What is a README?

Every programming project in this class requires that you submit a corresponding README file. One of the common questions that we get is what is a readme file? This page hopes to answer all of your questions.

According to [makeareadme.com](https://www.makeareadme.com/) (https://www.makeareadme.com/), a README file "is a text file that introduces and explains a project. It contains information that is commonly required to understand what the project is about" and how to use it. You should spend some time and read through the material on this website since they say it much better than I can.

The standard format for README files today is markdown. Markdown is a lightweight markup language for creating formatted text using a plain-text editor. The advantage to this is that we can have a nicely formatted file but have the flexibility of plain-text (and thus can tag lines on the Handins server). Here's a couple of references on how to write a readme in markdown:

- **https://www.markdownguide.org/getting-started/ (https://www.markdownguide.org/getting-started/)**
- **https://guides.github.com/features/mastering-markdown/ (https://guides.github.com/features/mastering-markdown/)**

What should I put in my README?

Break your README into sections using titles:

1. **About/Overview.** Give a general overview of the problem and how your program solves the problem.

2. **List of features.** List all features that are present in your program.

3. **How To Run.** Instructions to run the program should include the following:

    - How to run the jar file
    - What arguments are needed (if any) to run the jar file, what do they mean

4. **How to Use the Program.** Instructions on how to use functionality in your program. if interactive, how to interact with your program? Pay particular attention to the parts that are not part of the example runs that you provided.

5. **Design/Model Changes.** It is important to document what changes that you have made from earlier designs. Why were those changes required? You can write these changes in terms of version if you wish.

6. **Assumptions.** List what assumptions you made during program development and implementation. Be sure that these do not conflict with the requirements of the project.

7. **Limitations.** Limitations of your program if any. This should include any requirements that were not implemented or were not working correctly (including something that might work some of the time).

8. **Citations.** Be sure to include any citations that are required for your project. Citations should include references (paper, website, etc.) for any site that you used to research a solution. Proper APA format should be used. For websites this includes the name of the website, title of the article, the url, and the date of retrieval. If you have nothing to cite, you should indicate this.