



COMSATS University Islamabad, Attock Campus

Department of computer Science

Program: BSE

Course: DS Theory

Name: Huzaifa Murtaza

Reg no: SP23- BSE-007

Assignment no: 01

Date: 24 September 2024

Submitted to: Mr. Kamran

Introduction

The objective of this assignment is to implement a task management system using a singly linked list in C++.

Each task contains a unique ID, description, and priority level. The following functionalities are implemented:

1. Add a new task based on its priority (higher priority tasks come first).
2. View all tasks.
3. Remove the task with the highest priority.
4. Remove a specific task by its task ID

Code Explanation:

1. `addTask()`: This function inserts a new task into the list, maintaining the order of tasks based on priority.
2. `viewTasks()`: Displays all tasks currently in the list, showing the task ID, description, and priority.
3. `removeHighestPriorityTask()`: Removes the task with the highest priority (i.e., the first task in the list).
4. `removeTaskByID()`: Deletes a task based on its unique task ID, updating the linked list accordingly

Output Screenshots:

```
--- Task Management Menu ---
1. Add a new task
2. View all tasks
3. Remove the highest priority task
4. Remove a task by ID
5. Exit
Enter your choice: 1
Enter task ID: 01
Enter task description: lab assignmnet completion
Enter task priority: 7
Task added successfully!

--- Task Management Menu ---
1. Add a new task
2. View all tasks
3. Remove the highest priority task
4. Remove a task by ID
5. Exit
Enter your choice: 2
Task ID: 1 | Description: lab assignmnet completion | Priority: 7

--- Task Management Menu ---
1. Add a new task
2. View all tasks
3. Remove the highest priority task
4. Remove a task by ID
5. Exit
Enter your choice: 1
Enter task ID: 02
Enter task description: book reading
Enter task priority: 4
Task added successfully!
```

--- Task Management Menu ---

1. Add a new task
2. View all tasks
3. Remove the highest priority task
4. Remove a task by ID
5. Exit

Enter your choice: 1

Enter task ID: 03

Enter task description: playing games

Enter task priority: 3

Task added successfully!

--- Task Management Menu ---

1. Add a new task
2. View all tasks
3. Remove the highest priority task
4. Remove a task by ID
5. Exit

Enter your choice: 2

Task ID: 1 | Description: lab assignmnet completion | Priority: 7

Task ID: 2 | Description: book reading | Priority: 4

Task ID: 3 | Description: playing games | Priority: 3

--- Task Management Menu ---

1. Add a new task
2. View all tasks
3. Remove the highest priority task
4. Remove a task by ID
5. Exit

Enter your choice: 1

Enter task ID: 04

Enter task description: preapring quiz

Enter task priority: 9

Task added successfully!

--- Task Management Menu ---

Task ID: 2 | Description: book reading | Priority: 4

Task ID: 3 | Description: playing games | Priority: 3

```

--- Task Management Menu ---
5. Exit
Enter your choice: 2
Task ID: 4 | Description: preapring quiz | Priority: 9
Task ID: 1 | Description: lab assignmnet completion | Priority: 7
Task ID: 2 | Description: book reading | Priority: 4
Task ID: 3 | Description: playing games | Priority: 3

--- Task Management Menu ---
Task ID: 1 | Description: lab assignmnet completion | Priority: 7
Task ID: 2 | Description: book reading | Priority: 4
Task ID: 3 | Description: playing games | Priority: 3

--- Task Management Menu ---
Task ID: 3 | Description: playing games | Priority: 3

--- Task Management Menu ---
--- Task Management Menu ---
1. Add a new task
1. Add a new task
2. View all tasks
3. Remove the highest priority task
4. Remove a task by ID
5. Exit
Enter your choice: █

```

Conclusion

Through this assignment, I have learned how to implement and manage a singly linked list in C++ for practical applications such as task management.

The most challenging part was maintaining the list in priority order during task insertion.

This exercise provided valuable experience in linked list manipulation and memory management in C++