

Build Your Own Final Assignment



Cary Jardin · Nov 29, 2021 · Notified 40 people

Motivation Of Assignment:

Using and creating Algorithms and Data Structures are a daily part of Computer Science and programming. Over this semester we have talked about Time complexity in big O notation, seen how even simple algorithms can out perform more complicated ones, and how finding the perfect answer can be optimized into a "good enough" solution.

But above all, it is about your own growth and journey. Exploring, growing an opinion, and using the tools you have to be successful.

Objective of Assignment:

- Build something that interests you!
 - Your code tells a story, and shows how interested you are in what you are doing.
- Attempt something that pushes you to think about things like:
 - Data Structure Usage
 - Time Complexity
 - How "Best" to attack a problem.
- Build your resume

Scope:

Final Project!!

Submitting Your Assignment:

Email me the following:

- Email me when you are done including an URL to your repo so I can see your code.

~

- Screen recording showcasing your code working, and discussing your work.

How the Assignment will be Graded:

I am not going to give you an A-F grading. I will grade each individually based on the following criteria.

- Usage of concepts lectured throughout the class
- Breadth and Depth of Functionality
- Creativity and Novelty
 - Will negatively grade "cut, paste, and submit" assignments

Completeness:

You MUST check your code into your own fork so I can see it, as well as email your screen recording showcasing your work.

Assignment Due Date:

By Cal State Rules, all take home finals must be completed by the end of the scheduled Final Data/Time which is 12/17/2021, Friday 3:45PM

Ideas To Get You Started:

- Extend the Artificial Neural Network We Worked on in Class
 - Implement Back Propagation
 - <https://en.wikipedia.org/wiki/Backpropagation>
 - Explore and experiment with your own training methods
- Implement Assignment 2 Pathing using:
 - A*
 - https://en.wikipedia.org/wiki/A*_search_algorithm
 - Dijkstra's
 - https://en.wikipedia.org/wiki/Dijkstra%27s_algorithm
 - Map size needs to be at least 128x128 to be interesting :)
- Attempt Solving a Problem on the DWAVE Quantum Computer

- Free DWave Account
 - <https://cloud.dwavesys.com/leap/signup/>
- Some ideas
 - <https://github.com/dwave-examples/nurse-scheduling>
 - https://docs.ocean.dwavesys.com/en/stable/examples/map_coloring.html
- Survey of Sorting Algorithms
 - Create a bunch of random data
 - Implement at least 2 sorting algorithms
 - The more compressive your survey, the higher your grade
 - Again.. not looking for "cut, paste, and submit" assignments
 - Run them a bunch of times
 - Graph the results and the expected results from the Big O complexity of the algorithm
- Purpose your own Idea!

I am available to consult on ideas, answer questions, and help however I can. The goal is for this to be something you can be proud of, and show off as a part of your resume.

