CC 204 – DATA STRUCTURES AND ALGORITHMS
FINAL PROJECT RUBRICS

Project Requirements Overview

The program should:

- Use at least two learned data structure.
- Include sorting functionality that does not rely on a tree-based structure.
- Show data before and after sorting.
- Be interactive with validated user input.
- Contain computational logic that models real-world processes.
- Implement CRUD functionalities.
- Have comprehensive error handling.
- Include a documentation on what data structures are used, why, and what data it stores.
- Conduct a project presentation and Q&A session.
- Submit peer and self-evaluation.

Criteria	Description	Total Points	Scoring Guide
1. Program Logic & Real-World Relevance	The program models or imitates a realistic process that reflects real-world logic or operations.	15	15 pts – Logic is realistic, innovative, and closely mirrors real-world systems 12 pts – Logic is realistic and mostly consistent 9 pts – Logic shows some real-world application but has flaws 6 pts – Logic is weak or unrealistic 3 pts – Program logic broken or disconnected from real-world use
2. Data Structures Used	At least two learned data structures (Linked List, Stack, Queue, Tree Set, Tree Map, Heaps, Priority Queues) must be implemented effectively.	15	15 pts – Two or more appropriate data structures used efficiently and correctly 12 pts – Two used appropriately with minor logic or integration flaws 9 pts – Two used but poorly integrated or not justified 6 pts – Only one data structure used correctly 3 pts – Data structures misused or missing

3. Sorting Functionality	Sorting may use a built-in or manually implemented algorithm but must clearly show data before and after sorting. Sorting must not rely on a tree-based structure.	10	10 pts – Displays unsorted and sorted data correctly using a valid method; sorting logic fully integrated 8 pts – Displays both states with minor display or logic issues 6 pts – Sorting works but output or logic only partially shown 4 pts – Sorting present but unsorted/sorted outputs unclear 2 pts – Sorting poorly implemented or missing visible comparison
4. CRUD Operations	Demonstrates Create, Read, Update, and Delete functionality.	10	10 pts – All CRUD operations work flawlessly 8 pts – CRUD mostly works; minor function errors 6 pts – 3 out of 4 functionalities correct 4 pts – 2 functions implemented correctly 2 pts – CRUD mostly nonfunctional
5. User Interactivity & Input Handling	Program must include a clear, user-friendly interface (console or GUI), handle user inputs with validation, and provide a brief user guide explaining how to run the program and access all functions.	10	10 pts – Fully interactive; input handled smoothly with robust validation; well-written user guide included8 pts – Interactive with minor validation issues; user guide mostly clear6 pts – Some validation present; limited interactivity; lacks clarity in guide4 pts – Poor handling of input; minimal or incomplete user guide2 pts – No user interaction, validation, or guide provided
6. Error Handling & Exceptions	Safeguards against runtime errors.	10	10 pts – Excellent handling; all major exceptions addressed 8 pts – Most exceptions covered; a few missed 6 pts – Basic try-catch present but poorly applied 4 pts – Minimal or incomplete exception handling 2 pts – No usable error handling
7. Code Quality & Structure	Organization, readability, naming, and modular structure.	10	10 pts – Clean, modular, and consistent with Java standards 8 pts – Mostly clear; minor style issues 6 pts – Understandable but repetitive or cluttered 4 pts – Poor structure; readability issues 2 pts – Very messy or unreadable
8. Documentation & Explanation of Data Structures	Clear explanation of chosen data structures, purpose, and stored data.	8	8 pts – Comprehensive documentation detailing structure choice and rationale 6 pts – Mostly complete but missing some explanations 4 pts – Basic documentation without rationale 2 pts – Minimal comments or unclear reasoning 0 pt – No documentation

9. Presentation	Oral presentation and demonstration of the program.	6	6 pts – Professional, clear, and engaging demonstration 5 pts – Clear and understandable with small errors 4 pts – Adequate but lacks depth or polish 3 pts – Somewhat unclear explanation 2 pts – Disorganized or incomplete presentation
10. Q&A Performance	Ability to defend and explain code decisions.	6	6 pts – Answers all questions clearly and accurately 5 pts – Mostly correct answers; slight uncertainty 4 pts – Understands basics but lacks depth 3 pts – Vague or hesitant responses 2 pts – Unable to explain or defend code
6. Peer and Self Evaluation		**30**	
	TOTAL	**130**	

Summary:

Program Logic & Relevance	15
Data Structures Used	15
Sorting Functionality	10
CRUD Operations	10
User Interactivity, Input Handling, and Guide	10
Error Handling	10
Code Quality	10
Documentation	8
Presentation	6
Q&A	6
Peer and Self Evaluation	30
Total	130