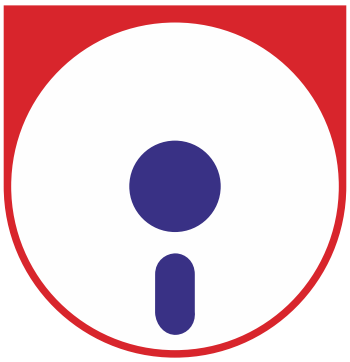


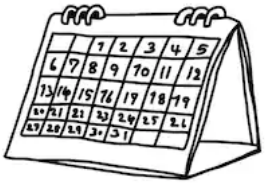
Croatian Open Competition in Informatics

Round 6, March 7th 2020

Tasks

Task	Time limit	Memory limit	Score
Datum	1 second	512 MiB	50
Birmingham	1 second	512 MiB	70
Konstrukcija	1 sekunda	512 MiB	110
Skandi	10 seconds	512 MiB	110
Trener	2 seconds	512 MiB	110
Total			450

Task Datum



The exam season at University of Zagreb is over and students are doing what they love the most – sleeping. In the rare moments of wakefulness, they usually scroll over their Instagram feed. Fabijan is one of those students.

Recently, he read the following caption – the date `02.02.2020.` is the first palindromic date in the last 909 years.

He realized the caption was incorrect and this made him wonder about palindromic dates so he asked himself for each of the N dates what is the first palindromic date that comes after that date. The date is considered *palindromic* if , when disregarding the dots, it is the same when read from left-to-right as if it was read from right-to-left. For example, dates `02.02.2020.` and `12.10.0121.` are palindromic, while `03.02.2020.` and `12.07.1993.` are not.

Note: In this task it is important to take account of leap years which have 29 days in February. For the purposes of this task, we consider a year to be a leap year if it is divisible by 4. Otherwise, months have 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30 and 31 days in order.

Input

The first line contains an integer N ($1 \leq N \leq 10\,000$) from the task description.

The next N lines contain a valid date in format `DD.MM.YYYY.`

Output

For each date from the input, you should output the first palindromic date that comes strictly after it. That date should be printed in the `DD.MM.YYYY.` and we guarantee that the solution exists in this format.

Scoring

In the test cases worth a total of 10 points, each date in the output will have the same month and year as the corresponding date from the input. Also, N will be equal to 10.

In the test cases worth an additional 10 points, each date in the output will have the same year as the corresponding date from the input. Also, N will be equal to 10.

In the test cases worth an additional 20 points, $N = 10$ will hold.

Examples

input

```
1
02.02.2020.
```

output

```
12.02.2021.
```

input

```
2
01.01.1000.
31.12.2026.
```

output

```
10.01.1001.
03.02.2030.
```

input

```
3
01.01.0100.
05.07.0321.
05.05.0505.
```

output

```
10.10.0101.
10.01.1001.
10.01.1001.
```

Clarification of the first example: Although the given date is palindromic, Fabijan is interested in the first date that strictly comes after it. That date is `12.02.2021.`

Task Birmingham

It is well known that all horse races in Birmingham are fixed days in advance. It is a little less known that certain people that fix these races (and thereby know the winner) start spreading that information around the city the next day.

The first day after the meeting, all people that know the information about the winner start sharing that information with all people that live at most K steps away from their house.

The second day after the meeting, all people that know the information about the winner start sharing that information with all people that live at most $2 \cdot K$ steps away from their house.

In general, X-th day after the meeting, all people that know the information about the winner start sharing that information with all people that live at most $X \cdot K$ steps away from their house.

We can represent Birmingham as a graph where vertices represent the houses and edges represent bidirectional roads which connect these houses. Houses are indexed with increasing integers from 1 to N and we say that a person can travel each road in a single step. It is possible to reach each house from each other house by traversing a sequence of roads.

Your task is to determine, for each house, on which day will the information about the race winner reach it.

Input

The first line contains four integers N, M, Q and K ($1 \leq N, Q, K \leq 100\,000, Q \leq N, 1 \leq M \leq 200\,000$), the number of houses in Birmingham, the number of roads in Birmingham, the number of people that were present on the secret meeting and the number K from task description.

The next line contains Q integers where the i-th integer represents the index of a house where the i-th person from the secret meeting lives in.

The i-th of the next M lines contains integers A_i and B_i ($1 \leq A_i, B_i \leq N, A_i \neq B_i$), which denote that the i-th road connects houses with indices A_i and B_i.

