Problem B. Billboard

Input file:	`billboard.in`
Output file:	`billboard.out`
Time limit:	3 seconds
Memory limit:	256 megabytes

At the entrance to the university, there is a huge rectangular billboard of size $h \times w$ (h is its height and w is its width). The board is the place where all possible announcements are posted: nearest programming competitions, changes in the dining room menu, and other important information.

On September 1, the billboard was empty. One by one, the announcements started being put on the billboard.

Each announcement is a stripe of paper of unit height. More specifically, the i-th announcement is a rectangle of size $1 \times w_i$.

When someone puts a new announcement on the billboard, she would always choose the topmost possible position for the announcement. Among all possible topmost positions she would always choose the leftmost one.

If there is no valid location for a new announcement, it is not put on the billboard (that's why some programming contests have no participants from this university).

Given the sizes of the billboard and the announcements, your task is to find the numbers of rows in which the announcements are placed.

Input

The first line of the input file contains three integer numbers, h, w, and n ($1 \le h, w \le 10^9$; $1 \le n \le 200\,000$) — the dimensions of the billboard and the number of announcements.

Each of the next n lines contains an integer number w_i ($1 \le w_i \le 10^9$) — the width of i-th announcement.

Output

For each announcement (in the order they are given in the input file) output one number — the number of the row in which this announcement is placed. Rows are numbered from 1 to h, starting with the top row. If an announcement can't be put on the billboard, output "`-1`" for this announcement.

Example

`billboard.in`	`billboard.out`
`3 5 5`	`1`
`2`	`2`
`4`	`1`
`3`	`3`
`3`	`-1`
`3`	