National University of Sciences and Technology
School of Electrical Engineering and Computer Science
Department of Computing

CS-250 Data Structures and Algorithms

Spring 2024

Assignment 3

Sorting Algorithms and Heap Data Structure

Announcement Date: 26th April 2024

Due Date: 5th May 2024 at 11:59 pm (on LMS)

Instructor: Bostan Khan

Objective: The objective of this assignment is to familiarize students with various sorting algorithms, their implementations, complexities, and applications. The students will also implement priority queues using the heap data structure.

Please read the instructions for each task with great care and be sure to implement the requirements stated against each task.

This is a group assignment where each group can have a maximum of **2 students**. Only one student should submit the assignment from each group. **The student should submit a report in PDF format along with the code source files.** Remember to include **the names and registration numbers** of both students at the start of the report. **No late submissions will be accepted so make sure to submit your assignment on LMS within the given time.**

The report PDF and the sources files should be compressed in a single zip file and submitted on LMS with the name in following format: <Student_name>_<Reg_no>.zip

- **Tasks 1 - Research: (20)**
  Research and provide brief explanations for the following sorting algorithms:
  - Bubble Sort
  - Insertion Sort
  - Merge Sort
  - Heap Sort

- **Task 2 - Implementation: (30)**
  - Implement **merge sort** and **heap sort** algorithms in C++. Provide well-commented code along with explanations of your implementation choices.
  - You should show application of your merge sort and heap sort implementations on 3 examples each.

- **Task 3 – Analysis of sorting algorithms: (20)**
  - Compare the time complexity of the two sorting algorithms you implemented.
  - Discuss the best-case, average-case, and worst-case scenarios for merge sort and heap sort algorithms.
  - Analyze and explain the advantages and disadvantages of each algorithm.

- **Task 4 – Heap Data Structure: (30)**
  Implement a **max-priority queue** using the max-heap data structure in C++.
  Include functions/methods for the following operations:
  - Inserting an element with a priority.
  - Extracting the maximum priority element.
  - Displaying the elements in the priority queue.
  - Increasing the priority of an element in the queue.
  - Discuss some applications of the implemented priority queue using the heap data structure.

- **Documentation Report PDF:** Provide a report PDF for your assignment, including explanations of the algorithms and data structures used, as well as any decisions made for each task. **The marks of your coding tasks will be directly influenced by the documentation report.**

    - **Testing Results in Documentation:** Test your implementations thoroughly to ensure correctness and robustness. Provide sample input-output scenarios for each scenario to demonstrate the functionality of your implementations.

    Remember to document your work well in the documentation report. Explain the code, its design and test results with details for obtaining max marks for the whole assignment.

- **Submission:** You will have to submit a separate .cpp source file for each task. Present The documentation report including the test results in a well-prepared PDF.
- **Additional Notes:**
    - Collaboration with classmates is allowed for discussing concepts and problem-solving strategies, but each group must submit their own solution.
    - **Plagiarism or copying of code from other students will result in zero marks.**