CMPS12A Introduction to Programming - Spring 2005
Programming Assignment 4

Due May 20, 2005, 9pm
(Late submissions will not be accepted.)

Objective: arrays.

Assignment: The goal of this assignment is to implement John Conway’s Game of Life. Contrary to what the name suggests, the Game of Life is not really a game. It is a cellular automaton devised by the British Mathematician John Conway in 1970 which simulates birth, death and survival of organisms based on certain rules. (You are encouraged to read more about the Game of Life from the following URL http://en.wikipedia.org/wiki/Conway%27s_Game_of_Life.) The Game of Life is played on a grid of square cells. In the actual game, this grid of square cells extend infinitely in every direction. For this assignment, however, we assume the Game of Life is played on a 10 x 10 grid of cells. Every cell can either be alive or dead. A cell that is alive is denoted by a “*”. Otherwise, the cell is dead. Every cell in the grid, except for those at the boundaries, has a neighborhood consisting of eight cells. For example a cell at coordinate (3,4) has eight neighborhood cells (2,3), (2,4), (2,5), (3,3), (3,5), (4,3), (4,4), and (4,5) while a cell at the boundary (0,0) has only three neighborhood cells (0,1), (1,0) and (1,1).

The rules of the Game of Life are shown below:

- **Birth.** A dead cell with exactly three live neighbors becomes a live cell in the next generation.
- **Survival.** A live cell with two or three live neighbors stays alive in the next generation.
- **Death.** In all other cases, a cell dies or remains dead in the next generation.

It is important to note that the number of live neighbors is based on the state of the grid before the rules are applied. In other words, a new generation of 10x10 grid of cells is generated based on the previous generation of 10x10 grid of cells.

In this assignment, you will implement John Conway’s Game of Life on a 10x10 grid of cells. You are to ask the user for the initial live cells and prompt the user on whether she wants to continue with the simulation until the user chooses to quit. Some sample runs are shown below.
Enter the coordinates of live cells for the first generation.
Enter the row ('-1' when done): 4
Enter the column: 3
Enter the row ('-1' when done): 4
Enter the column: 4
Enter the row ('-1' when done): 4
Enter the column: 5
Enter the row ('-1' when done): -1
The grid is initialised to:

```
0 1 2 3 4 5 6 7 8 9
+-+-+-+-+-+-+-+-+-+-+-+-
0| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+-+-
1| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+-+-
2| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+-+-
3| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+-+-
4| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+-+-
5| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+-+-
6| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+-+-
7| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+-+-
8| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+-+-
9| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+-+-
```

Press ‘0’ to exit or ‘1’ to continue to the next generation: 1

```
0 1 2 3 4 5 6 7 8 9

0 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
1 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
2 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
3 | | | | |*| | | | |
---|---|---|---|---|---|---|---|---|---
4 | | | | |*| | | | |
---|---|---|---|---|---|---|---|---|---
5 | | | | |*| | | | |
---|---|---|---|---|---|---|---|---|---
6 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
7 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
8 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
9 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
```

Press ‘0’ to exit or ‘1’ to continue to the next generation: 0

```
0 1 2 3 4 5 6 7 8 9

0 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
1 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
2 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
3 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
4 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
5 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
6 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
7 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
8 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
9 | | | | | | | | | |
---|---|---|---|---|---|---|---|---|---
```

Press ‘0’ to exit or ‘1’ to continue to the next generation: 0

os-prompt>

os-prompt> java GameOfLife

3
Enter the coordinates of live cells for the first generation.
Enter the row (’-1’ when done): 5
Enter the column: 1

Enter the row (’-1’ when done): 4
Enter the column: 1

Enter the row (’-1’ when done): 4
Enter the column: 2

Enter the row (’-1’ when done): -1
The grid is initialised to:
Enter the coordinates of live cells for the first generation.
Enter the row (‘-1’ when done): 9
Enter the column: 0

Enter the row (‘-1’ when done): 8
Enter the column: 1

Enter the row (‘-1’ when done): 7
Enter the column: 1

Enter the row (‘-1’ when done): 8
Enter the column: 2

Enter the row (‘-1’ when done): 9
Enter the column: 2

Enter the row (‘-1’ when done): -1
The grid is initialised to:
Press '0' to exit or '1' to continue to the next generation: 1

