Problem Set 7: Games
15-295 Spring 2018

A. Win or Freeze

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

You can't possibly imagine how cold our friends are this winter in Nvodsk! Two of them play the following game to warm up: initially a piece of paper has an integer q. During a move a player should write any integer number that is a *non-trivial* divisor of the last written number. Then he should run this number of circles around the hotel. Let us remind you that a number's divisor is called *non-trivial* if it is different from one and from the divided number itself.

The first person who **can't make a move wins** as he continues to lie in his warm bed under three blankets while the other one keeps running. Determine which player wins considering that both players play optimally. If the first player wins, print any winning first move.

Input

The first line contains the only integer q ($1 \le q \le 10^{13}$).

Please do not use the `%lld` specificator to read or write 64-bit integers in C++. It is preferred to use the `cin`, `cout` streams or the `%I64d` specificator.

Output

In the first line print the number of the winning player (1 or 2). If the first player wins then the second line should contain another integer — his first move (if the first player can't even make the first move, print 0). If there are multiple solutions, print any of them.

Examples

input
```
6
```
output
```
2
```

input
```
30
```
output
```
1
6
```

input
```
1
```
output
```
1
0
```

Note

Number 6 has only two non-trivial divisors: 2 and 3. It is impossible to make a move after the numbers 2 and 3 are written, so both of them are winning, thus, number 6 is the losing number. A player can make a move and write number 6 after number 30; 6, as we know, is a losing number. Thus, this move will bring us the victory.

B. Dinner with Emma

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

Jack decides to invite Emma out for a dinner. Jack is a modest student, he doesn't want to go to an expensive restaurant. Emma is a girl with high taste, she prefers elite places.

Munhattan consists of n streets and m avenues. There is exactly one restaurant on the intersection of each street and avenue. The streets are numbered with integers from 1 to n and the avenues are numbered with integers from 1 to m. The cost of dinner in the restaurant at the intersection of the i-th street and the j-th avenue is c_{ij}.

Jack and Emma decide to choose the restaurant in the following way. Firstly Emma chooses the street to dinner and then Jack chooses the avenue. Emma and Jack makes their choice optimally: Emma wants to maximize the cost of the dinner, Jack wants to minimize it. Emma takes into account that Jack wants to minimize the cost of the dinner. Find the cost of the dinner for the couple in love.

Input

The first line contains two integers n, m $(1 \le n, m \le 100)$ — the number of streets and avenues in Munhattan.

Each of the next n lines contains m integers c_{ij} $(1 \le c_{ij} \le 10^9)$ — the cost of the dinner in the restaurant on the intersection of the i-th street and the j-th avenue.

Output

Print the only integer a — the cost of the dinner for Jack and Emma.

Examples

input

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

output

```
2
```

input

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

output

```
1
```

Note

In the first example if Emma chooses the first or the third streets Jack can choose an avenue with the cost of the dinner 1. So she chooses the second street and Jack chooses any avenue. The cost of the dinner is 2.

In the second example regardless of Emma's choice Jack can choose a restaurant with the cost of the dinner 1.

C. Game

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

Two players play the following game. Initially, the players have a knife and a rectangular sheet of paper, divided into equal square grid cells of unit size. The players make moves in turn, the player who can't make a move loses. In one move, a player can take the knife and cut the paper along any segment of the grid line (not necessarily from border to border). The part of the paper, that touches the knife at least once, is considered cut. There is one limit not to turn the game into an infinite cycle: each move has to cut the paper, that is the knife has to touch the part of the paper that is not cut before.

Obviously, the game ends when the entire sheet is cut into 1×1 blocks. During the game, the pieces of the sheet are not allowed to move. It is also prohibited to cut along the border. The coordinates of the ends of each cut must be integers.

You are given an $n \times m$ piece of paper, somebody has already made k cuts there. Your task is to determine who will win if the players start to play on this sheet. You can consider that both players play optimally well. If the first player wins, you also need to find the winning first move.

Input

The first line contains three integers n, m, k $(1 \le n, m \le 10^9, 0 \le k \le 10^5)$ — the sizes of the piece of paper and the number of cuts. Then follow k lines, each containing 4 integers xb_i, yb_i, xe_i, ye_i $(0 \le xb_i, xe_i \le n, 0 \le yb_i, ye_i \le m)$ — the coordinates of the ends of the existing cuts.

It is guaranteed that each cut has a non-zero length, is either vertical or horizontal and doesn't go along the sheet border.

The cuts may intersect, overlap and even be the same. That is, it is not guaranteed that the cuts were obtained during any correct game.

Output

If the second player wins, print "SECOND". Otherwise, in the first line print "FIRST", and in the second line print any winning move of the first player (the coordinates of the cut ends, follow input format to print them).

Examples

input

```
2 1 0
```

output

```
FIRST
1 0 1 1
```

input

```
2 2 4
0 1 2 1
0 1 2 1
1 2 1 0
1 1 1 2
```

output

```
SECOND
```

D. Yet Another Number Game

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

Since most contestants do not read this part, I have to repeat that Bitlandians are quite weird. They have their own jobs, their own working method, their own lives, their own sausages and their own games!

