15-295 Fall 2018 #2

A. QAQ

1 second, 256 megabytes

"QAQ" is a word to denote an expression of crying. Imagine "Q" as eyes with tears and "A" as a mouth.

Now Diamond has given Bort a string consisting of only uppercase English letters of length n. There is a great number of "QAQ" in the string (Diamond is so cute!).

illustration by 猫屋 https://twitter.com/nekoyaliu

Bort wants to know how many subsequences "QAQ" are in the string Diamond has given. Note that the letters "QAQ" don't have to be consecutive, but the order of letters should be exact.

Input

The only line contains a string of length n ($1 \le n \le 100$). It's guaranteed that the string only contains uppercase English letters.

Output

Print a single integer — the number of subsequences "QAQ" in the string.

input
QAQAQYSYIOIWIN
output
4

input
QAQQQZZYNOIWIN
output
3

In the first example there are 4 subsequences "QAQ": "**QAQ**AQYSYIOIWIN", "**QA**QA**Q**YSYIOIWIN", "**Q**AQ**AQ**YSYIOIWIN", "QA**QAQ**YSYIOIWIN".

B. Woodcutters

1 second, 256 megabytes

Little Susie listens to fairy tales before bed every day. Today's fairy tale was about wood cutters and the little girl immediately started imagining the choppers cutting wood. She imagined the situation that is described below.

There are n trees located along the road at points with coordinates $x_1, x_2, ..., x_n$. Each tree has its height h_i.

Woodcutters can cut down a tree and fell it to the left or to the right. After that it occupies one of the segments $[x_i - h_i, x_i]$ or $[x_i;x_i + h_i]$. The tree that is not cut down occupies a single point with coordinate x_i. Woodcutters can fell a tree if the segment to be occupied by the fallen tree doesn't contain any occupied point. The woodcutters want to process as many trees as possible, so Susie wonders, what is the maximum number of trees to fell.

Input

The first line contains integer n $(1 \le n \le 10^5)$ — the number of trees.

Next n lines contain pairs of integers x_i, h_i $(1 \le x_i, h_i \le 10^9)$ — the coordinate and the height of the i-th tree.

The pairs are given in the order of ascending x_i. No two trees are located at the point with the same coordinate.

Output

Print a single number — the maximum number of trees that you can cut down by the given rules.

input

```
5
1 2
2 1
5 10
10 9
19 1
```

output

```
3
```

input

```
5
1 2
2 1
5 10
10 9
20 1
```

output

```
4
```

In the first sample you can fell the trees like that:

- fell the 1-st tree to the left — now it occupies segment $[-1;1]$
- fell the 2-nd tree to the right — now it occupies segment $[2;3]$
- leave the 3-rd tree — it occupies point 5
- leave the 4-th tree — it occupies point 10
- fell the 5-th tree to the right — now it occupies segment $[19;20]$

In the second sample you can also fell 4-th tree to the right, after that it will occupy segment $[10;19]$.

C. Zuma

2 seconds, 512 megabytes

Genos recently installed the game Zuma on his phone. In Zuma there exists a line of n gemstones, the i-th of which has color c_i. The goal of the game is to destroy all the gemstones in the line as quickly as possible.

In one second, Genos is able to choose exactly one continuous substring of colored gemstones that is a palindrome and remove it from the line. After the substring is removed, the remaining gemstones shift to form a solid line again. What is the minimum number of seconds needed to destroy the entire line?

Let us remind, that the string (or substring) is called *palindrome*, if it reads same backwards or forward. In our case this means the color of the first gemstone is equal to the color of the last one, the color of the second gemstone is equal to the color of the next to last and so on.

Input

The first line of input contains a single integer n $(1 \le n \le 500)$ — the number of gemstones.

The second line contains n space-separated integers, the i-th of which is c_i $(1 \le c_i \le n)$ — the color of the i-th gemstone in a line.

