Marathon of Parallel Programming - 2023
Problem: Turmites

Wilton Jaciel Loch

February 27, 2023

Problem definition

Turmites are Turing Machine equivalents which have a state, direction and a position in a two-dimensional grid (or board) tape over which they operate. Given the state of the turmite and the color of the cell over which it resides a diferent set of actions may be taken. These actions are governed by a state transition table and can be diverse: including doing nothing, changing the color of the cell, turning to a particular direction and moving forward, etc.

Particular shapes and patterns will appear on the board after the execution of a certain number of steps. These may be the result of both chaotic and regular phases of board alterations by one or more turmites.

The presented algorithm implements a two-color version of the turmites with a finite two-dimensional grid. The state transition table is predefined and hard-coded, containing only two possible states for each turmite. The resulting board is printed using ASCII characters after all turmites finished the execution of all the steps.

Input

The input is comprised only of natural numbers (unsigned integers) and represents a single test case. The first line contains two integers r and c, respectively representing the number of rows and columns of the board. The second line contains an integer n defining the total number of steps that each turmite should perform. Following, an integer t representing the number of turmites to be placed on the board is read. Finally, t initial turmite positions are sequentially informed, each denoted by two integers x and y, with $0 \leq x < r$ and $0 \leq y < c$.

The input must be read form the standard input.

Output

The output contains r lines with c characters each. A line represents a row of the board, with white space characters corresponding to white color tiles and

asterisks corresponding to black color tiles.

The output must be written to the standard output.

Example

Input example 1	Output example 1
10 20	`*     *   **   ****`
1000	`*   **      * * * *****`
2	`      *****  *       **`
5 10	`** * *   *******   ***`
3 3	` ** ** **    * *   ***`
	`* ***    *   **   ***`
	`   * *     ***   * ** *`
	` *      *** *   *   ***`
	`* *******    *** **`
	`  **    **     * ** **`

Input example 2	Output example 2
20 40	`         * ** **     *     *`
10000	`    *    *          ** *            ***`
2	`    *  **  *      *** *        ** ****`
2 3	`*    * **  *** *  **  ***        *** **`
13 20	`*** *   *    **  ****  **  *   * *`
	`   **** ***** ** *  **       ** *  * * **`
	`  *** * * * ***  **      *  ***   **  *  *`
	`   **    * *   * **   *** * * *  ***   ** *`
	`  ****   *   *** * **  ** ******** *  **`
	`*    *** *** * **** *      ** *** **`
	` *   *    *  ** *   * *     *   ***     *`
	`   ***     *  * ** ** **  * ** **  *  *`
	`      **********   *  ***     *  * *`
	` *  ** ** ****    ** **       *       *`
	` *  *  *** ** **      *  * **  ** **     *`
	`*** *  *****      * * * *********** **`
	`   * ****    *** *      **        **`
	`***    * * *  ***   * *    **** **`
	`*       *  *** *** ** **     * *** *`
	`      * ** **  *        * **`