Software Project Description

Your task will be to create an original software project of your choosing that is related to the subject area you chose in part 1. There is no restriction on which language(s) are used for this project, except that I must be able to easily run your code after it is turned in. For instance, you shouldn't do your project in Matlab, because I can't afford Matlab, and therefore I wouldn't be able to run your code. Throughout your project you will attempt to use solid software development principles.

Requirements

- Choose a topic for a software project relating to your chosen subject area.
- Implement the project in your language(s) of choice. The project should be built with the intent that someone would want to use the software for something. It shouldn't be a simple toy program, similar to those that are assigned in class. There should be an end user in mind.
- You must choose 3 software development best practices to use throughout your project. These can be either language independent or language specific. For instance, if I am implementing a project in Go, the 3 best practices could be version control (using git maybe), unit testing, and proper code documentation as described in the Practical Go Code documentation page.
- You must include an original implementation of some algorithm or data structure. The choice is up to you. I am not asking you to invent an algorithm or data structure, but simply to implement one without copying code.
- Finally, include a README file with the following information:
  - Explanation of what the program is and how it is intended to be used.
  - Directions on how to run the program, as well as how to use it.
  - Explanation of chosen best practices and how they were used.
  - Explanation of chosen algorithm / data structure and how / why it was used.

Grading

This project will be graded based on the following criteria:

- 20 points: Code readability and organization
- 20 points: Code documentation
- 20 points: Best practices used
- 20 points: Algorithm / data structure implementation
- 20 points: README file
- 20 bonus points: Awesomeness