Output

Print a single integer — the minimum number of seconds needed to destroy the entire line.

input
```
3
1 2 1
```
output
```
1
```

input
```
3
1 2 3
```
output
```
3
```

input
```
7
1 4 4 2 3 2 1
```
output
```
2
```

In the first sample, Genos can destroy the entire line in one second.

In the second sample, Genos can only destroy one gemstone at a time, so destroying three gemstones takes three seconds.

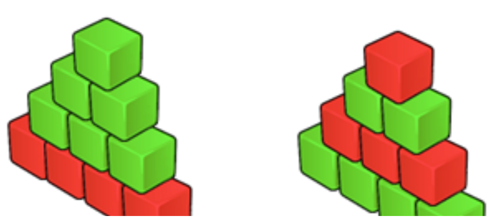
In the third sample, to achieve the optimal time of two seconds, destroy palindrome `4 4` first and then destroy palindrome `1 2 3 2 1`.

D. Red-Green Towers

2 seconds, 256 megabytes

There are r red and g green blocks for construction of the *red-green* tower. *Red-green* tower can be built following next rules:

- Red-green tower is consisting of some number of levels;
- Let the red-green tower consist of n levels, then the first level of this tower should consist of n blocks, second level — of $n - 1$ blocks, the third one — of $n - 2$ blocks, and so on — the last level of such tower should consist of the one block. In other words, each successive level should contain one block less than the previous one;
- Each level of the red-green tower should contain blocks of the same color.



Let h be the maximum possible number of levels of red-green tower, that can be built out of r red and g green blocks meeting the rules above. The task is to determine how many different red-green towers having h levels can be built out of the available blocks.

Two red-green towers are considered different if there exists some level, that consists of red blocks in the one tower and consists of green blocks in the other tower.

You are to write a program that will find the number of different red-green towers of height h modulo $10^9 + 7$.

Input

The only line of input contains two integers r and g, separated by a single space — the number of available red and green blocks respectively ($0 \le r, g \le 2 \cdot 10^5$, $r + g \ge 1$).

Output

Output the only integer — the number of different possible red-green towers of height h modulo $10^9 + 7$.

input
4 6
output
2

input
9 7
output
6

input
1 1
output
2

The image in the problem statement shows all possible red-green towers for the first sample.

E. Wilbur and Trees

2 seconds, 256 megabytes

Wilbur the pig really wants to be a beaver, so he decided today to pretend he is a beaver and bite at trees to cut them down.

There are n trees located at various positions on a line. Tree i is located at position x_i. All the given positions of the trees are distinct.

The trees are equal, i.e. each tree has height h. Due to the wind, when a tree is cut down, it either falls left with probability p, or falls right with probability $1 - p$. If a tree hits another tree while falling, that tree will fall in the same direction as the tree that hit it. A tree can hit another tree only if the distance between them is strictly less than h.

For example, imagine there are 4 trees located at positions 1, 3, 5 and 8, while $h = 3$ and the tree at position 1 falls right. It hits the tree at position 3 and it starts to fall too. In it's turn it hits the tree at position 5 and it also starts to fall. The distance between 8 and 5 is exactly 3, so the tree at position 8 will not fall.

As long as there are still trees standing, Wilbur will select either the leftmost standing tree with probability 0.5 or the rightmost standing tree with probability 0.5. Selected tree is then cut down. If there is only one tree remaining, Wilbur always selects it. As the ground is covered with grass, Wilbur wants to know the expected total length of the ground covered with fallen trees after he cuts them all down because he is concerned about his grass-eating cow friends. Please help Wilbur.

Input

The first line of the input contains two integers, n $(1 \le n \le 2000)$ and h $(1 \le h \le 10^8)$ and a real number p $(0 \le p \le 1)$, given with no more than six decimal places.

The second line of the input contains n integers, $x_1, x_2, ..., x_n$ $(-10^8 \le x_i \le 10^8)$ in no particular order.

