A. Far Relative’s Birthday Cake

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

Door's family is going celebrate Famil Doors's birthday party. They love Famil Door so they are planning to make his birthday cake weird!

The cake is a $n \times n$ square consisting of equal squares with side length 1. Each square is either empty or consists of a single chocolate. They bought the cake and randomly started to put the chocolates on the cake. The value of Famil Door's happiness will be equal to the number of pairs of cells with chocolates that are in the same row or in the same column of the cake. Famil Doors's family is wondering what is the amount of happiness of Famil going to be?

Please, note that any pair can be counted no more than once, as two different cells can't share both the same row and the same column.

Input

In the first line of the input, you are given a single integer n ($1 \le n \le 100$) — the length of the side of the cake.

Then follow n lines, each containing n characters. Empty cells are denoted with '.', while cells that contain chocolates are denoted by 'C'.

Output

Print the value of Famil Door's happiness, i.e. the number of pairs of chocolate pieces that share the same row or the same column.

Examples

input

```
3
.CC
C..
C.C
```

output

```
4
```

input

```
4
CC..
C..C
.CC.
.CC.
```

output

```
9
```

Note

If we number rows from top to bottom and columns from left to right, then, pieces that share the same row in the first sample are:

1. $(1, 2)$ and $(1, 3)$
2. $(3, 1)$ and $(3, 3)$

Pieces that share the same column are:

1. $(2, 1)$ and $(3, 1)$
2. $(1, 3)$ and $(3, 3)$

B. Far Relative’s Problem

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

Famil Door wants to celebrate his birthday with his friends from Far Far Away. He has n friends and each of them can come to the party in a specific range of days of the year from a_i to b_i. Of course, Famil Door wants to have as many friends celebrating together with him as possible.

Far cars are as weird as Far Far Away citizens, so they can only carry two people of opposite gender, that is exactly one male and one female. However, Far is so far from here that no other transportation may be used to get to the party.

Famil Door should select some day of the year and invite some of his friends, such that they all are available at this moment and the number of male friends invited is equal to the number of female friends invited. Find the maximum number of friends that may present at the party.

Input

The first line of the input contains a single integer n ($1 \le n \le 5000$) — then number of Famil Door's friends.

Then follow n lines, that describe the friends. Each line starts with a capital letter 'F' for female friends and with a capital letter 'M' for male friends. Then follow two integers a_i and b_i ($1 \le a_i \le b_i \le 366$), providing that the i-th friend can come to the party from day a_i to day b_i inclusive.

Output

Print the maximum number of people that may come to Famil Door's party.

Examples

input

```
4
M 151 307
F 343 352
F 117 145
M 24 128
```

output

```
2
```

input

```
6
M 128 130
F 128 131
F 131 140
F 131 141
M 131 200
M 140 200
```

output

```
4
```

Note

In the first sample, friends 3 and 4 can come on any day in range $[117, 128]$.

In the second sample, friends with indices 3, 4, 5 and 6 can come on day 140.

C. Famil Door and Brackets

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

As Famil Door’s birthday is coming, some of his friends (like Gabi) decided to buy a present for him. His friends are going to buy a string consisted of round brackets since Famil Door loves string of brackets of length n more than any other strings!

The sequence of round brackets is called *valid* if and only if:

1. the total number of opening brackets is equal to the total number of closing brackets;
2. for any prefix of the sequence, the number of opening brackets is greater or equal than the number of closing brackets.

Gabi bought a string s of length m ($m \le n$) and want to complete it to obtain a valid sequence of brackets of length n. He is going to pick some strings p and q consisting of round brackets and merge them in a string $p + s + q$, that is add the string p at the beginning of the string s and string q at the end of the string s.

Now he wonders, how many **pairs** of strings p and q exists, such that the string $p + s + q$ is a valid sequence of round brackets. As this number may be pretty large, he wants to calculate it modulo $10^9 + 7$.

Input

First line contains n and m ($1 \le m \le n \le 100\,000, n - m \le 2000$) — the desired length of the string and the length of the string bought by Gabi, respectively.

The second line contains string s of length m consisting of characters '(' and ')' only.

Output

Print the number of pairs of string p and q such that $p + s + q$ is a valid sequence of round brackets modulo $10^9 + 7$.

