maintaining the state of the initial configuration.

Sample Input

```
2
2 2 1
..
**
2 2 2
..
**
```

Sample Output

```
4
2
```

Wormtongue looked from face to face. In his eyes was the hunted look of a beast seeking some gap in the ring of his enemies. 'Nay, Eomer, you do not fully understand the mind of Master Wormtongue,' said Gandalf, turning his piercing glance upon him. 'He is bold and cunning. Even now he plays a game with peril and wins a throw.'

– Gandalf, trying to figure out Wormtongue's mind.

In fact, Wormtongue's mind is a complicated system of evaluating various variables and parameters. In essence, each parameter is a uniform random floating point variable between 0 and 1 (inclusive). Further, his mind works on calculating best and worst-case values, which are equivalent to min/max of 2 expressions.

For example, right now Wormtongue is calculating :
'Chances of escaping' = max('Theoden letting me go', 'Me killing everyone')
'Theoden letting me go' = max('Theoden is forgiving by nature', 'Gandalf advises him to let me go').
'Me killing everyone' = min('Me killing Gandalf', 'Me killing Theoden').

So, you are given an expression consisting of independent uniform $[0, 1]$ random variables, on which you have an expression consisting of "min", and "max" alone. Help Gandalf figure out Wormtongue's mind by finding the expected value of this expression.

Input

The first line contains T, the number of test cases.

Each test case consists of a single line describing the expression. The characters of the string are derived from the set {'`M`','`m`','`x`'}, where '`M`' stands for max, '`m`' stands for min, and '`x`' is a random variable.

Formally, in the expression tree, each node which asks for max is labeled as '`M`', each node which asks for min is labelled '`m`', and all the leaves are labeled '`x`'.

The description of the expression is preorder traversal of this tree. For example, `Mxmxx` describes the expression $\max(x_1, \min(x_2, x_3))$.

Output

For each test case, output one line which contains the expected value of the expression. The results should be accurate within an error range of 10^{-6}.

Constraints:
$1 \leq T \leq 1,000$ $1 \leq$ *input string length* ≤ 100

Notes/Explanation of Sample Input:

For the first test case, it asks for the mean of a random number between 0 and 1, which is 0.5.

It is recommended to use **long long** and **long double** data types in calculation to avoid precision errors.

Sample Input

```
4
x
mxx
Mxx
MmxxMxx
```

Sample Output

```
0.500000
0.333333
0.666667
0.700000
```

'"For I am Saruman the Wise, Saruman Ring-maker, Saruman of Many Colours!" 'I looked then and saw that his robes, which had seemed white, were not so, but were woven of all colours. And if he moved they shimmered and changed hue so that the eye was bewildered.'

– Gandalf the Grey.

And so it was that Saruman decided to brand his Uruk-hai army with the many colours that he fancied. His method of branding his army was as follows.

He straps his army of N Uruk-hai onto chairs on a conveyor belt. This conveyor belt passes through his colouring-room, and can be moved forward or backward. The Uruk-hai are numbered 0 to $N-1$ according to the order in which they are seated. Saruman wishes that the i'th Uruk-hai be coloured with the colour $c[i]$.

Further, his colouring-room has space for exactly K chairs. Once the chosen K consecutive Uruk-hai are put into the room, a colour jet sprays all K of them with any fixed colour. The conveyor belt is not circular (which means that the $N-1$'th and the 0'th Uruk-hai are not consecutive).

Note that Uruk-hai can be recoloured in this process.

Saruman wants to find out what is the minimum number of times that the jet needs to be used in order to colour his army in the required fashion. If it is not possible to colour the army in the required fashion, output '`-1`'.

Input

The first line contains the number of test-cases T.

Each test case consists of 2 lines. The first line contains two space-separated integers, N and K. This is followed by a single like containing a string of length N, describing the colours of the army. The i-th character of the string denotes the colour of the i-th Uruk-hai in the army.

Output

Output T lines, one for each test case containing the answer for the corresponding test case. Remember if it is not possible to colour the army as required, output '`-1`'.

Constraints:

$1 \leq T \leq 50$

$1 \leq K \leq N \leq 20,000$

The string c has length exactly N and contains only the characters '`a`',...,'`z`'.

Notes/Explanation of Sample Input:

In the first test case, soldiers 0 and 1 can first be painted with '`r`', and then soldiers 1 and 2 can be painted with '`g`'.

In the second test case, since $N = K$, all the soldiers will only have the same color.

Sample Input

```
2
3 2
rgg
3 3
rgg
```

Sample Output

```
2
-1
```

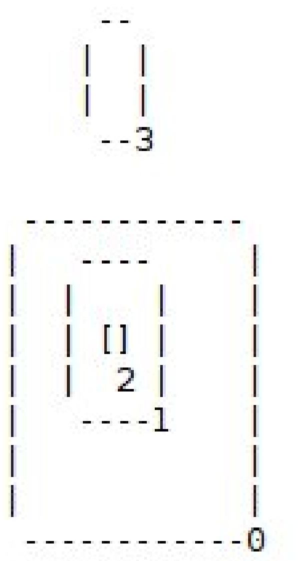
It was after nightfall when they had entered the Mines. They had been going for several hours with only brief halts, when Gandalf came to his first serious check. Before him stood a wide dark arch opening into three passages: all led in the same general direction, eastwards; but the left-hand passage plunged down, while the right-hand climbed up, and the middle way seemed to run on, smooth and level but very narrow.

