Codeforces Beta Round #69 (Div. 1 Only)

A. Heroes

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

The year of 2012 is coming...

According to an ancient choradrican legend in this very year, in 2012, Diablo and his brothers Mephisto and Baal will escape from hell, and innumerable hordes of demons will enslave the human world. But seven brave heroes have already gathered on the top of a mountain Arreat to protect us mere mortals from the effect of this terrible evil.

The seven great heroes are: amazon `Anka`, barbarian `Chapay`, sorceress `Cleo`, druid `Troll`, necromancer `Dracul`, paladin `Snowy` and a professional hit girl `Hexadecimal`. Heroes already know how much experience will be given for each of the three megabosses: a for Mephisto, b for Diablo and c for Baal.

Here's the problem: heroes are as much as seven and megabosses are only three! Then our heroes decided to split into three teams, where each team will go to destroy their own megaboss. Each team member will receive a $\frac{x}{y}$ of experience, rounded down, where x will be the amount of experience for the killed megaboss and y — the number of people in the team.

Heroes do not want to hurt each other's feelings, so they want to split into teams so that the difference between the hero who received the maximum number of experience and the hero who received the minimum number of experience were minimal. Since there can be several divisions into teams, then you need to find the one in which *the total amount of liking in teams* were maximum.

It is known that some heroes like others. But if hero p likes hero q, this does not mean that the hero q likes hero p. No hero likes himself.

The total amount of liking in teams is the amount of ordered pairs (p, q), such that heroes p and q are in the same group, and hero p likes hero q (but it is not important if hero q likes hero p). In case of heroes p and q likes each other and they are in the same group, this pair should be counted twice, as (p, q) and (q, p).

A team can consist even of a single hero, but it is important that every megaboss was destroyed. All heroes must be involved in the campaign against evil. None of the heroes can be in more than one team.

It is guaranteed that every hero is able to destroy any megaboss alone.

Input

The first line contains a single non-negative integer n ($0 \le n \le 42$) — amount of liking between the heroes. Next n lines describe liking in the form "`p likes q`", meaning that the hero `p` likes the hero `q` (`p` $\neq$ `q`). Every liking is described in the input exactly once, no hero likes himself.

In the last line are given three integers a, b and c ($1 \le a, b, c \le 2 \cdot 10^9$), separated by spaces: the experience for Mephisto, the experience for Diablo and experience for Baal.

In all the pretests, except for examples from the statement, the following condition is satisfied: $a = b = c$.

Output

Print two integers — the minimal difference in the experience between two heroes who will receive the maximum and minimum number of experience points, and the maximal total amount of liking in teams (the number of friendships between heroes that end up in one team).

When calculating the second answer, the team division should satisfy the difference-minimizing contraint. I.e. primary you should minimize the difference in the experience and secondary you should maximize the total amount of liking.

Sample test(s)

input

```
3
Troll likes Dracul
Dracul likes Anka
Snowy likes Hexadecimal
210 200 180
```

output

```
30 3
```

input

```
2
Anka likes Chapay
Chapay likes Anka
10000 50 50
```

output
1950 2

Note

A note to first example: it the first team should be `Dracul`, `Troll` and `Anka`, in the second one `Hexadecimal` and `Snowy`, and in the third `Cleo` и `Chapay`.

B. Falling Anvils

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

For some reason in many American cartoons anvils fall from time to time onto heroes' heads. Of course, safes, wardrobes, cruisers, planes fall sometimes too... But anvils do so most of all.

Anvils come in different sizes and shapes. Quite often they get the hero stuck deep in the ground. But have you ever thought who throws anvils from the sky? From what height? We are sure that such questions have never troubled you!

It turns out that throwing an anvil properly is not an easy task at all. Let's describe one of the most popular anvil throwing models.

Let the height p of the potential victim vary in the range $[0;a]$ and the direction of the wind q vary in the range $[-b;b]$. **p and q could be any real (floating) numbers.** Then we can assume that the anvil will fit the toon's head perfectly only if the following equation has at least one real root:

$$x^2 + \sqrt{p} \cdot x + q = 0$$

Determine the probability with which an aim can be successfully hit by an anvil.

You can assume that the p and q coefficients are chosen equiprobably and independently in their ranges.

Input

The first line contains integer t ($1 \le t \le 10000$) — amount of testcases.

Each of the following t lines contain two space-separated integers a and b ($0 \le a, b \le 10^6$).

Pretests contain all the tests with $0 < a < 10, 0 \le b < 10$.

Output

Print t lines — the probability of a successful anvil hit for each testcase. The absolute or relative error of the answer should not exceed 10^{-6}.

Sample test(s)

input
2 4 2 1 2

output
0.6250000000 0.5312500000

C. Beavermuncher-0xFF

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

"Eat a beaver, save a tree!" — That will be the motto of ecologists' urgent meeting in Beaverley Hills.

And the whole point is that the population of beavers on the Earth has reached incredible sizes! Each day their number increases in several times and they don't even realize how much their unhealthy obsession with trees harms the nature and the humankind. The amount of oxygen in the atmosphere has dropped to 17 per cent and, as the best minds of the world think, that is not the end.