Since you are so curious about Bitland, I'll give you the chance of peeking at one of these games.

BitLGM and BitAryo are playing yet another of their crazy-looking genius-needed Bitlandish games. They've got a sequence of n non-negative integers $a_1, a_2, ..., a_n$. The players make moves in turns. BitLGM moves first. Each player can and must do one of the two following actions in his turn:

- Take one of the integers (we'll denote it as a_i). Choose integer x $(1 \le x \le a_i)$. And then decrease a_i by x, that is, apply assignment: $a_i = a_i - x$.
- Choose integer x $(1 \le x \le \min_{i=1}^{n} a_i)$. And then decrease all a_i by x, that is, apply assignment: $a_i = a_i - x$, for all i.

The player who cannot make a move loses.

You're given the initial sequence $a_1, a_2, ..., a_n$. Determine who wins, if both players plays optimally well and if BitLGM and BitAryo start playing the described game in this sequence.

Input

The first line contains an integer n $(1 \le n \le 3)$.

The next line contains n integers $a_1, a_2, ..., a_n$ $(0 \le a_i < 300)$.

Output

Write the name of the winner (provided that both players play optimally well). Either "`BitLGM`" or "`BitAryo`" (without the quotes).

Examples

input

```
2
1 1
```

output

```
BitLGM
```

input

```
2
1 2
```

output

```
BitAryo
```

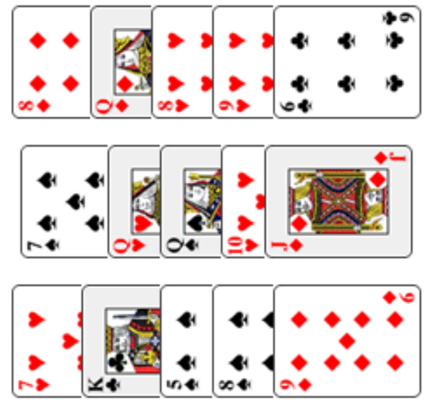
input

```
3
1 2 1
```

output

```
BitLGM
```

E. Game of Cards



Alice and Bob created a new game while at the beach this summer. All they need is a set of numbered playing cards. They start by creating P piles with all cards face-up and select a non-negative number K. After that, they take turns like this:

1. A player starts by selecting one of the piles.
2. Then, he removes from 0 up to K cards from the top of that pile, leaving at least one card in the pile.
3. Next, he looks at the card left at the top of the pile and must remove a number of cards equal to its value (from the top of the same pile).

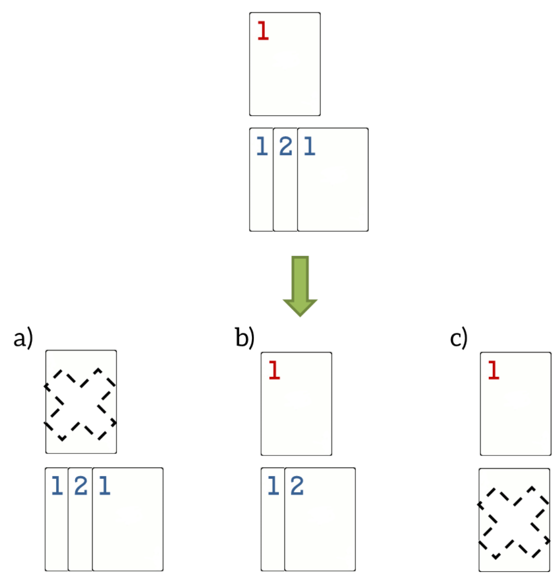
Whoever doesn't have more cards to remove, or whoever is forced to remove more cards than those available on a pile, loses the game.

In the figure, you can see an example with two piles and $K = 1$. The player to move might:

a) Select the first pile and 0 cards to remove, being forced to remove 1 card from the top next.

b) Select the second pile and 0 cards to remove, having to remove 1 card from the top next.

c) Select the second pile and 1 card to remove, having to remove 2 cards from the top next.



Alice has realized that Bob is very good at this game and will always win if he has the chance. Luckily, this time Alice is first to play. Is Alice able to win this game?

Task

Given the description of the piles with all the cards and the maximum number of cards they can start to remove, your goal is to find out whether Alice can win the game if she is the first to play.

Input

The first line contains 2 space separated integers, P, the number of piles, and K, the maximum number of cards they can start to remove on their turn. The next P lines start with an integer N, indicating the number of cards on a pile. N space separated integers follow, representing the cards on that pile from the bottom to the top.

Constraints

$1 \leq P \leq 100$	Number of piles.
$1 \leq K \leq 10$	Maximum number of cards a player can start to remove.
$1 \leq c \leq 10$	Number on each card.
$1 \leq N \leq 1\,000$	Size of each pile.

