A6 – Pack the House

Companies like Ticketmaster sell tickets to events and need to ensure that no seat is sold twice. Additionally, people usually buy "groups" of tickets, and would like all tickets in the group to be adjacent.

In this problem, you will allocate groups of tickets to a fictional arena. The arena can be imagined as a rectangle with R rows, and S seats per row. Tickets will be requested as groups, each group requesting a number of tickets N. If a request is to be satisfied, the N tickets must be allocated to adjacent empty seats in the same row (people in groups don't like to sit in front or behind their friends).

Seats are allocated in a first-come, first-served basis. As a request arrives, tickets are allocated to the first row (closest to the stage) with enough seats to hold all N members of the group. If there is no row with enough seats, the customer is told that the arena is too full to hold the group. We're interested in finding out how many groups are rejected in a given situation.

Input:

A test case will start with values for R, S, and G (the number of groups) on the first line. These values will be non-negative ($\leqslant$ 10,000). There will follow a line with G values on it, each line representing a request. Each request will list its value for N (the number of seats requested) on a separate line.

Output:

Output a line of the form:

```
X of the G groups did not get tickets.
```

or

```
All groups got tickets.
```

depending on which situation applies. X is the number of groups that could not be seated according to the algorithm described above

Input and output samples:

Input:
4 5 6 1 2 3 4 4 3

Output:
1 of the 6 groups did not get tickets.

Input:
10 10 10 1 1 1 1 1 1 1 1 1 1

Output:
All groups got tickets.