2016 Canadian Computing Olympiad
Day 2, Problem 1
O Canada

Time Limit: 1 second

Problem Description

In this problem, a *grid* is an N-by-N array of cells, where each cell is either red or white.

Some grids are *similar* to other grids. Grid A is similar to grid B if and only if A can be transformed into B by some sequence of *changes*. A change consists of selecting a 2-by-2 square in the grid and flipping the colour of every cell in the square. (Red cells in the square will become white; white cells in the square will become red.)

You are given G grids. Count the number of pairs of grids which are similar. (Formally, number the grids from 1 to G, then count the number of tuples (i, j) such that $1 \leq i < j \leq G$ and grid i is similar to grid j.)

Input Specification

The first line of input contains N ($2 \leq N \leq 10$), the size of the grids. The second line contains G ($2 \leq G \leq 10000$), the number of grids. The input then consists of $N \cdot G$ lines, where each line contains N characters, where each character is either `R` or `W`, indicating the colour (red or white) for that element in the grid. Moreover, after the first two lines of input, the next N lines describe the first grid, the following N lines describe the second grid, and so on.

For 12 out of the 25 marks available for this question, $2 \leq G \leq 10$.

Output Specification

Output the number of pairs of grids which are similar.

Sample Input

```
2
2
RW
WR
WR
RW
```

Output for Sample Input

```
1
```

Explanation for Output for Sample Input

There are exactly two grids, and they are similar because the first grid can be transformed into the second grid using one change (selecting the 2-by-2 square consisting of the entire grid).

2016 Canadian Computing Olympiad
Day 2, Problem 2
Zombie Apocalypse

Time Limit: 2 seconds

Problem Description

Your country has a problem with zombies. That is, it has zombies, which are a problem. Thankfully, you are gainfully employed at the Forsenic Institute for Zoology and Zombie Emerging Studies (FIZZES), and your job is simply to give a measure of how bad the problem is.

You have mapped out your country on an an N-by-M array of cells marked with non-negative integers.

You have the exact locations of all the zombies, and know that no two zombies are in the same location. The cells containing a zombie are marked with 0. Next, all the unmarked cells touching a cell (where *touching a cell* means touching on any side or corner of a cell; so each cell touches up to 8 other cells) marked with 0 are marked with 1. Then, all the unmarked cells touching a cell marked with 1 are marked with 2. This process continues until all the cells are marked. These numbers indicate the level of concern your office has about the spread of zombies.

A small example is shown below.

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

Your boss has given you an integer Q, and you must determine the number of cells which are marked with the integer Q.

Input Specification

The first line of input will contain two space-separated integers N and M ($1 \leq N \leq 10^9; 1 \leq M \leq 10^9$) indicating the size of the grid. The next line contains the number K ($1 \leq K \leq 2000$), indicating the number of cells that contain zombies. The next K lines each contain two space-separated integers r_i c_i indicating the row and column of the ith zombie ($1 \leq r_i \leq N; 1 \leq c_i \leq M$). No two zombies are in the same cell: thus if $i \neq j$ then $(r_i, c_i) \neq (r_j, c_j)$. The last line will contain the integer Q ($0 \leq Q \leq N + M$).

For 5 of the 25 marks available, $N \leq 1000$ and $M \leq 1000$.

For an additional 5 of the 25 marks available, $K \leq 50$.

For an additional 5 of the 25 marks available, $N \leq 1000$.

Output Specification
Output the number of cells in the grid that are marked with the integer Q.

Sample Input
```
5 6
2
3 3
2 4
2
```

Output for Sample Input
```
15
```

Explanation for Output for Sample Input
The sample input is the example shown above, which has 15 2's.

2016 Canadian Computing Olympiad
Day 2, Problem 3
Pirates

Time Limit: 10 seconds

Problem Description

A group of N pirates found K gold coins. They must decide on a way to distribute the coins amongst themselves. They have agreed on the following rules:

The oldest pirate proposes a distribution. (You can assume that all the pirates' ages are distinct.) A distribution assigns an non-negative integer number of coins to each pirate such that the sum of these numbers equals K.

Then, each pirate will vote either 'yes' or 'no' on the proposal. The number of 'yes' votes required for the proposal to pass depends on the number of pirates. If there are X pirates, then $V[X]$ 'yes' votes are required for the proposal to pass. If the proposal passes, the coins are assigned according the proposed distribution and the process ends. Otherwise, the oldest pirate is thrown overboard and the process is repeated without him.

