

Intro to Computers for Engineers (14:440:127)

Final Project Information, Summer 2018

You will need to complete a final project in this class. You will complete this project in a group of 4-5 students. As a group, you will choose one of the project options below. It will then be up to you to do the necessary work to meet the project deliverable level you are aiming for (an A, of course) by the project deadline.

The project has the following important dates:

Week of 7/24: You are assigned teams.

Ongoing After Teams are Assigned: Meet with your instructor in class for 10 minutes as a team each class meeting. Tell your instructor what you worked on, what you need help with, and your plan for getting that help (office hours, finding resources, etc.).

Last day of class: Final projects due by the end of the day. They should include:

- All project code.
- A 2-3 page write-up describing the development process behind your project:
 - o What new concepts did you learn, as a team?
 - o What were the greatest challenges?
 - o What would you like to have added, given more time?
 - o What would you have done differently?
- A 1-2 page **manual** for your instructor. *This document is important, as it tells me how to view the deliverables of the project, thus assign them a grade. Be as clear and simple as possible with these instructions. Give them to a friend first to try out, if you need to.*
- Citations of external sources. You can use external sources, so long as they **do not directly provide you with the code you need to complete a deliverable**. Any citation format is acceptable.
- An honor statement signed by the entire team.
- Completion of the team evaluation survey.

Grading Breakdown:

Project write-up: 30%

Project deliverables: 60%

Team evaluation survey: 10%

Final Project: Game of Life

Th

C Level

1. Write a function that takes a matrix as an input, and returns a number as an output. That number should be the sum of all of the elements of the input matrix. Please use no built in functions for this.
2. Write a function that takes a matrix as an input and returns a matrix as an output. Your function should create a “ring” of zeros around your input matrix and return it as your output. For example, the matrix

| | | | | |
|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 |

Would become:

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Note that your output has two more rows and two more columns than your input.

B Level

You should call the previous functions you wrote to more easily complete this section of the project.

For this problem, your program **will have two inputs:**

- `mat`, a 2D binary matrix of 1's and 0's (this matrix contains numbers that can only be 0 or 1).
- `n`, a number of iterations to perform Conway's Game of Life.

It should have one output:

- `finalMat`, the final binary image after `n` iterations of Conway's Game of Life.

Conway's Game of Life is an iterative “game” with no user input. Each round of the game, the matrix representing living organisms gets updated. The game has the following rules:

- Pixels with the value of 1 are alive.
- Pixels with the value of 0 are dead.
- Pixels that are alive during this iteration with 2-3 living neighbors (not including themselves) stay alive for the next iteration. **Note that the numNeighbors program returns neighbors including the pixel itself.**
- Pixels that are dead during this iteration with exactly 3 neighbors turn alive for the next iteration.
- All other pixels are dead in the next iteration.

These rules should be applied to update the image matrix for each iteration of the game, until n iterations have been performed.

For example, the result of the following input `mat`:

```
0    0    0    0    0
0    0    0    0    0
0    1    1    1    0
0    0    0    0    0
0    0    0    0    0
```

`finalMat` would be identical to `mat` for any even input `n`, and for any odd input `n` would look like this:

```
0    0    0    0    0
0    0    1    0    0
0    0    1    0    0
0    0    1    0    0
0    0    0    0    0
```

Meaning that the game “oscillates” between different shapes.

A Level

Create a GUI that lets the user set up an initial matrix and run the Conway’s game of life algorithm. You can decide how to accomplish this, so long as you create a GUI with MATLAB’s GUIDE tool.