Examples

input

```
4 1
(
```

output

```
4
```

input

```
4 4
(())
```

output

```
1
```

input

```
4 3
(((
```

output

```
0
```

Note

In the first sample there are four different valid pairs:

1. p = "(", q = "))"
2. p = "()", q = ")"
3. p = "", q = "())"
4. p = "", q = ")())"

In the second sample the only way to obtain a desired string is choose empty p and q.

In the third sample there is no way to get a valid sequence of brackets.

D. Babaei and Birthday Cake

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

As you know, every birthday party has a cake! This time, Babaei is going to prepare the very special birthday party's cake.

Simple cake is a cylinder of some radius and height. The volume of the simple cake is equal to the volume of corresponding cylinder. Babaei has n simple cakes and he is going to make a *special cake* placing some cylinders on each other.

However, there are some additional culinary restrictions. The cakes are numbered in such a way that the cake number i can be placed only on the table or on some cake number j where $j < i$. Moreover, in order to impress friends Babaei will put the cake i on top of the cake j only if the volume of the cake i is strictly greater than the volume of the cake j.

Babaei wants to prepare a birthday cake that has a maximum possible total volume. Help him find this value.

Input

The first line of the input contains a single integer n $(1 \le n \le 100\,000)$ — the number of simple cakes Babaei has.

Each of the following n lines contains two integers r_i and h_i ($1 \le r_i, h_i \le 10\,000$), giving the radius and height of the i-th cake.

Output

Print the maximum volume of the cake that Babaei can make. Your answer will be considered correct if its absolute or relative error does not exceed 10^{-6}.

Namely: let's assume that your answer is a, and the answer of the jury is b. The checker program will consider your answer correct, if $\frac{|a-b|}{max(1,b)} \le 10^{-6}$.

Examples

input

```
2
100 30
40 10
```

output

```
942477.796077000
```

input

```
4
1 1
9 7
1 4
10 7
```

output

```
3983.539484752
```

Note

In first sample, the optimal way is to choose the cake number 1.

In second sample, the way to get the maximum volume is to use cakes with indices 1, 2 and 4.

E. Famil Door and Roads

time limit per test: 5 seconds
memory limit per test: 512 megabytes
input: standard input
output: standard output

Famil Door's City map looks like a tree (undirected connected acyclic graph) so other people call it Treeland. There are n intersections in the city connected by $n - 1$ bidirectional roads.

There are m friends of Famil Door living in the city. The i-th friend lives at the intersection u_i and works at the intersection v_i. Everyone in the city is unhappy because there is exactly one simple path between their home and work.

Famil Door plans to construct exactly one new road and he will randomly choose one among $n \cdot (n - 1) / 2$ possibilities. Note, that he may even build a new road between two cities that are already connected by one.

He knows, that each of his friends will become happy, if after Famil Door constructs a new road there is a path from this friend home to work and back that doesn't visit the same road twice. Formally, there is a simple cycle containing both u_i and v_i.

Moreover, if the friend becomes happy, his pleasure is equal to the length of such path (it's easy to see that it's unique). For each of his friends Famil Door wants to know his expected pleasure, that is the expected length of the cycle containing both u_i and v_i if we consider only cases when such a cycle exists.

Input

The first line of the input contains integers n and m ($2 \le n, \ m \le 100\,000$) — the number of the intersections in the Treeland and the number of Famil Door's friends.

Then follow $n - 1$ lines describing bidirectional roads. Each of them contains two integers a_i and b_i ($1 \le a_i, b_i \le n$) — the indices of intersections connected by the i-th road.

Last m lines of the input describe Famil Door's friends. The i-th of these lines contain two integers u_i and v_i ($1 \le u_i, v_i \le n, u_i \ne v_i$) — indices of intersections where the i-th friend lives and works.

Output

For each friend you should print the expected value of pleasure if he will be happy. Your answer will be considered correct if its absolute or relative error does not exceed 10^{-6}.

Namely: let's assume that your answer is a, and the answer of the jury is b. The checker program will consider your answer correct, if $\frac{|a-b|}{\max(1,b)} \le 10^{-6}$.

Examples

input

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

output

```
4.00000000
3.00000000
3.00000000
```

input

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

output

```
2.50000000
2.50000000
3.00000000
```

Note

Consider the second sample.

1. Both roads $(1, 2)$ and $(2, 3)$ work, so the expected length if $\frac{2+3}{2} = 2.5$
2. Roads $(1, 3)$ and $(2, 3)$ make the second friend happy. Same as for friend 1 the answer is 2.5
3. The only way to make the third friend happy is to add road $(2, 3)$, so the answer is 3