Problem Set 2 - Arrays, matrices and the Game of life

The reference-book makes reference to the **Matrix** class, representing a two dimension array, and the **Game of Life**. During this lab, you will be asked to work with these classes to better grasp the use of arrays.

Assignment 0

Transcribe the classes necessary to work with Matrices:

- **`adt_array.py`** - containing the code to efficiently manage an array
- **`adt_matrix.py`** -containing the code to efficiently manage a matrix

Transcribe the Game of Life example from the reference-book:

- **`lifegrid.py`** - containing the code to play the Game of Life

Is important that you read the code that has been provided and you understand its content.

ATTENTION: do not call the array module (.py file) “array” it causes problems, call it “adt_array”

ATTENTION 2: there may be a few mistakes in the code of the book, fix them!

Assignment 1

Complete the Matrix examples by adding the following functionality (*self* is omitted):

1. **`__sub__(other)`** - The same as the add() operation but subtracts the two matrices.
2. **`__mult__(other)`** - Creates and returns a new matrix that is the result of multiplying this matrix to the given rhsMatrix. The two matrices must be of appropriate sizes as defined for matrix multiplication.
3. **`transpose()`** - Returns a new matrix that is the transpose of this matrix.

Develop a simple main to showcase these new functionality.

Assignment 2

Implement the **`numLiveNeighbors(self, row, col)`** method of the **LifeGrid class**.

Implement the **`evolve(self, generation=1)`** method in the **LifeGrid class** such that it takes the number of generations to evolve (by default use 1).

Assignment 3

Complete the implementation of the **`gameoflife.py`** program by implementing the **`draw(self)`** function. The output should look similar to the following, where dead cells are indicated using a **period** and live cells are indicated using the **# symbol**.

Assignment 4

Use your program from Exercise 2.4 to experiment with the initial configurations shown in **Figure 1**. Answer the following questions for each configuration using a variety of grid sizes and assuming no more than 10 generations.

a) Does the configuration die out?
b) Does the configuration become stable?
c) Does the configuration become an oscillator?
d) How many generations were required before each configuration resulted in one of the states indicated in parts (a) – (c)?

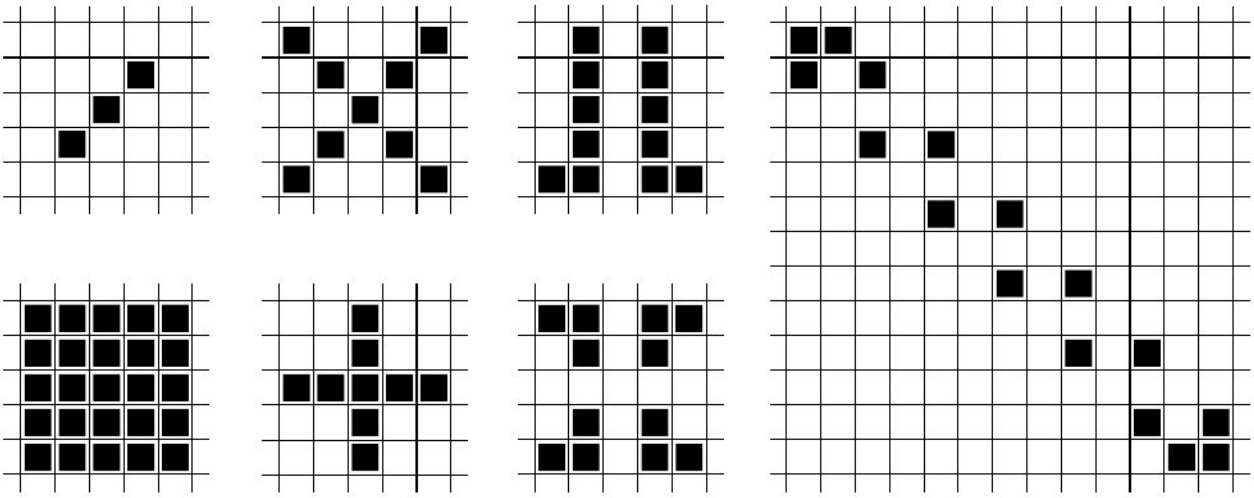

Figure 1 - Sample game of Life configurations.