To run the program manually

How to run the program:

Before running the program, you have to compile the 3 source files that I provide. (Please use the StdDraw from the source file, otherwise the outcomes may be different because I modify the StdDraw a little bit.)

How to modify the board size:

To modify the board size, you can do it by change the first argument when you call the tileBoard().

Note: If n is the first argument, it mean that the board will have the size of 2^n*2^n

Decoration:

If you want to see the colorful board, please put true in the last argument when you call tileBoard().

Screen size:

You can set the application display size by calling the method setScreenSize(int n), the application display size will be set to n*n pixels.

You can put the hole anywhere you want:

You can simply put the hole anywhere by put in x-coordinates and y-coordinates after the board size argument. (If you don't put the coordinates in, the coordinates will be random)

Note: The coordinates can't be more than $2^n - 1$ (n is the argument that you put in at first), for example, if $n = 5$, the coordinates can be any number from 0 to 31.

** There will be an example and more explanation below. **

To run the program automatically:

How to run the program:

Just compile the source files and run the BoardTilingApp class.

The display size is 800*800 pixels. (Default)

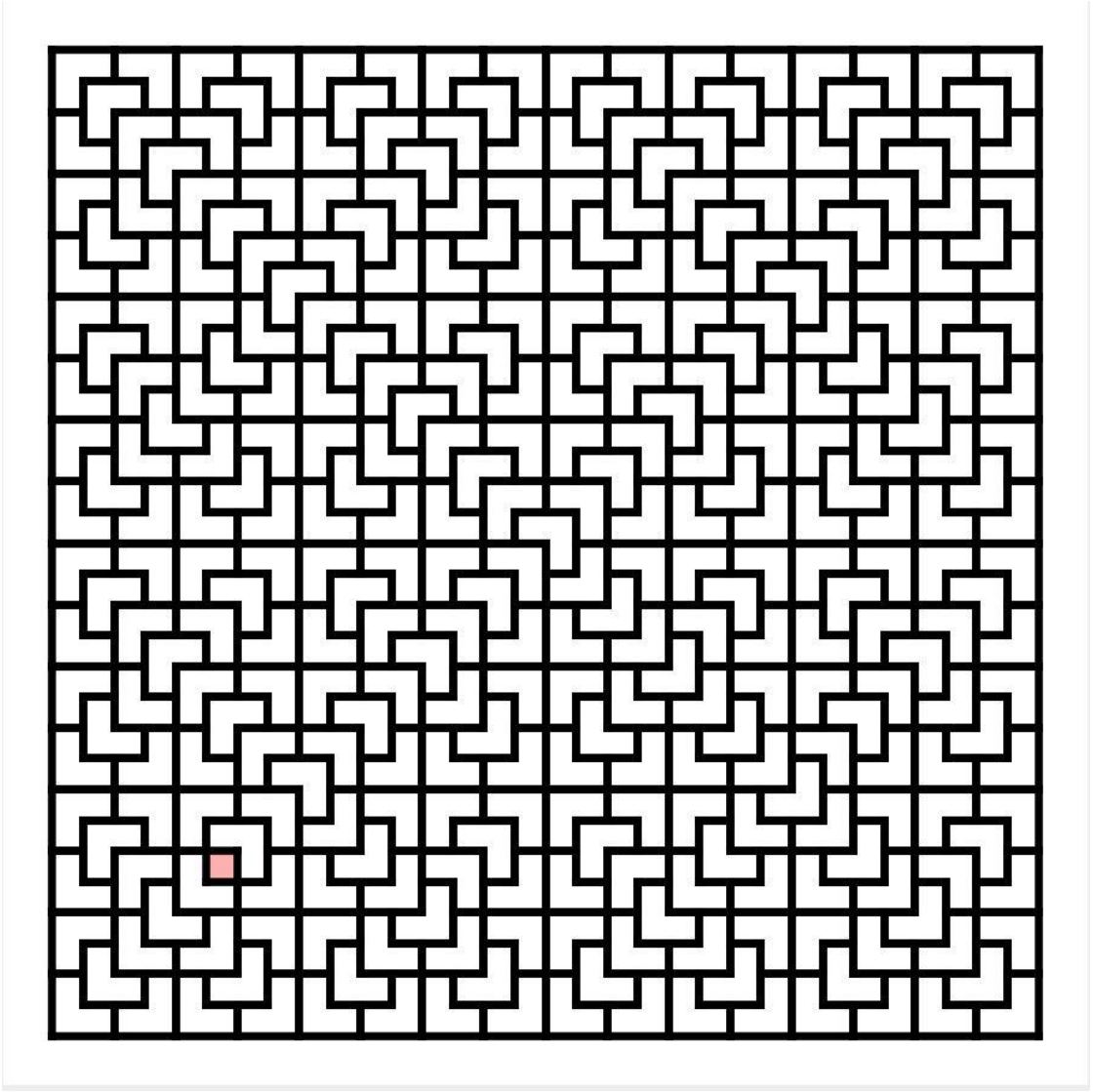
Note: the rules are the same as run the program manually.