In the middle of the 50-s of the previous century a group of soviet scientists succeed in foreseeing the situation with beavers and worked out a secret technology to clean territory. The technology bears a mysterious title "Beavermuncher-0xFF". Now the fate of the planet lies on the fragile shoulders of a small group of people who has dedicated their lives to science.

The prototype is ready, you now need to urgently carry out its experiments in practice.

You are given a tree, completely occupied by beavers. A tree is a connected undirected graph without cycles. The tree consists of n vertices, the i-th vertex contains k_i beavers.

"Beavermuncher-0xFF" works by the following principle: being at some vertex u, it can go to the vertex v, if they are connected by an edge, and eat **exactly one** beaver located at the vertex v. It is impossible to move to the vertex v if there are no beavers left in v. "Beavermuncher-0xFF" **cannot** just stand at some vertex and eat beavers in it. "Beavermuncher-0xFF" must move without stops.

Why does the "Beavermuncher-0xFF" works like this? Because the developers have not provided place for the battery in it and eating beavers is necessary for converting their mass into pure energy.

It is guaranteed that the beavers will be shocked by what is happening, which is why they will not be able to move from a vertex of the tree to another one. As for the "Beavermuncher-0xFF", it can move along each edge in both directions while conditions described above are fulfilled.

The root of the tree is located at the vertex s. This means that the "Beavermuncher-0xFF" begins its mission at the vertex s and it must return there at the end of experiment, because no one is going to take it down from a high place.

Determine the maximum number of beavers "Beavermuncher-0xFF" can eat and return to the starting vertex.

Input

The first line contains integer n — the number of vertices in the tree ($1 \le n \le 10^5$). The second line contains n integers k_i ($1 \le k_i \le 10^5$) — amounts of beavers on corresponding vertices. Following $n - 1$ lines describe the tree. Each line contains two integers separated by space. These integers represent two vertices connected by an edge. Vertices are numbered from 1 to n. The last line contains integer s — the number of the starting vertex ($1 \le s \le n$).

Output

Print the maximum number of beavers munched by the "Beavermuncher-0xFF".

Please, do not use `%lld` specificator to write 64-bit integers in C++. It is preferred to use `cout` (also you may use `%I64d`).

Sample test(s)

input

```
5
1 3 1 3 2
2 5
3 4
4 5
1 5
4
```

output

```
6
```

input

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

output

```
2
```

D. Domino Carpet

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

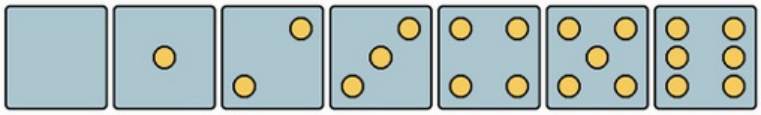
...Mike the TV greets you again!

Tired of the monotonous furniture? Sick of gray routine? Dreaming about dizzying changes in your humble abode? We have something to offer you!

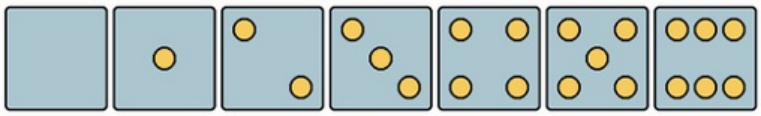
This domino carpet for only $99.99 will change your life! You can lay it on the floor, hang it on the wall or even on the ceiling! Among other things ...

Having watched the commercial, virus Hexadecimal also wanted to get a Domino Carpet and wanted badly to be photographed in front of it. But of course, a virus will never consent to buying a licensed Carpet! So she ordered a truck of dominoes and decided to make such a Carpet herself.

The original Domino Carpet is a field of squares $n \times m$ in size. Each square is half of a domino, and can be rotated either vertically or horizontally, independently from its neighbors. Vertically rotated domino halves look like this:



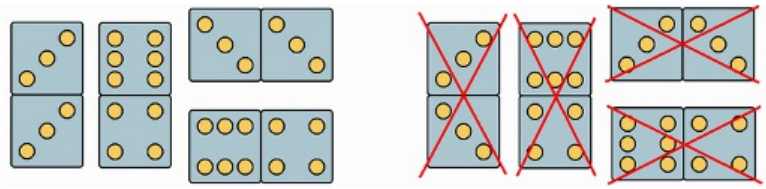
And horizontally rotated halves look like this:



Notice, that some halves looks the same in both rotations, but other halves differ.

Dominoes bought by Hexadecimal are represented by uncuttable chips 1×2 in size, which can be laid either vertically or horizontally. If the chip is laid vertically, then both of it's halves should be laid vertically orientated; if the chip is laid horizontally, then both of it's halves should be laid horizontally.

The samples of valid and invalid dominoes laid vertically and horizontally are:



Virus Hexadecimal assembles her own Domino Carpet so that the following conditions are satisfied:

- each carpet square is covered by a domino chip, i.e. there are no empty squares;
- all domino chips lie entirely within the carpet and don't overlap with each other;
- if there is a horizontal domino chip with its left half in column j then there are no horizontal domino chips with their left halves in columns j - 1 or $j + 1$.