Output

A single string, stating "Alice can win." or "Bob will win.", as appropriate.

Sample Input 1

```
4 1
4 1 1 1 1
6 2 1 2 1 2 1
4 1 1 1 1
6 2 1 2 1 2 1
```

Sample Output 1

```
Bob will win.
```

Output 1 Explanation

The piles are the same, so Bob will always be able to mirror whatever move Alice makes.

Sample Input 3

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

Sample Output 3

```
Alice can win.
```

Sample Input 2

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

Sample Output 2

```
Alice can win.
```

Output 2 Explanation

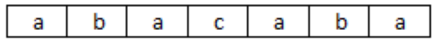
Alice can start by removing 0 cards from the second pile and then 1 card from its top. Two legal moves will be possible next, Bob will make one and Alice the other.

F. Playing with String

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

Two people play the following string game. Initially the players have got some string s. The players move in turns, the player who cannot make a move loses.

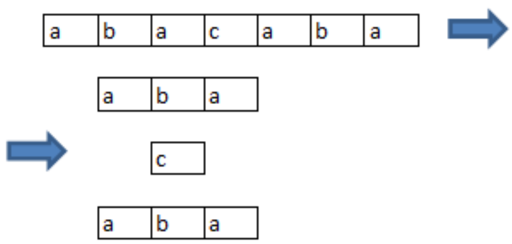
Before the game began, the string is written on a piece of paper, one letter per cell.



An example of the initial situation at s = "abacaba"

A player's move is the sequence of actions:

1. The player chooses one of the available pieces of paper with some string written on it. Let's denote it is t. Note that initially, only one piece of paper is available.
2. The player chooses in the string $t = t_1t_2... t_{|t|}$ character in position i $(1 \le i \le |t|)$ such that for some positive integer l $(0 < i - l; i + l \le |t|)$ the following equations hold: $t_{i-1} = t_{i+1}$, $t_{i-2} = t_{i+2}$, ..., $t_{i-l} = t_{i+l}$.
3. Player cuts the cell with the chosen character. As a result of the operation, he gets three new pieces of paper, the first one will contain string $t_1t_2... t_{i-1}$, the second one will contain a string consisting of a single character t_i, the third one contains string $t_{i+1}t_{i+2}... t_{|t|}$.



An example of making action $(i = 4)$ with string s = «abacaba»

Your task is to determine the winner provided that both players play optimally well. If the first player wins, find the position of character that is optimal to cut in his first move. If there are multiple positions, print the minimal possible one.

Input

The first line contains string s $(1 \le |s| \le 5000)$. It is guaranteed that string s only contains lowercase English letters.

Output

If the second player wins, print in the single line "Second" (without the quotes). Otherwise, print in the first line "First" (without the quotes), and in the second line print the minimal possible winning move — integer i $(1 \le i \le |s|)$.

Examples

input Copy
```
abacaba
```
output
```
First
2
```

input Copy
```
abcde
```
output
```
Second
```

Note

In the first sample the first player has multiple winning moves. But the minimum one is to cut the character in position 2.

In the second sample the first player has no available moves.

G. Sharti

time limit per test: 5 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

During the last 24 hours Hamed and Malek spent all their time playing "Sharti". Now they are too exhausted to finish the last round. So they asked you for help to determine the winner of this round.

"Sharti" is played on a $n \times n$ board with some of cells colored white and others colored black. The rows of the board are numbered from top to bottom using number 1 to n. Also the columns of the board are numbered from left to right using numbers 1 to n. The cell located at the intersection of i-th row and j-th column is denoted by (i, j).

The players alternatively take turns. In each turn the player must choose a square with side-length at most k with its lower-right cell painted white. Then the colors of all the cells in this square are inversed (white cells become black and vice-versa). The player who cannot perform a move in his turn loses.

You know Hamed and Malek are very clever and they would have played their best moves at each turn. Knowing this and the fact that Hamed takes the first turn, given the initial board as described in the input, you must determine which one of them will be the winner.

Input

In this problem the initial board is specified as a set of m rectangles. All cells that lie inside at least one of these rectangles are colored white and the rest are colored black.

In the first line of input three space-spereated integers n, m, k ($1 \le k \le n \le 10^9$, $1 \le m \le 5 \cdot 10^4$) follow, denoting size of the board, number of rectangles and maximum size of the turn square during the game, respectively.

In i-th line of the next m lines four space-seperated integers a_i, b_i, c_i, d_i ($1 \le a_i \le c_i \le n$, $1 \le b_i \le d_i \le n$) are given meaning that i-th rectangle determining the initial board is a rectangle with upper-left cell at (a_i, b_i) and lower-right cell at (c_i, d_i).

Output

If Hamed wins, print "Hamed", otherwise print "Malek" (without the quotes).

Examples

input

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

output

```
Malek
```

input

```
12 5 7
3 4 5 6
1 2 1 2
4 5 9 9
8 6 12 10
12 4 12 4
```

output

```
Hamed
```