PEG-O-Stripes

Time limit: 1.0s **Memory limit:** 16M

Mo has been attending every single PEG practice lately (so now we know which Mo we are talking about) and he got a little bit — hmmm — bored. To bring some excitement, he invented a new game: PEG-O-STRIPES, and decided to challenge another Mo to a duel. However, he wants to win for sure, so he hired David (Pritchard, of course) to come up with a winning strategy for him, or at least to tell him whether he can win. Dave agreed under the condition that Mo (A.) will always begin.

PEG-O-STRIPES involves two players who are given an infinite supply of stripes in three colours: red, green and blue. All of the red stripes have dimensions $R \times 1$, blue ones: $B \times 1$, and green ones: $G \times 1$, where R, B, and G are given natural numbers. Players take turns by placing given stripes on a board with dimensions $L \times 1$. They have to follow the following rules:

1. stripes can be placed anywhere within the board
2. stripes cannot overlap

The first player who cannot place any stripes on the board according to the given rules loses. The player that begins is said to have a winning strategy, if he wins no matter how the second player plays. Write a program that can determine whether the first player has a winning strategy for given dimensions L, R, B, and G. If yes, output `1`, if no, output `2`.

Input Specification

One line containing three numbers: R, B, and G $(1 < R, B, G \leq 1\,000)$.

One line containing M $(1 < M \leq 1\,000)$, a number of boards to consider.

M lines each containing the length L $(1 < L \leq 1\,000)$ of a board to be considered.

Output Specification

For each test case, output `1` if the first player has a winning strategy, and `2` if not.

Separate test cases by a blank line.

Sample Input

```
1 5 1
4
1
5
6
999
```

Sample Output

```
1
1
2
1
```