The pirates act according to the following rules. The rules are given in order of priority; for example, rule 2 is only applied to distinguish between multiple options that are optimal according to rule 1.

1. A pirate will act to prevent himself from being thrown overboard.
2. A pirate will act to maximize the number of coins he receives.
3. A pirate will act to maximize the number of pirates thrown overboard (excepting himself, because rule 1 takes priority).
4. A pirate will act to maximize the number of coins received by the oldest pirate. If there are still multiple choices that fit these rules, he will maximize the gold received by the second-oldest pirate, then the third-oldest pirate, etc.

If there are multiple options that are optimal according to these rules, then the pirate chooses an action arbitrarily. (You can assume that the answer to this problem does not depend on the pirate's choice in this case.) Additionally, all the pirates are perfectly logical and know all the information contained in this problem statement. They cannot form agreements or coalitions because they do not trust each other.

We will number the pirates from 1 to N, where these are numbered from the youngest (pirate 1) to the oldest (pirate N).

If there were only i pirates (where $i = 1, \ldots, N$), how many coins would the oldest of them get?

Input Specification
The first line of input will be the number N ($2 \le N \le 10^6$).

The second line of input will be the integer K ($1 \le K \le 10^{18}$).

The next N lines of input contain $V[i]$ ($1 \le V[i] \le i$) indicating the number of 'yes' votes required for a proposal to pass if there are i pirates remaining ($i = 1, \ldots, N$).

For 5 of the 25 available marks for this problem $N \le 2000$.

For an additional 5 of the 25 available marks for this problem $\max(1, i-3) \le V[i] \le i$ for all $i = 1, \ldots, N$.

For an additional 5 of the 25 available marks for this problem $K = 10^{18}$.

Output Specification
The output should consist of N integers, printed one per line. The ith line of output is the number of coins that the ith pirate would get if they were the oldest pirate; in other words, if only pirates $1, \ldots, i$ existed. If the ith pirate is thrown overboard, output -1 on the ith line.

Sample Input 1
```
5
100
1
1
2
2
3
```

Output for Sample Input 1
```
100
100
99
99
98
```

Explanation for Output for Sample Input 1
If there are 2 pirates left, pirate 2 can propose that all of the gold coins go to him. Only 1 vote is required for this proposal to pass, so he can guarantee that it passes by voting for it.

If there are 3 pirates left, pirate 3 needs someone else to vote for his proposal. He can ensure this by giving 1 coin to pirate 1 and 99 to himself. Pirate 1 knows that if the proposal doesn't pass, he will receive nothing. So a single coin is enough to secure his vote.

If there are 4 pirates left, pirate 4 gives 1 gold coin to pirate 2 and 99 to himself.

If there are 5 pirates left, pirate 5 gives 1 gold coin to pirates 1 and 3 and keeps 98 coins for himself.

Sample Input 2

```
5
100
1
2
3
4
4
```

Output for Sample Input 2

```
100
-1
-1
-1
100
```

Explanation for Output for Sample Input 2

In this case, a full consensus is required for a proposal to pass, except when there are 5 pirates, in which case only 4 of the 5 votes are required.

If there is full consensus required, and there are 2 or more pirates, the youngest pirate will vote against the proposal to maximize their coins and also throw the most pirates overboard.

In the case where there are 5 pirates, the oldest pirate will propose to give himself 100 coins. Everyone except the youngest pirate will vote 'yes', because if this proposal doesn't pass, the youngest pirate will get all of them thrown overboard and then take the coins for himself when only he remains. Since the pirates don't want to be thrown overboard, they will vote 'yes', even though the oldest pirate will offer them nothing.

Sample Input 3

```
4
100
1
1
2
3
```

Output for Sample Input 3

```
100
100
```

```
99
97
```

Explanation for Output for Sample Input 3

The first three cases were explained in Sample Input 1.

When there are 4 pirates, the oldest pirate needs 3 votes. He will give 2 coins to the youngest pirate and 1 coin to the second-youngest pirate, keeping the rest for himself.

Sample Input 4

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

Output for Sample Input 4

```
2
2
1
-1
```

Explanation for Output for Sample Input 4

The situation is the same as in Example 3, but now there are only 2 coins. With 1, 2 or 3 pirates, we have the same situations as in Example 3.

With 4 pirates, the oldest pirate does not have enough coins to ensure the 3 votes which he needs, so he will be thrown overboard.