Codeforces Beta Round #69 (Div. 2 Only)

A. Panoramix's Prediction

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

A prime number is a number which has exactly two distinct divisors: one and itself. For example, numbers 2, 7, 3 are prime, and 1, 6, 4 are not.

*The next prime number after x is the **smallest** prime number greater than x. For example, the next prime number after 2 is 3, and the next prime number after 3 is 5. Note that there is exactly one next prime number after each number. So 5 **is not** the next prime number for 2.*

One cold April morning Panoramix predicted that soon Kakofonix will break free from his straitjacket, and this will be a black day for the residents of the Gallic countryside.

Panoramix's prophecy tells that if some day Asterix and Obelix beat exactly x Roman soldiers, where x is a prime number, and next day they beat exactly y Roman soldiers, where y is **the next prime number** after x, then it's time to wait for Armageddon, for nothing can shut Kakofonix up while he sings his infernal song.

Yesterday the Gauls beat n Roman soldiers and it turned out that the number n was prime! Today their victims were a troop of m Romans ($m > n$). Determine whether the Gauls should wait for the black day after today's victory of Asterix and Obelix?

Input

The first and only input line contains two positive integers — n and m ($2 \le n < m \le 50$). It is guaranteed that n is prime.

Pretests contain all the cases with restrictions $2 \le n < m \le 4$.

Output

Print `YES`, if m is the next prime number after n, or `NO` otherwise.

Sample test(s)

input
3 5
output
YES

input
7 11
output
YES

input
7 9
output
NO

B. Depression

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

Do you remember a kind cartoon "Beauty and the Beast"? No, no, there was no firing from machine guns or radiation mutants time-travels!

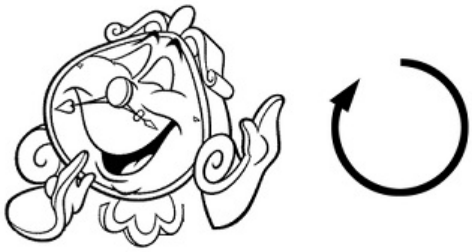
There was a beauty named Belle. Once she had violated the Beast's order and visited the West Wing. After that she was banished from the castle...

Everybody was upset. The beautiful Belle was upset, so was the Beast, so was Lumiere the candlestick. But the worst thing was that Cogsworth was upset. Cogsworth is not a human, but is the mantel clock, which was often used as an alarm clock.

Due to Cogsworth's frustration all the inhabitants of the castle were in trouble: now they could not determine when it was time to drink morning tea, and when it was time for an evening stroll.

Fortunately, deep in the basement are lying digital clock showing the time in the format `HH:MM`. Now the residents of the castle face a difficult task. They should turn Cogsworth's hour and minute mustache hands in such a way, that Cogsworth began to show the correct time. Moreover they need to find turn angles in degrees for each mustache hands. The initial time showed by Cogsworth is `12:00`.

You can only rotate the hands forward, that is, as is shown in the picture:



As since there are many ways too select such angles because of full rotations, choose the smallest angles in the right (non-negative) direction.

Note that Cogsworth's hour and minute mustache hands move evenly and continuously. Hands are moving independently, so when turning one hand the other hand remains standing still.

Input

The only line of input contains current time according to the digital clock, formatted as `HH:MM` ($00 \le \texttt{HH} \le 23$, $00 \le \texttt{MM} \le 59$). The mantel clock initially shows `12:00`.

Pretests contain times of the beginning of some morning TV programs of the Channel One Russia.

Output

Print two numbers x and y — the angles of turning the hour and minute hands, respectively ($0 \le x, y < 360$). The absolute or relative error in the answer should not exceed 10^{-9}.

Sample test(s)

input
12:00
output
0 0

input
04:30
output
135 180

input
08:17
output
248.5 102

Note

A note to the second example: the hour hand will be positioned exactly in the middle, between 4 and 5.

C. 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`.

D. 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

E. 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