MCS 253P - Lab 10

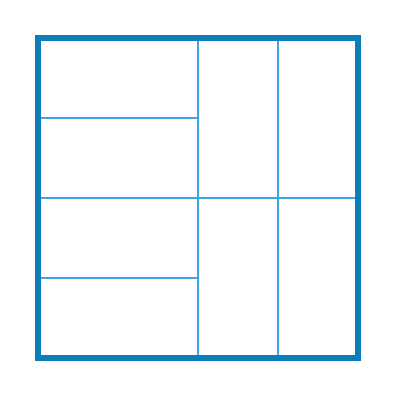
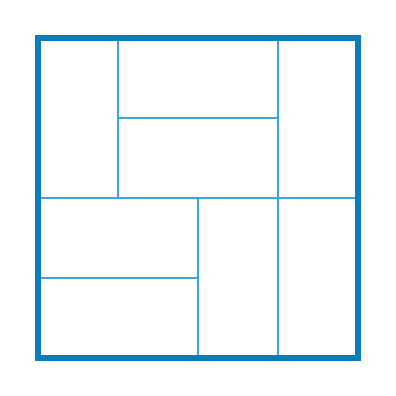
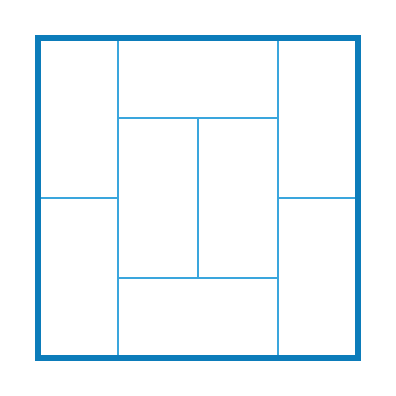
***Show edge-case testing in your report!***

* ***Please don’t use silly edge cases like empty inputs, etc. Instead think of interesting test inputs your program should elegantly handle.***
* ***WE WILL BE LOOKING FOR A SPECIFIC EDGE CASE!!! HOPE YOU INCLUDE IT =)***

**Lab 10 - Tiling Using Dominoes (150 pts + 50 pts = 200 pts)**

You are working for a tiling company that designs large rectangular tiles (called mega-tiles) that are patterned with smaller rectangles (called mini-tiles) within. Clients of your company provide dimensions for a mega-tile, and your company, in turn, provides them with possible designs which they can purchase.

For example, a client may wish to purchase a 4 x 4 mega-tile. Your company will provide examples of mega-tiles for purchase:



...etc

The client provides dimensions of their desired mega-tile by providing the width (W) and then height (H) of their desired mega-tile. To keep a standard orientation, we will think of the width as the length spanning left/right, and the height as the length spanning up/down. Each mini-tile is always 2 x 1 or 1 x 2.

**(150 pts)** Your job is to implement an efficient program to complete the following **two** tasks…

1. calculate the **number of possible mega-tile designs**
   * rotationally equivalent designs still count as different designs
2. construct and output a **design matrix for *each* possible design**. The design matrix (we will name simply as design) should be a design[H][W] matrix (namely have H rows and W columns). Each cell of the matrix will contain a number representing the identity of the mini-tile covering that position of the mega-tile.
   * for example, the design matrix for the left-most design shown above can be:

0 0 4 5

1 1 4 5

2 2 6 7

3 3 6 7

Input:

* two integers representing the W and H, respectively, of the mega-tile to be designed

Output:

* the number of possible designs
* for each design, a design matrix illustrating the design

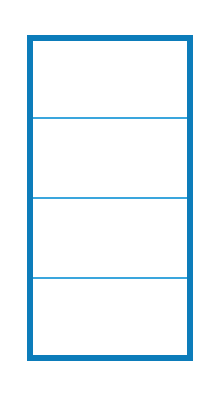
Example:

input:

2 4

output:

5 (you don’t need to output images)

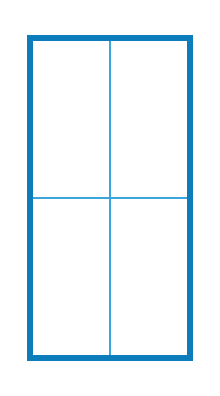


0 0

1 1

2 2

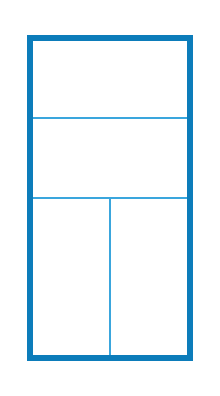
3 3



0 2

0 2

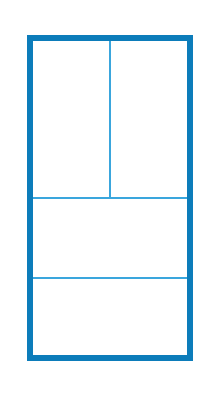
1 3

1 3

0 0

1 1

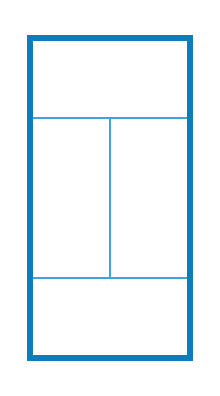
2 3

2 3

0 3

0 3

1 1

2 2

0 0

1 3

1 3

2 2

Test Inputs:

test input 1:

3 2

test input 2

4 5

*Note*: for test input 2, you may include only the first several design matrices as long as it is clear they were generated by your program):

**(50 pts)**

***On the last page of your report, please also include a high-level (but specific) description of your algorithm (ie. what kind of algorithm is it, and specifically how does it work). Include any analysis as compared to other algorithms you considered.***