

Create a subdirectory of your home directory called `final-project`. Within this directory, create a C project with a topic of your choosing. This assignment will be due on August 2, 2019 at the beginning of class. See below for requirements. Remember that this project will be worth 35% of your final grade, so put in as much effort as possible. As soon as you have chosen a topic, which you should do by the end of the week, email me to confirm or get pointers.

Requirements:

- The assignment should be large enough to justify three weeks worth of work. You should spend at least 3 hours per week on the project, so the final project should be the equivalent of 9 hours of work.
- A `README` file should be present within your `final-project` directory that describes both the topic / purpose of your project, as well as how to use your program. Also included in the `README` file should be a description of what each file in your project contains. See the `README` file for the Caesar cipher in-class project for an example.
- Your code should be organized in multiple files as necessary. Proper indentation and formatting must be followed, as the code needs to be readable in order for a grade to be issued.
- Your code should be heavily commented to convey the meaning behind blocks of code and the purpose of functions. Any code that is the result of research on your part should be cited with a resource in your comments.
- A `Makefile` should also be included for ease of compilation for both myself during grading and for yourself during implementation.

Presentations:

- An optional presentation can be prepared for extra credit and presented to the class. This presentation will raise your letter grade up to 5%.
- The presentation should be between 10 and 15 minutes long. The format of the presentation is up to you.
- You can describe the history/relevance of your topic, how you implemented it, demo your project, etc.
- Bonus points will be assigned based on how much I learned about your topic.

Grading Rubric:

- 25%: Code readability
- 25%: Code organization
- 25%: Quality of work
- 25%: Documentation

Topic Suggestions:

- Make a simple game e.g. a text based adventure, dungeon crawler, etc. (look into the curses header file)

- Some form of data processing e.g. simple linear regression classifier
- Implement an interesting algorithm / data structure and apply it to some problem. See here for some ideas.
- Make either your own system program or an original implementation of an existing one e.g. make your own `ls` program.

I will collect your projects in class on August 2, 2019. The last day of class will be devoted to any students wishing to present their projects to the class for extra credit.