Output

Print a single real number — the expected total length of the ground covered by trees when they have all fallen down. 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 $\frac{|a-b|}{\max(1,b)} \le 10^{-6}$.

input

```
2 2 0.500000
1 2
```

output

```
3.250000000
```

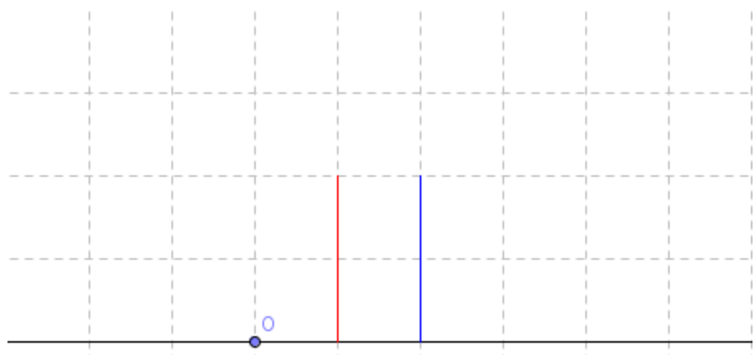
input

```
4 3 0.4
4 3 1 2
```

output

```
6.631200000
```

Consider the first example, we have 2 trees with height 2.



There are 3 scenarios:

1. Both trees falls left. This can either happen with the right tree falling left first, which has $\frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4}$ probability (also knocking down the left tree), or the left tree can fall left and then the right tree can fall left, which has $\frac{1}{4} \cdot \frac{1}{2} = \frac{1}{8}$ probability. Total probability is $\frac{1}{4} + \frac{1}{8} = \frac{3}{8}$.

2. Both trees fall right. This is analogous to (1), so the probability of this happening is $\frac{3}{8}$.

3. The left tree fall left and the right tree falls right. This is the only remaining scenario so it must have $1 - \frac{3}{8} - \frac{3}{8} = \frac{1}{4}$ probability.

Cases 1 and 2 lead to a total of 3 units of ground covered, while case 3 leads to a total of 4 units of ground covered. Thus, the expected value is $3 \cdot (\frac{3}{8}) + 3 \cdot (\frac{3}{8}) + 4 \cdot (\frac{1}{4}) = 3.25$.

F. Painting The Wall

1 second, 256 megabytes

User ainta decided to paint a wall. The wall consists of n^2 tiles, that are arranged in an $n \times n$ table. Some tiles are painted, and the others are not. As he wants to paint it beautifully, he will follow the rules below.

1. Firstly user ainta looks at the wall. If there is at least one painted cell on each row and at least one painted cell on each column, he stops coloring. Otherwise, he goes to step 2.
2. User ainta choose any tile on the wall with uniform probability.
3. If the tile he has chosen is not painted, he paints the tile. Otherwise, he ignores it.
4. Then he takes a rest for one minute even if he doesn't paint the tile. And then ainta goes to step 1.

However ainta is worried if it would take too much time to finish this work. So he wants to calculate the expected time needed to paint the wall by the method above. Help him find the expected time. You can assume that choosing and painting any tile consumes no time at all.

Input

The first line contains two integers n and m $(1 \le n \le 2 \cdot 10^3; 0 \le m \le min(n^2, 2 \cdot 10^4))$ — the size of the wall and the number of painted cells.

Next m lines goes, each contains two integers r_i and c_i $(1 \le r_i, c_i \le n)$ — the position of the painted cell. It is guaranteed that the positions are all distinct. Consider the rows of the table are numbered from 1 to n. Consider the columns of the table are numbered from 1 to n.

Output

In a single line print the expected time to paint the wall in minutes. Your answer will be considered correct if it has at most 10^{-4} absolute or relative error.

input

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

output

```
11.7669491886
```

input

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

output

```
2.0000000000
```

input

```
1 1
1 1
```

output

```
0.0000000000
```

Codeforces (c) Copyright 2010-2018 Mike Mirzayanov
The only programming contests Web 2.0 platform