Task Management System Using Singly Linked List

Introduction

The objective of this assignment is to design and implement a task management system using a singly linked list. The system allows users to add new tasks, view all tasks, remove the highest priority task, and remove a task by ID. This report explains the logic behind each function implemented in the code.

Code Explanation

1. Node Structure

The node structure, TaskNode, consists of three members: taskId, taskDescription, and priority. Each node represents a task.

2. Add Task Function

The addTask() function adds a new task to the list at the correct position based on priority.

void addTask() {

3. View Tasks Function

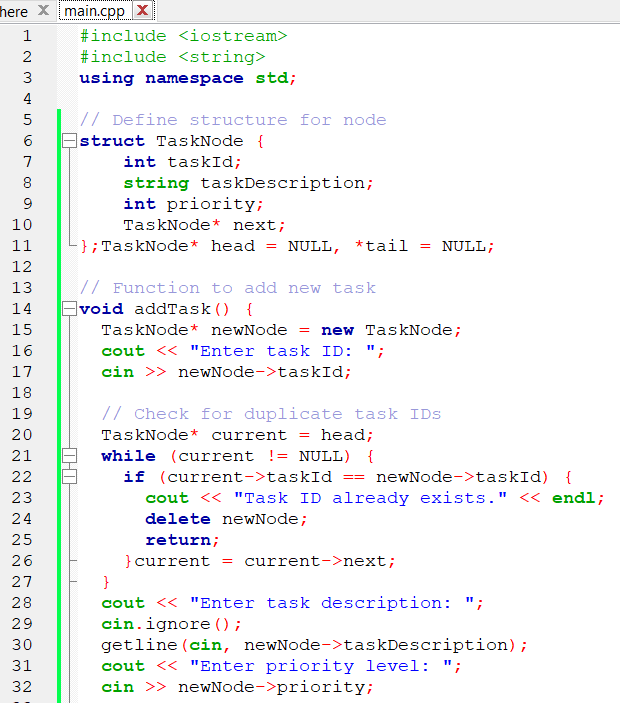
The viewTasks() function displays all tasks in the list.

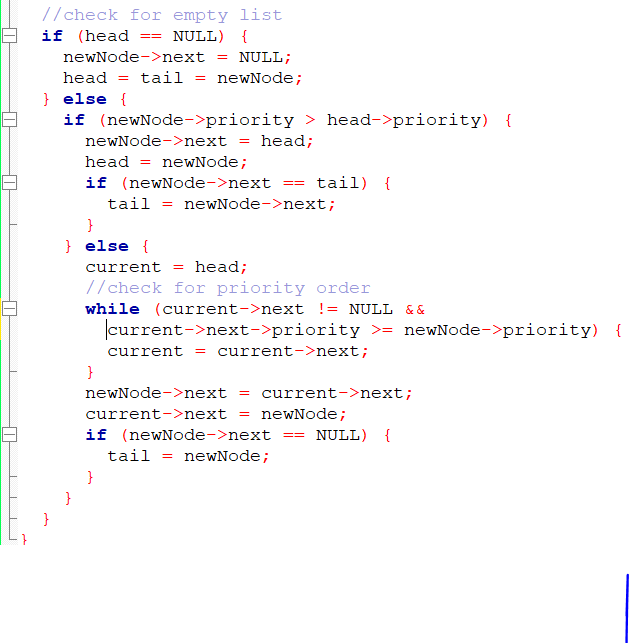
4. Remove High Priority Task Function

5. Remove Task by ID Function

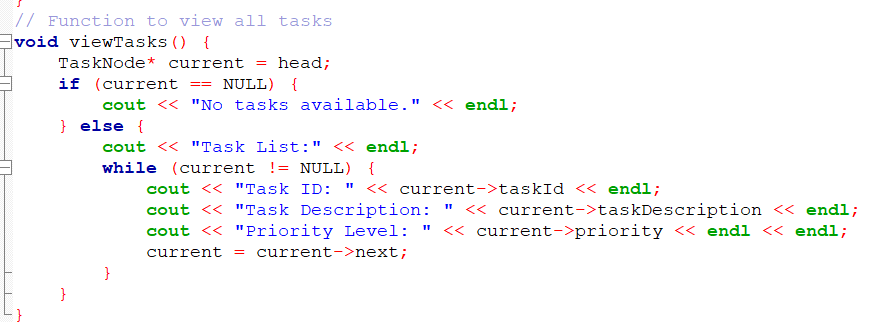
The removeTaskById() function removes a specific task using its task ID.

