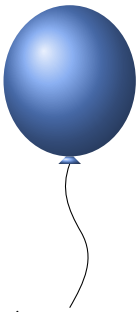
XAPC Algorithmic Programming Contest > badas

F Aux Champs Élysées



Time limit: 5.0s
Memory limit: 512MB

Maximilien is a student officer at Polytechnique and is preparing for the traditional *défilé du 14 juillet* on the Champs-Élysées. However, Maximilien prefers chaos over order, and thus plans to organize a prank that will definitely make an impression on the big day!

For the prank, Maximilien settles on the following idea: for each row of students parading, he will choose some number a_i of students in that row to wear their hats backward.

Not satisfied with this alone, Maximilien adds an extra constraint: for every student wearing their hat backward, the immediate neighbors in the same row must wear their hats straight. More precisely, if such a student has a neighbor directly to their left in the row, that neighbor must wear their hat straight; similarly, if they have a neighbor directly to their right in the row, that neighbor must also wear their hat straight.

As a loyal comrade of Maximilien and a zealous mathematician, you decide to help him determine how many different valid configurations of hats for each row satisfy this rule. As this number can be large, output it modulo $10^9 + 7$.

Input

The input consists of the following lines: · The first line contains two integers n and m ($1 \leq n, m \leq 10^5$), the dimensions of the parade block (there are n rows, each containing m students). · Each of the next n lines contains a single integer a_i ($0 \leq a_i \leq m$), the number of students in row i who must wear their hats backward.

Output

Output n integers in n different lines where the i-th integer represents the number of valid hat configurations for the i-th row that satisfy all the given constraints. As this number can be large, output it modulo $10^9 + 7$.

Samples

Sample input 1	Sample output 1
3 3 0 1 2	1 3 1