Data Structures and Algorithms in Python

Final Project – Due May 31, 2024

For your final project, you will either expand upon one of the five projects you have completed throughout the year or propose a new, original project. This project should involve significant programming work and demonstrate your understanding of many concepts covered in this course. Python is the preferred language, but you may use another language if it better suits your project.

Project Requirements

Project Proposal

- Description: Provide a detailed description of your project idea. If building on a previous project, explain how you will extend or enhance it.
- Objectives: Clearly outline the goals of your project.
- Methodology: Describe the steps you will take to achieve your objectives.
- Feasibility: Ensure your project is realistic and achievable within the given time frame.
- Your proposal should be submitted in-person or over email for approval by Friday, May 17.

Programming Component

- Language: Primarily Python, but other languages are allowed if justified.
- Complexity: Ensure your project involves a substantial amount of programming. It should be comparable to or more complex than the five previous projects.
- Functionality: Your program should be fully functional, demonstrating the concepts and skills you have learned throughout the year.

Documentation

- Code Comments: Include comments in your code to explain your logic and decision-making.
- User Guide: Provide a brief guide on how to run and use your program. This can be in the comments of your Python code.
- Technical Report: Write a report detailing your project, including the project goal(s), methodology, results, and any challenges faced.

Project Rubric

Project Proposal

Excellent *9-10 pts*	Good *8-9 pts*	Satisfactory *7-8 pts*	Needs Improvement *0-7 pts*
Detailed and well-structured proposal with clear objectives, methodology, and feasibility.	Clear proposal with defined objectives and methodology, minor details missing.	Adequate proposal but lacks some clarity or detail.	Missing, or vague proposal lacking detail or feasibility.

Documentation

Excellent *9-10 pts*	Good *8-9 pts*	Satisfactory *7-8 pts*	Needs Improvement *0-7 pts*
Comprehensive documentation including well-commented code and detailed project report.	Adequate documentation with minor gaps, including clear code comments and a project report.	Basic documentation with some clarity or detail issues.	Incomplete or unclear documentation lacking essential information.

Programming

Excellent *75-80 pts*	Good *70-75 pts*	Satisfactory *50-70 pts*	Needs Improvement *0-50 pts*
High-quality, well-organized code demonstrating significant complexity and functionality. You demonstrate mastery of a variety of concepts covered during class this year.	Functional and organized code with minor issues in complexity or clarity.	Functional code but lacking in complexity or organization.	Incomplete or poorly organized code with limited functionality.

Project Ideas Based on Previous Projects

Data Science Story (Project 1): Expand the data analysis project by incorporating machine learning models to predict outcomes based on the data set analyzed. For example, use regression analysis or classification to enhance the storytelling aspect of the data.

2D Platformer Game Development (Project 2): Enhance the 2D platformer game by adding more complex game mechanics such as power-ups, additional levels, or multiplayer features. Consider incorporating AI elements for enemy behavior to increase the game's challenge.

Advanced Data Structure Implementations (Project 3): Build upon the data structures video project by implementing and visualizing complex data structures such as red-black trees, AVL trees, or B-trees, and demonstrate their efficiency in handling large datasets.

LLM Applications (Project 4): Use an LLM API for a creative purpose different than what you did in project 4. Examples include real-time language translation, sentiment analysis, or synthetic data generation.

Graph Theory Applications (Project 5): Take the graph theory applications project further by creating an interactive application that models real-world transportation networks or social networks, allowing users to visualize changes in real-time as they adjust parameters or simulate different scenarios.