NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES

(KARACHI CAMPUS)

FAST School of Computing Spring 2024

PROJECT REPORT

MEMBERS:

- Ayesha Ansari (22K-4453) [Group Leader]
- Kausar Saeed (22K-4588)

COURSE INSTRUCTOR:

 MISS ANUM QURESHI

Project Topic:

Task Manager

Introduction:

The project aims to develop a Task Manager program that efficiently manages tasks with various attributes, such as title, priority, description, due date, importance, and status. The program uses multiple data structures to enhance task management and provides features like insertion, display, editing, searching, marking as completed, statistics, and undo/redo functionality.

Motivation:

With our project, we can dive deep into analytics, uncovering hidden patterns that lead to impactful solutions. Together, we'll revolutionize decision-making by harnessing the power of data.

Features of the project:

Mentioned below are the precise features of our Task Manager project:

1. Task Insertion: Add new tasks with details like title, priority, description, due date, and importance.

2. Task Display: View all tasks with their details: title, priority, description, due date, importance, and status.

3. Task Editing: Modify existing tasks, including priority, description, due date, importance, and status.

4. Task Searching: Find specific tasks quickly by their title using the search functionality.

5. Mark Task as Completed: Flag tasks as completed to track progress and manage the task list efficiently.

6. Task Statistics: Get insights into task management with statistics like completed and remaining tasks.
7. Display by Priority: View tasks sorted by priority to prioritize work effectively.

8. Display Highest Priority Task: Identify and view the task with the highest priority for focused attention.

9. Task Deletion: Remove tasks from the list when they're no longer relevant or needed.

10. Undo and Redo: Reverse or redo actions such as task insertion, deletion, or editing for flexibility.

11. Display Due Tasks: See tasks that are overdue or due on the current date for timely completion.

Challenge faced during this project:

Developing the Task Manager project posed several challenges. Managing user input validation and error handling was crucial for ensuring application stability. Implementing complex data structures like heaps and AVL trees required a deep understanding of algorithms. Balancing algorithm efficiency with the complexity of task management operations was challenging. Thorough testing and debugging were necessary to identify and resolve issues, ensuring smooth application functionality.

Data Structures used:

The project utilizes four main data structures:

1. Heap (Priority Queue): The project employs a max heap to maintain tasks based on their priority. This ensures quick access to the task with the highest priority (top of the heap) in constant time (O(1)).
2. AVL Tree: An AVL tree is implemented to efficiently search and manage tasks based on their priority. This self-balancing binary search tree provides logarithmic time complexity for search operations (O(log n)).
3. Stack: Two stacks are used for implementing undo and redo functionality. These stacks store task objects and allow users to revert or redo their recent actions efficiently.

4. Queue: A simple queue is employed for level-order traversal during the display of tasks. This ensures that tasks are presented in a structured manner, prioritizing those with higher levels of importance and urgency.

Real Life Applications:

A task manager project using data structures can be applied in various real-life scenarios such as,

1. Project management
2. Academic task scheduling
3. Healthcare appointment scheduling
4. Supply chain management
5. Personal time management
6. Restaurant order management.

Data structures like heap, queues, trees and stack can be utilized to efficiently organize, prioritize, and manage tasks or orders in these domains.

Conclusion:

In conclusion, the Task Manager project effectively utilizes data structures like the heap, AVL tree, stack, and queue to enhance task management. The careful selection and implementation of these structures contribute to the project's efficiency in terms of time and space complexity. The project stands out by providing quick access to high-priority tasks and efficient undo/redo functionality.