1. Adding a New Task

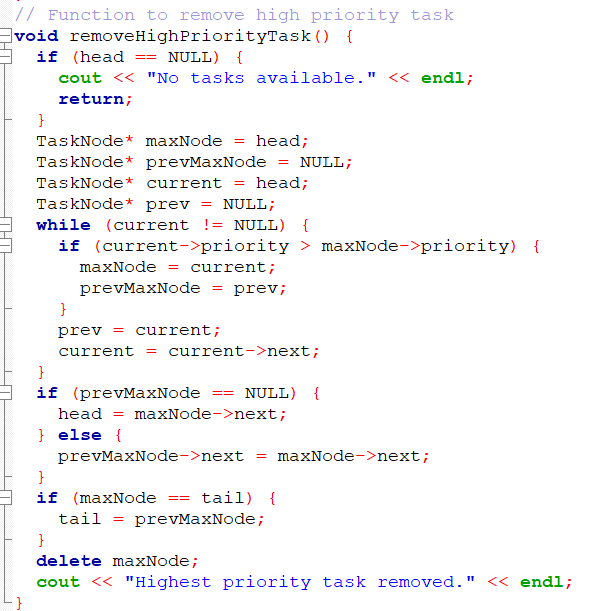




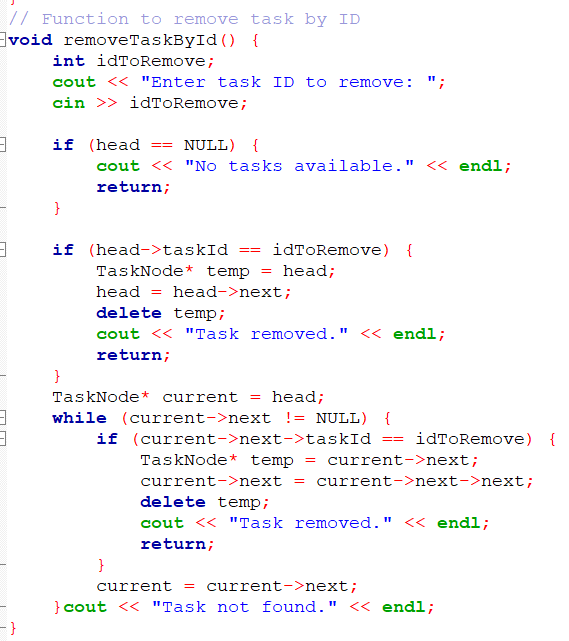
2. Viewing All Tasks



3. Removing Highest Priority Task

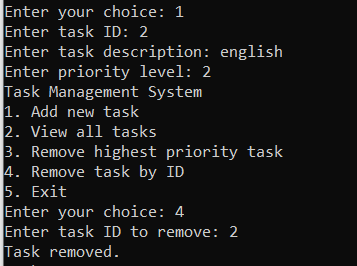
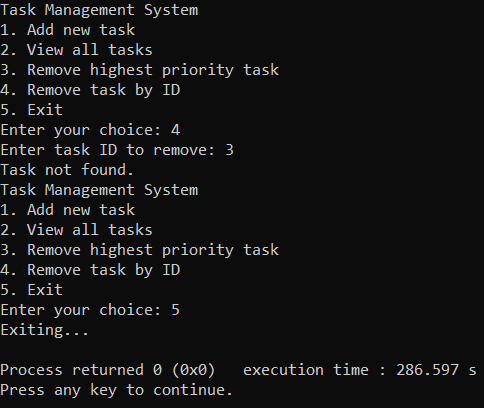
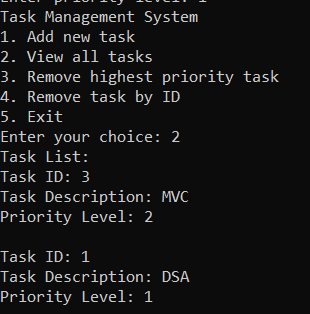
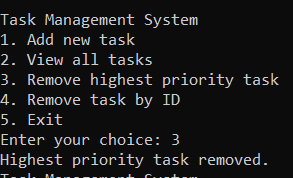
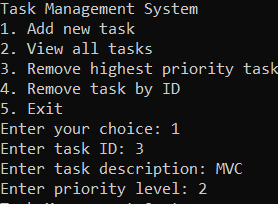
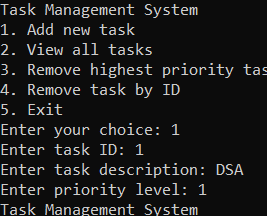


4. Removing Task by ID





Output:



Conclusion

Through this assignment, I gained hands-on experience implementing a singly linked list to manage tasks. The challenges faced included handling duplicate task IDs and updating the tail pointer correctly. This assignment reinforced my understanding of data structures and algorithm design.