Before starting to assemble her own Domino Carpet, the virus wants to know the number of ways to achieve the intended purpose modulo $10^9 + 7$.

You can assume that the virus has an infinitely large number of dominoes of each type.

Input

The first line contains two integers n and m, separated by a space — the size of the Domino Carpet ($1 \le n, m \le 250$). Next $4n + 1$ lines contain $4m + 1$ symbols.

Each square of the Domino Carpet, which is a domino half, is described by a 3×3 square. Symbol `'o'` in this square indicates the presence of a point, symbol `'.'` — its absence.

Each 3×3 square is delineated from adjacent squares by symbols `'#'` as shown in the examples.

It is guaranteed that every box describes the correct half of a domino.

In all pretests the Domino Carpets have the size of 2×2 and 4×4.

Output

Print a single number, the number of ways to assemble the Domino Carpet modulo $10^9 + 7$, using only standard dominoes of size 1×2.

Sample test(s)

input
3 4 ################# #o..#...#o.o#...# #.o.#.o.#.o.#...#

```
#..0#...#0.0#...#
#################
#0.0#000#0.0#...#
#.0.#...#...#.0.#
#0.0#000#0.0#...#
#################
#0.0#...#0.0#...#
#...#...#...#.0.#
#0.0#...#0.0#...#
#################
```

output

```
3
```

input

```
2 2
#########
#0.0#0.0#
#.0.#...#
#0.0#0.0#
#########
#...#0.0#
#...#...#
#...#0.0#
#########
```

output

```
2
```

input

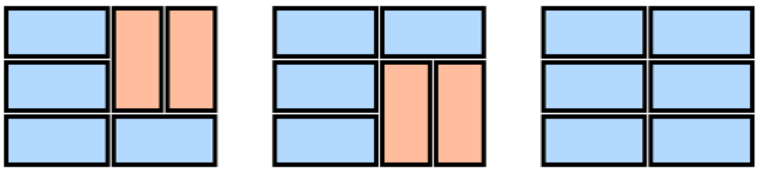
```
2 2
#########
#..0#0..#
#...#...#
#0..#..0#
#########
#0..#..0#
#...#...#
#..0#0..#
#########
```

output

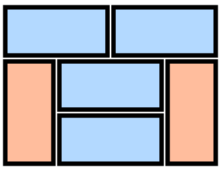
```
0
```

Note

A note to the first example: all correct ways to make Domino Carpet are represented below:



And this way is incorrect:



E. Martian Food

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Have you ever tasted Martian food? Well, you should.

Their signature dish is served on a completely black plate with the radius of R, flat as a pancake.

First, they put a perfectly circular portion of the Golden Honduras on the plate. It has the radius of r and is located as close to the edge of the plate as possible staying entirely within the plate. I. e. Golden Honduras touches the edge of the plate from the inside. It is believed that the proximity of the portion of the Golden Honduras to the edge of a plate demonstrates the neatness and exactness of the Martians.

Then a perfectly round portion of Pink Guadeloupe is put on the plate. The Guadeloupe should not overlap with Honduras, should not go beyond the border of the plate, but should have the maximum radius. I. e. Pink Guadeloupe should touch the edge of the plate from the inside, and touch Golden Honduras from the outside. For it is the size of the Rose Guadeloupe that shows the generosity and the hospitality of the Martians.

Further, the first portion (of the same perfectly round shape) of Green Bull Terrier is put on the plate. It should come in contact with Honduras and Guadeloupe, should not go beyond the border of the plate and should have maximum radius.

Each of the following portions of the Green Bull Terrier must necessarily touch the Golden Honduras, the previous portion of the Green Bull Terrier and touch the edge of a plate, but should not go beyond the border.

To determine whether a stranger is worthy to touch the food, the Martians ask him to find the radius of the k-th portion of the Green Bull Terrier knowing the radii of a plate and a portion of the Golden Honduras. And are you worthy?

Input

The first line contains integer t ($1 \le t \le 10^4$) — amount of testcases.

Each of the following t lines contain three positive integers: the radii of the plate and a portion of the Golden Honduras R and r ($1 \le r < R \le 10^4$) and the number k ($1 \le k \le 10^4$).

In the pretests $1 \le k \le 2$.

Output

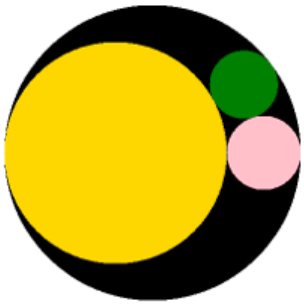
Print t lines — the radius of the k-th portion of the Green Bull Terrier for each test. The absolute or relative error of the answer should not exceed 10^{-6}.

Sample test(s)

input
2 4 3 1 4 2 2
output
0.9230769231 0.6666666667

Note

Dish from the first sample looks like this:



Dish from the second sample looks like this: