

CSCI 136 C++ programming

Lab exercise 12

Saad Mneimneh
Computer Science
Hunter College of CUNY

Two-dimensional arrays and the game of life

PART 1:

Write a program to perform the following:

- Declare a 50 by 50 two-dimensional array a of bools and initialize it randomly. Each $a[i][j]$ represents a cell that can be either alive (true) or dead (false)
- Repeat forever
 - for each cell
 - * if the cell is alive and has ≤ 1 or ≥ 4 alive neighbors, the cell dies
 - * if the cell is dead and has exactly 3 alive neighbors, it becomes alive
 - display the updated array
 - wait for the user to press Return and clear the screen (see code below)

Note that updating a cell should not affect neighbor information for other cells until the next iteration. Therefore, one way is to repeatedly copy the entire array a into another array b , and use neighbor information in b to update a .

To wait for user input and clear the screen use the following code (include `cstdlib`):

```
char x;  
cin.readline(&x,1);  
system('clear');
```

PART 2 (for fun):

Instead of initializing the array randomly, try to initialize it with patterns such that:

- all cells will eventually stabilize and stop changing
- all cells will eventually die
- the cells keep on changing indefinitely