Passing

Herbert is organizing a simple game of "passing". This game involves players passing a single ball among themselves. To play this game, Herbert has invited his friends over to come and play this simple game. Being an expert in the game, Herbert does not play together with his friends. After all, he does not want to take any advantage and wants the level of the playing field to be balanced. He becomes the game master and organizes the game instead.

Once his friends have arrived, Herbert first explained the game to them. This game is very simple and easy to play.

The passing game goes like this:

1. There are N people, arranged in a circle, that are taking part in this game, numbered from 1 to N.
2. Player `i+1` is located to the right of player `i` while player `1` is located at the right of player **N**.
3. There are multiple turns. In each turn, there will be **A** number of passes. The player holding the ball will always pass the ball to the player located at his right side, and so on, until **A** passes occurred.
4. Suppose after **A** passes, player **X** receives the ball:
   - If player **X** has received the ball **K** times before and including this turn, he is eliminated from the game and gives the ball to the player located at his right side.
   - If player **X** has received the ball less than **K** times (before and including this turn), player **X** will swap places with the player that held the ball at the start of this turn. If player **X** is the player that held the ball at the start of this turn, no swap will take place.
5. After each turn, the one with the ball will have to report to the game master, in this case, Herbert.

Note that a player is defined as receiving a ball if he is the player who holds the ball at the end of each turn (after the passing occurred). Receiving a ball in the middle of a round or getting a ball from a player who has just been eliminated does not count as receiving the ball officially.

Input

The first line of the input contains three integers **N, M, and K** respectively, separated by a single space.

N lines follow, each consists of a single string representing the name of a person, starting from the first person to the N^{th} person.

M lines follow, each line represents a turn. Each line consists of an integer **A**, indicating the value of **A** for that specific turn.

It is guaranteed that at any moment, there are at least two participants in the game.

Output

At the end of every turn, print the name of the player who reports to Herbert, the game master. The one who reports to Herbert is the one holding the ball at the end of the turn.

Sample Input

```
5 4 2
sunteck
jonathan
rey
harshul
basu
2
1
6
5
```

Sample Output

```
rey
jonathan
rey
harshul
```

Explanation

Initially, the players are arranged in this order (with player 1 located in between player 2 and 5). The box that is shaded in darker indicates the player holding the ball. Initially, the ball is with sunteck:

1	2	3	4	5
sunteck	jonathan	rey	harshul	basu
0	0	0	0	0

At the end of the first round (2 passes occurred) and before the start of the second round, rey receives the ball. This means rey has received 1 ball so far and rey is still in the game. rey swaps position with sunteck and the state now looks like this:

1	2	3	4	5
rey	jonathan	sunteck	harshul	basu
1	0	0	0	0

At the end of the second round (1 pass occurred), jonathan receives the ball. He swaps places with rey and the state now looks like this:

1	2	3	4	5
jonathan	rey	sunteck	harshul	basu
1	1	0	0	0

At the end of the third round (6 passes occurred), rey receives the ball. Since has now received the ball twice (and K = 2), he is out of the game (after reporting to Herbert) and gives the ball to sunteck, the player on his right. The state of the game now looks like this:

1	2	3	4
jonathan	sunteck	harshul	basu
1	0	0	0

At the end of the fourth and last round (5 passes occurred), harshul receives the ball for the first time. The final state of the game looks like this:

1	2	3	4
jonathan	harshul	sunteck	basu
1	1	0	0

Skeleton

You are given the skeleton file Passing.java (see contents on file).

Notes

1. You **must use linked list** to solve this problem.
2. You are free to define your own linked list class (encouraged for practice and skeleton file given), but you are allowed to use Java's built-in linked list implementation if it is suitable for this problem.
3. You are free to (and should) modify the skeleton file and add more attributes or methods when necessary.