CMPE 436: Assignment 1
Fall 2018   Instructor: Alper Sen
Due Date: Oct 8 Demo Date: TBD

The goal of this programming assignment is to familiarize you with Java programs and Eclipse IDE.

Question 1- (40 points)
Write a **sequential (no threads)** Java program to multiply two matrices. Your program will read the contents of two matrices from data files, multiply the two matrices, and write the resulting matrix both to a third file and to the console. **File Format:** A matrix data file is a text file consisting of *all integer values* formatted something like this

R	C		
M_{11}	M_{12}	...	M_{1C}
M_{21}	M_{22}	...	M_{2C}
...	...	...	...
M_{R1}	M_{R2}	...	M_{RC}

R is the number of rows, C is the number of columns, and the M's are the data themselves. All numbers are separated by one or more spaces. This format is used for both reading and writing a matrix. The three run arguments to your code are:

> Name of file containing the first matrix. Name of file containing the second matrix. Name of file into which to store the result matrix.

If the input matrices cannot be multiplied because their dimensions are not compatible, display a descriptive message explaining the problem and terminate the program. Check other potential error conditions as well.

Question 2- (60 points)
Write a multi-class **sequential (no threads)** Java program that simulates the *game of life*. The simulation occurs on an M-by-N rectangular array of cells. Each cell has only two values: 0 and 1. Each cell has eight neighbors: up, down, left, right, and four diagonally. Cells on the four edges of the grid have fewer than eight neighbors, of course. If the grid is declared as

> int[][] grid = new int[N][N];

then the neighbors of an "interior cell" grid[i][j] are

> grid[i-1][j], grid[i+1][j], grid[i][j-1], grid[i][j+1],
> grid[i-1][j-1], grid[i-1][j+1], grid[i+1][j-1], grid[i+1][j+1]

The game works like this. Fill the grid with the initial values. Then for each cell in the grid, compute the new value that the cell becomes in the next generation of the game. All changes occur simultaneously, called a generation.

- A 1 cell value stays 1 if exactly two or three neighbors are 1 valued.
- A 1 cell value goes to 0 if less than two or greater than three neighbors are 1 valued.
- A 0 cell value goes to 1 if exactly three neighbors are 1 valued.
- A 0 cell value stays 0 if less than three or greater than three neighbors are 1 valued.

Input Data: The input data to your Java program is

- M (rows) and N (columns), the size of the rectangular grid;
- maxGenerations, the number of generations to simulate;
- the initial values of the cells (aka initial population), which can be user-specified in a file (input.txt) or random.

If the only input data is M, N and maxGenerations, then initialize the grid to random values (0 or 1). If the input data is M, N, maxGenerations and a file name (input.txt) then you may read initial values of the cells from the file (We will post an input file reading code as a reference but you may develop your own as well). Assume that input.txt has correct number of entries corresponding to M and N values. Remember to check for illegal input values, like negative numbers or cell initial values that are not 0 or 1. Here are some sample input data sets to try.

This initial population dies out.	This initial population dies more slowly.
0 0 0 0 0	0 0 0 0 0
0 1 0 0 0	0 1 0 0 0
0 0 1 0 0	0 1 0 0 0
0 0 0 1 0	0 0 1 0 0
0 0 0 0 0	0 0 0 0 0
This initial population reaches a stable state.	This initial population oscillates.
0 0 0 0 0	0 0 0 0 0
0 1 1 0 0	0 0 1 0 0
0 1 0 0 0	0 0 1 0 0
0 0 0 0 0	0 0 1 0 0
0 0 0 0 0	0 0 0 0 0

Guidelines:
1- Email your assignment solution.
2- Add the following to the start of your programs.
   // your name // your student ID // your email address
   // CMPE436-Assignment *n* - where *n* is the assignment number (1, 2, ...)
3- Add comments to your programs. Program clarity is very important. You get graded on this.
4- Also add a README.txt file to explain your programs.
5- Demo your homework to the instructor. Bring your laptop.
6- DO NOT DISCUSS WITH YOUR CLASSMATES. DO NOT USE SOLUTIONS FROM OTHERS. CHEATING WILL NOT BE TOLERATED.