Programming Part: Nonograms

Problem definition

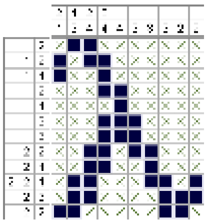
In this homework, you need to write a program to solve the game of nonogram. More info about the game can be found at Wikipedia (English, Chinese), or 漫談 nonogram (in Chinese). You can also google "nonogram" or "picross" to find interactive pages to play the game, such as here and there.

Write a program to output the map based on the input criteria. The input has the following format:

```
m n ← The numbers of rows and columns of the bit map (matrix)
a11 a12 . . . . . .  ← lengths of unbroken lines in row 1
a21 a22 . . . . . .  ← lengths of unbroken lines in row 2
.
.
.
am1 am2 . . . . . .  ← lengths of unbroken lines in row m
b11 b12 . . . . . .  ← lengths of unbroken lines in column 1
b21 b22 . . . . . .  ← lengths of unbroken lines in column 2
.
.
.
bn1 bn2 . . . . . .  ← lengths of unbroken lines in column m
```

The output format is as shown in the following test cases.

Here are some examples of nonograms:



One way to solve the problem is shown here. You can find many different methods, from simple to complex ones, to solve the problem.

Requirements & suggestions

- To solve the problem, you can simply use the maze traversal technique covered in the class, which uses a stack for DFS (depth first search).
- Since there is a time limit, brutal use of DFS may takes too long. So you should try the following strategies:
    a. Try to create an good initial guess of the map. Sometimes the initial guess can lead you to the answer directly. Here are some youtube tutorials:
        - Nonogram Tutorial
        - How to Play Picture Cross (Picross / Nonograms / Griddlers)
        - How To Do Nonogram Puzzles With Pencil and Paper
        - Pic-a-Pix Tutorial: How to solve a Pic-a-Pix logic puzzle

    Apply the trick systematically, for instance, row by row and column by column, and repeat the same procedure until the map converges. Try it over simple examples to make sure you understand the process.

    b. Start DFS if the initial guess does not lead to an answer. But remember to cut down as many infeasible braches as you can during DFS.

  If you are clueless about these strategies, please consult TAs ASAP.

- You can safely assume that
    a. 0 < m, n < 31.
    b. There is at least one solution to each test case.
    c. Each row and each column should have at least a black tile.
- Your program should take the input from the standard input and send the output to the standard output.
- The output may not be unique. As long as the output fulfils the given input, it is counted as a correct answer.
- TAs will use 5 test cases (with 2 out of the 8 basic test cases shown below, and 3 of 30x30 problems with similar difficulty as those shown in the 38 random test cases shown below) to test the correctness of your program, with a time limit of 60 seconds for each case.
- You need to write the program from scratch. You cannot use any open-source or readily available nonogram solvers.