Output

Output N numbers where the i-th number represents on which day after the meeting will the person living in house with index i find out who will win the race. If the person living in that house was present on the secret meeting, output 0 instead.

Scoring

In the test cases worth a total of 20 points, it will hold $K = 1$, $1 \leq N, Q \leq 100$ and $1 \leq M \leq 200$.
In the test cases worth an additional 15 points, it will hold $1 \leq N, Q \leq 100$ and $1 \leq M \leq 200$.

Examples

input

```
6 8 1 1
6
1 3
1 5
1 6
2 5
2 6
3 4
3 5
5 6
```

output

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

input

```
6 8 2 1
6 4
1 3
1 5
1 6
2 5
2 6
3 4
3 5
5 6
```

output

```
1 1 1 0 1 0
```

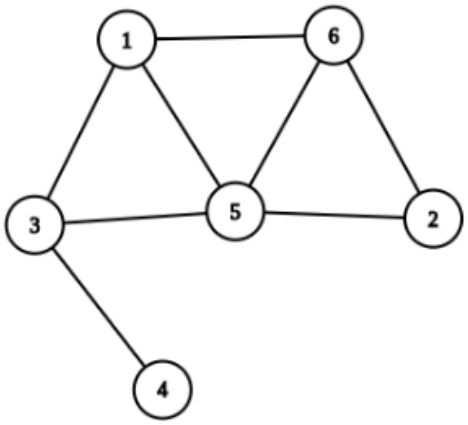
input

```
6 8 1 2
6
1 3
1 5
1 6
2 5
2 6
3 4
3 5
5 6
```

output

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

Clarification of the third example: The figure represents a graph from the third example. Since houses 1, 2, 3 and 5 are at most two steps away from house 6, people living in them will find out about the winner the day after the meeting. Person living in house 4 will find out about the winner two days after the meeting.



Task Konstrukcija

Let G be a directed acyclic graph. If $c_1, c_2, c_3, \dots c_n$ are distinct vertices of G such that there is a path from c_1 to c_2, there is a path from c_2 to c_3, ...and there is a path from c_{n-1} to c_n, we say that array $C = (c_1, c_2, c_3, \dots c_n)$ is an ordered array starting at c_1 and ending at c_n. Note that between neighbouring elements c_i and c_{i+1} of ordered array C it isn't necessary to exist a direct edge, it is enough for the path to exist from c_i to c_{i+1}.

For this definition of an ordered array $C = (c_1, c_2, c_3, \dots c_n)$, we define its length $len(C) = n$. Therefore, the length of an ordered array is equal to the number of vertices it holds. Note that the ordered array can have a length of 1 when holding a single vertex which represents both its beginning and its end.

Also, for an ordered array $C = (c_1, c_2, c_3, \dots c_n)$ we can define its sign as $sgn(C) = (-1)^{len(C)+1}$. For vertices x and y of G, let's denote with $S_{x,y}$ a set of all ordered arrays that start in x and end in y.

Finally, we define the tension between nodes x and y as $tns(x, y) = \sum_{C \in S_{x,y}} sgn(C)$. Therefore, the tension between nodes x and y equals the sum of signs of all ordered arrays that start in x and end in y.

An integer K is given. Your task is to construct a directed acyclic graph with **at most 1000** vertices and **at most 1000** edges for which $tns(1, N) = K$ holds. Number N in the previous expression denotes the number of vertices in a graph. Vertices of a graph should be indexed using positive integers from 1 to N.

Input

The first line contains an integer K ($|K| \leq 10^{18}$) from the task description.

Output

In the first line you should output the number of vertices and the number of edges of the constructed graph. Let's denote the number of vertices of that graph with N ($1 \leq N \leq 1000$), and the number of edges with M ($0 \leq M \leq 1000$).

In the i-th of the next M lines you should output two distinct integers X_i and Y_i ($1 \leq X_i, Y_i \leq N$), which represent the i-th edge which is directed from vertex with index X_i towards vertex with index Y_i. Each edge must appear only once in the output.

Also, the absolute value of tension between each two nodes in the graph must be less or equal to 2^{80}.

If there are multiple solutions, output any of them.

