Problem 4: Movie

Introduction

Residents of OneDimensionVille like movies very much, therefore a new, very large SinglePlex^{TM} movie theater has been built in the city. Due to the topology of OneDimensionVille, the seats in the movie are arranged in such a way that every row contains only one seat, but there are a very large number of rows. Row 1 is the row closest to the screen.

Unfortunately, the state-of-the-art computerized ticket management system developed by MacroHard Inc. is not always working correctly. Sometimes a seat is sold multiple times. However, the moviegoers are very polite and try to avoid conflicts if possible: if someone finds out that his/her seat is already occupied, then he/she sits on the first empty seat closer to the screen. (Thus if the ticket is for row 17 and rows 14–17 are occupied, then row 13 will be chosen.) If there is no empty seat closer to the screen, then our moviegoer goes home sadly, even if there are empty seats farther from the screen.

The manager of the movie theater finally decided to do something about this problem. It is very difficult and very expensive to fix the software, and anyway, the next version will be bug-free (at least that's what MacroHard says, so it must be true). Instead, the manager hires You to write a program which simulates the arrival of the audience and determines who will sit where and who will go home without watching the movie. This simulation will be very important in better understanding the situation and will help the manager make the best decision.

Input

The input contains several blocks of test cases. Each case begins with a line containing two numbers n and m, the input is terminated by a block with $n = m = 0$. The number n ($1 \le n \le 300000$) is the number of people arriving and m ($1 \le m \le 300000$) is the number of rows (movies are *very* popular in OneDimensionVille!) The first line is followed by n lines, each containing an integer: the number in the ith line is the row number on the ticket of the ith people arriving.

Output

For each test case, you have to output n lines. The ith line should contain a single number, which is the row where the ith people sits, or 0 if the ith people did not find an empty seat and went home.

Sample Input

```
5 10
3
3
4
4
8
3 10
2
2
2
0 0
```

Sample Output

```
3
2
4
1
8
2
1
0
```