0123456789
+-----------------------+
0|*|*|*|*|*|*|*|*|*|
+-----------------------+
1|*|*|*|*|*|*|*|*|*|
+-----------------------+
2|*|*|*|*|*|*|*|*|*|
+-----------------------+
3|*|*|*|*|*|*|*|*|*|
+-----------------------+
4|*|*|*|*|*|*|*|*|*|
+-----------------------+
5|*|*|*|*|*|*|*|*|*|
+-----------------------+
6|*|*|*|*|*|*|*|*|*|
+-----------------------+
7|*|*|*|*|*|*|*|*|*|
+-----------------------+
8|*|*|*|*|*|*|*|*|*|
+-----------------------+
9|*|*|*|*|*|*|*|*|*|
+-----------------------+
Press '0' to exit or '1' to continue to the next generation: 1

0123456789
+-----------------------+
0|*|*|*|*|*|*|*|*|*|
+-----------------------+
1|*|*|*|*|*|*|*|*|*|
+-----------------------+
2|*|*|*|*|*|*|*|*|*|
+-----------------------+
3|*|*|*|*|*|*|*|*|*|
+-----------------------+
4|*|*|*|*|*|*|*|*|*|
+-----------------------+
5|*|*|*|*|*|*|*|*|*|
+-----------------------+
6|*|*|*|*|*|*|*|*|*|
+-----------------------+
7|*|*|*|*|*|*|*|*|*|
+-----------------------+
8|*|*|*|*|*|*|*|*|*|
+-----------------------+
9|*|*|*|*|*|*|*|*|*|
+-----------------------+
Press '0' to exit or '1' to continue to the next generation: 1

0 1 2 3 4 5 6 7 8 9
+-+-+-+-+-+-+-+-+-+-+
0| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
1| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
2| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
3| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
4| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
5| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
6| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
7| |*|*| | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
8| | | |*| | | | | | |
+-+-+-+-+-+-+-+-+-+-+
9| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
Press '0' to exit or '1' to continue to the next generation: 1

0 1 2 3 4 5 6 7 8 9
+-+-+-+-+-+-+-+-+-+-+
0| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
1| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
2| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
3| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
4| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
5| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
6| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
7| |*|*| | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
8| | | |*| | | | | | |
+-+-+-+-+-+-+-+-+-+-+
9| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
Press '0' to exit or '1' to continue to the next generation: 1

0 1 2 3 4 5 6 7 8 9
+-+-+-+-+-+-+-+-+-+-+
0| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
1| | | | | | | | | | |
+-+-+-+-+-+-+-+-+-+-+
|   |   |   |   |   |   |
|---+---+---+---+---+---|
|   |   |   |   |   |   |
|   +---+---+---+---+---+---|
|   |   |   |   |   |   |
|   +---+---+---+---+---+---|
|   |   |   |   |   |   |
|   +---+---+---+---+---+---|
|       |       |       |       |       |
| Press '0' to exit or '1' to continue to the next generation: 1 |

|   |   |   |   |   |   |
|---+---+---+---+---+---|
|   |   |   |   |   |   |
|   +---+---+---+---+---+---|
|   |   |   |   |   |   |
|   +---+---+---+---+---+---|
|   |   |   |   |   |   |
|   +---+---+---+---+---+---|
|       |       |       |       |       |
| Press '0' to exit or '1' to continue to the next generation: 1 |

|   |   |   |   |   |   |
|---+---+---+---+---+---|
|   |   |   |   |   |   |
|   +---+---+---+---+---+---|
|   |   |   |   |   |   |
|   +---+---+---+---+---+---|
|   |   |   |   |   |   |
|   +---+---+---+---+---+---|
|       |       |       |       |       |
| Press '0' to exit or '1' to continue to the next generation: 1 |
Submitting Your Program

Follow these instructions carefully and you will score some easy points:

1. Make sure the text output of your program matches that of the sample runs above exactly. Where a `+` symbol is used as the corner of a cell, you should use a `+` symbol, where a dash is used, use a dash, etc..

The wording of the prompts to the user should also match the specification precisely. Make sure you do not change a word like ‘exit’ to ‘quit,’ and that you match the spacing too.

2. Submit your program to the assignment labeled “Program 4: Conway’s Game of Life” through your account on WebCT.

3. Submit exactly two files:
   - GameOfLife.java - your source code, which should contain a class called GameOfLife. The class GameOfLife should define your main method, which will have the signature “public static
void main(String[] args)”. If you choose, you can define other classes in the file and you can use them in your program. But only the class GameOfLife will have a main method.

- GameOfLife.class - your compiled program.

4. If you are working with a partner, only one of you should submit the assignment. But your source code should list both of you as authors.

Start Early!

This assignment has some complexity, and to write a good program you will have to plan it out. If you start early you will have time to make mistakes and learn from them. The best programs we have seen so far have come from people who started working as soon as the assignment was released. And you’ll have a lot more fun with it too!