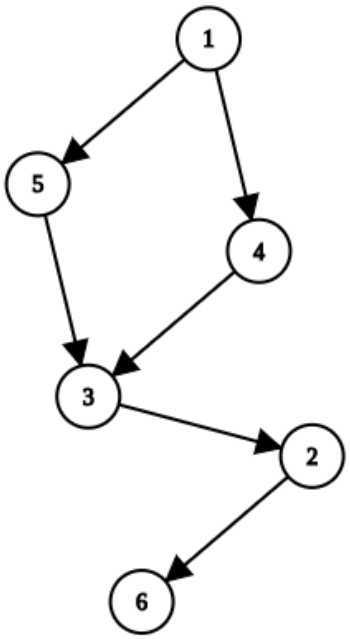
Scoring

Subtask	Score	Constraints
1	15	$1 \leq K < 500$
2	15	$-300 < K \leq 1$
3	20	$\|K\| < 10000$
4	60	No additional constraints.

Examples

input	input	input
0	1	2
output	**output**	**output**
6 6 1 4 1 5 4 3 5 3 3 2 2 6	1 0	6 8 1 2 1 3 1 4 1 5 5 4 2 6 3 6 4 6

Clarification of the first example: The constructed graph has 6 vertices. Ordered arrays that start in 1 and end in 6 are: $(1,6)$, $(1,4,6)$, $(1,5,6)$, $(1,3,6)$, $(1,2,6)$, $(1,4,3,6)$, $(1,4,2,6)$, $(1,5,3,6)$, $(1,5,2,6)$, $(1,3,2,6)$, $(1,4,3,2,6)$, $(1,5,3,2,6)$. Their lengths are (in order): $1,2,2,2,2,3,3,3,3,3,4,4$, so their signs are $-1,1,1,1,1,-1,-1,-1,-1,-1,1,1$. Therefore, the tension between 1 and 6 is equal to $-1+1+1+1+1-1-1-1-1-1+1+1=0$.



Task Skandi

Dragica is a captain of a local semi-professional bowling team, she is a passionate chef and probably one of the best crossword puzzle solvers in Croatia. A crossword puzzle consists of $N \times M$ squares arranged in N rows and M columns. Some of the squares are empty (and should be filled with letters by answering questions) and some of the squares are filled. Filled squares contain at most two questions which should either be answered horizontally to the right or vertically downwards. The answers to the questions are written in the empty squares before until we reach the crossword bound or an initially-filled square. An initially-filled square contains a horizontal question if a square to the right exists and is empty. Analogously, an initially-filled square contains a vertical question if a square beneath it exists and is empty.

Dragica usually knows all the answers to the crossword puzzle questions, but wishes to read and answer as little questions as possible and still solve the entire crossword puzzle. Help her achieve her goal.

Input

The first line contains integers N and M ($2 \leq N, M \leq 500$) from the task description.

Each of the next N lines contain M characters `'0'` or `'1'`, where `'0'` denotes an empty square which should be filled by answering a question and `'1'` denotes an initially filled square which may contain at most two questions as explained in the task description. The first row and first column will be filled with `'1'` characters.

It is guaranteed that there will be at least one character `'0'` in the input.

Output

In the first line you should output the minimal number of questions that can be answered in order to solve an entire crossword puzzle. Let's denote that number with X.

Each of the next X lines should describe one question that should be answered. The question description should be printed in the format `R C direction`, where R is the row number of the question, C is the column number of the question and direction is either `"DESNO"` (Croatian for `RIGHT`) or `"DOLJE"` (Croatian for `DOWN`) depending on whether Dragica should answer the vertical or horizontal question.

If there are multiple solutions, output any of them.

Scoring

Subtask	Score	Constraints
1	18	There will be at most 9 squares with `'1'`
2	32	$N \leq 500$ and $M \leq 10$
3	60	No additional constraints.

If your solution outputs the first line correctly on each test case of a subtask, but it fails to correctly output the other lines in some test case, you will score 50% of the allocated points for that subtask.

Examples

input

```
4 5
11111
10000
10000
10000
```

output

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

input

```
6 4
1111
1011
1000
1011
1010
1000
```

output

```
4
1 2 DOLJE
4 4 DOLJE
5 3 DOLJE
3 1 DESNO
```

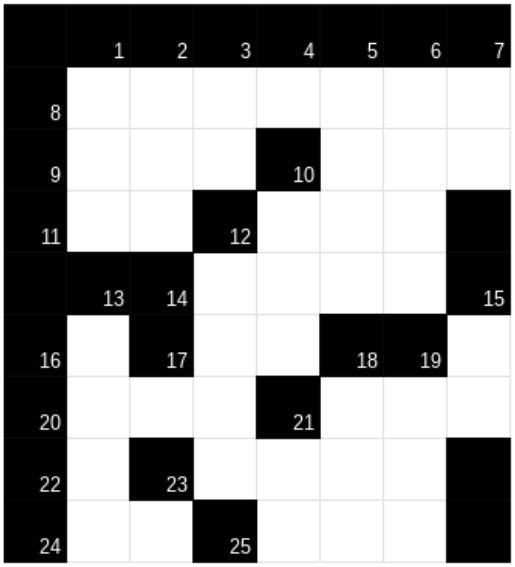
input

```
9 8
11111111
10000000
10001000
10010001
11100001
10100110
10001000
10100001
10010001
```