– The Fellowship of the Ring are lost in the Mines.

The Mines of Moria are a true testament of Dwarvish genius. And the Fellowship of the Ring are lost in the maze of rooms, halls and caverns. You have managed to acquire a copy of the blueprints, and if only you were part of the Fellowship, Gandalf need not have had to face the Balrog!

In this problem, we consider the Mines as consisting of rectangular rooms with their sides aligned parallel to the X (West-East) and Y (South-North) axes. Some rooms are situated within other rooms. The boundaries of any two rooms have no point in common. The rooms are numbered 0 to $N-1$. Your task is to determine for each room i, which room would you enter if you exit room i. If you exit into the open, output '`-1`'.

For example, if the blueprints of the Mines looked like:



Then, you should determine that:
Room 0 exits into the open (`-1`)
Room 1 exits into Room `0`
Room 2 exits into Room `1`
Room 3 exits into the open (`-1`)

Input

The first line contains an integer N followed by N lines.

The i-th line defines the coordinates of the i-th room in the mines: $x1_i$, $y1_i$, $x2_i$, $y2_i$, where $(x1_i, y1_i)$ are the coordinates of the southwest corner and $(x2_i, y2_i)$ are the coordinates of the northeast corner of the i-th room.

Output

Output N lines, the i-th line containing the number of the room into which the i-th room exits. Output '`-1`' if the i-th room exits into the open.

Constraints:
$1 \le N \le 100{,}000$
$0 \le x1_i < x2_i \le 1{,}000{,}000{,}000$
$0 \le y1_i < y2_i \le 1{,}000{,}000{,}000$
The borders of no two rooms have any point in common.

Notes/Explanation of Sample Input:
Given in the diagram.

Sample Input

```
4
0 0 10 10
2 3 7 8
3 4 5 6
12 10 13 15
```

Sample Output

```
-1
0
1
-1
```

'I think that the enemy brought his own enemy with him,' answered Aragorn. 'These are Northern Orcs from far away. Among the slain are none of the great Orcs with the strange badges. There was a quarrel, I guess: it is no uncommon thing with these foul folk. Maybe there was some dispute about the road.'

– Aragorn describing the nature of Orcs.

Indeed, everyone knows that the Orcs are treacherous creatures who look for their own satisfaction and more often than not disregard the rules. The only way to keep them in line, is by maintaining the chain of command over a strict hierarchy among the ranks, wherein each Orc is responsible to his immediate superior all the way up to the army's head.

Further, the powers that be, have decided to have regular checks of their army's loyalties, just in case some Orc has been killed and all his juniors end up turning rogue!

There are N orcs, numbered 1 to N, wherein the lead orc is numbered 1.

Step 1. Randomly choose a fixed order in which to test Orcs' loyalties.
Step 2. Going in this order, you make a "roll-call" to check if the current Orc is alive or not.
Step 3. If the current Orc is dead, then he is marked as "deleted".

With this information, it is possible to tell which all Orcs will be loyal, and which won't be. However, cunning Master Wormtongue suggests the following optimization:

In step 2, if any of the considered Orc's superiors (not necessarily immediate superior) is marked as deleted, then the roll-call is not made.

Now, given this algorithm and the hierarchy of the army, along with which Orcs are dead, what is the expected number of roll-calls (taken over all possible orderings in "Step 1") that you save by performing this optimization?

Input

The first line contains T, the number of test cases.

The first line of each test case contains N, the number of orcs in the army. The next $N-1$ lines contain two space-separated integers u v, denoting that u is the immediate superior of v or vice-versa. The head of the army is the orc labelled 1. The next line contains m, the number of dead orcs. The next line contains m space separated integers, which are the labels of the dead orcs.

Output

Output one real number for each test case containing the expected number of roll-calls that you save. The results should be accurate within an error range of 10^{-6}.

Constraints:
$1 \leq T \leq 5$
$1 \leq N \leq 100,000$
$1 \leq u, v \leq N$
$u \neq v$
The given set of u v pairs form a valid chain of command. That means every Orc, except the Orc labeled 1, has exactly one immediate superior.

Notes/Explanation of Sample Input:

For the first test case, the Orc labelled 1 is dead. The two possible orderings are [1, 2] and [2, 1]. With the optimization, for the order [1, 2], we save the roll-call to 2. So, the total number of roll-calls without the optimization is 4, and with the optimization is 3. Expected number of roll-calls is therefore, (4 - 3) / 2 = 0.5 .

For the second test case, the Orc labeled 2 is dead. Since he does not have any sub-ordinates, the optimization does not have any effect.

Sample Input

```
2
2
1 2
1
1
2
1 2
1
2
```

Sample Output

```
0.5
0
```