Explanation:

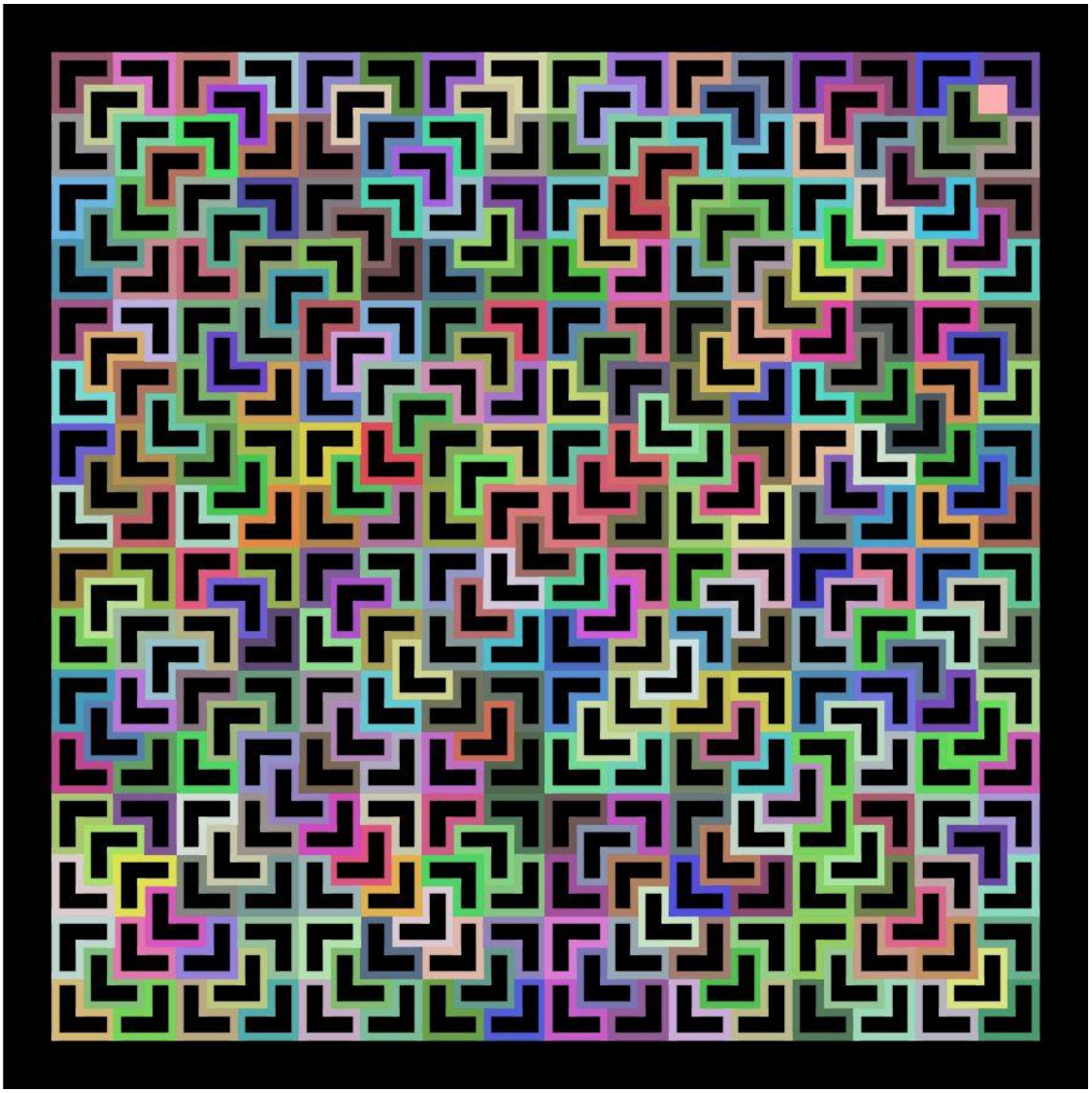
```
/**
 * set display size to n * n pixels
 * Note: this must be called before tileBoard, otherwise it will not do its job.
 */
BoardTiling.setScreenSize(n);
/**
 * n is the board size,and the board is square with 2 to the nth power on each side.
 * x is the x-coordinate of the hole.
 * y is the y-coordinate of the hole.
 * decoration is the boolean that will turn on or turn off the decoration,
 * true will turn on the decoration, false for no decoration.
 */
BoardTiling.tileBoard(n, x, y, decoration);
/**
 * the argument is the same as above, but this will random the coordinates
 * of the hole.
 */
BoardTiling.tileBoard(n, decoration);
/**
 * 7 is the board size, with 128 tiles on each side.
 * 30 is the x-coordinate of the hole.
 * 25 is the y-coordinate of the hole.
 * true is turn on the decoration.
 */
BoardTiling.tileBoard(7, 30, 25, true);
/**
 * 8 is the board size.
 * false is no decoration.
 * the hole coordinates will be random.
 */
BoardTiling.tileBoard(8, false);
```

Example:

1. BoardTiling.tileBoard(5, 5, 5, false)



2. BoardTiling.tileBoard(5, 30, 30, true);



3. Run automatically

```
Please input the size: 5
Please input the x-coordinate (-1 if you want it to be random): 5
Please input the y-coordinate (-1 if you want it to be random): 5
Do you want decoration (Y/N): Y
Generate board tiling with 32 tiles on each side, hole at (5 , 5) with decoration.
```