output

```
14
5 2 DOLJE
5 8 DOLJE
8 3 DOLJE
2 1 DESNO
3 1 DESNO
3 5 DESNO
4 1 DESNO
4 4 DESNO
5 3 DESNO
6 3 DESNO
7 1 DESNO
7 5 DESNO
8 3 DESNO
9 4 DESNO
```

Clarification of the third example: An example of a real crossword puzzle which is equivalent to the one described in this example is given on the next page. Initially-filled squares are colored black and those squares that contain at least one question are numerated. Below the puzzle you can see the questions that should be solved *to-the-right* (column "across") and those that should be solved downwards (column "down"). Note that some initially-filled squares contain no questions, some contain a single question (e.g. 8 and 13), while some contain two questions (e.g. 10 and 12). In order to solve this crossword puzzle you need to know the answers to at least 14 questions that are listed in the output. Can you solve it?



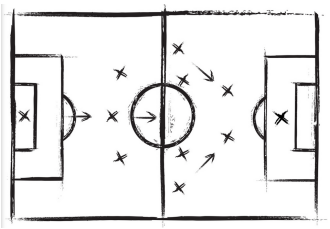
Across:

8. Father of dynamic programming, Richard
9. Frames per second
10. Exclamation when bullfighting
11. chr(115) * 2 in Python
12. Internet of things
14. Croatian version of COCI
16. Lexicographically smallest word
17. International system of units
19. Fictional character from James Bond
20. Balkan olympiad in informatics
21. Contest on TopCoder
22. Boron
23. Special pointer in C
24. Artificial intelligence
25. Popular encryption algorithm

Down:

1. Breadth first search
2. Machine epsilon constant
3. Unix command (contents of a directory)
4. Sixteenth letter of Croatian alphabet
5. REM hit - Man on the _____
6. Plural of altus
7. Croatian negation
10. International olympiad in informatics
12. Croatian computer science association
13. Palindromic pop-group from Sweden
15. Sqrt decomposition algoritam
17. Oxygen
18. Bad news when submitting a solution
19. Clear screen in QBasic
21. Last and first wovel
23. Iodine

Task Trener



At this point we already know that students love to sleep. Patrik is a record holder in this category. He wakes up only when he needs to eat or if he wishes to play *FIFA 20*. Therefore, his dreams usually revolve around football. In his last dream, he found himself in the role of a football manager of his favourite team – GNK Dinamo Zagreb.

His job is to select N players that will defend the blue colors in the next season, but the board has some peculiar requests. They are:

- All players must have surnames of distinct lengths.
- Surname of a player must appear as a continuous subsequence of surnames of all players whose surnames are longer.

To make his job easier, Patrik divided the potential players in N buckets such that players in i-th bucket have exactly i letters in their surname. In each of these buckets there are exactly K players. Patrik wants to know in how many distinct ways (modulo $10^9 + 7$) can he choose the players for his squad while also conforming to the given requests.

Input

The first line contains two integers N ($1 \leq N \leq 50$) and K ($1 \leq K \leq 1\,500$).

Each of the next N lines contains K not necessarily distinct surnames of players. The surnames of players in i-th of those lines consist of exactly i lowercase letters from the English alphabet.

Output

In the only line you should output the answer from the task description.

Bodovanje

Subtask	Score	Constraints
1	22	$N = 5$ and $K = 10$
2	33	$N = 50$ and $K = 100$
3	55	No additional constraints.

Examples

input	**input**	**input**

`3 2`
`a b`
`ab bd`
`abc abd`
	`3 3`
	`a b c`
	`aa ab ac`
	`aaa aab aca`
		`3 1`
		`a`
		`bc`
		`def`

output	**output**	**output**
`5`	`6`	`0`

Clarification of the first example: Patrik can choose the following teams: `(a, ab, abc)`, `(a, ab, abd)`, `(b, ab, abc)`, `(b, ab, abd)` and `(b, bd, abd)`.