

The objective is to create a comprehensive system that efficiently manages tasks assigned to team members, prioritizes tasks based on their urgency, and offers a wide range of operations for manipulating and retrieving task information. The assignment aims to provide a practical solution for project teams to streamline task management, enhance productivity, and maintain organized records of tasks.

## **Requirements:**

### **1. Task Management Features:**

Your Task Management System should include the following key features:

**Add Assignee:** Users should be able to add new assignees to the system, providing information such as First Name, Last Name, address, and DOB.

**Search Assignee by ID:** Search Function to find an assignee by their ID.

**Search Assignee by name:** Search Function to find an assignee by their first and last name, if there is more than 1 person with the same name; display all the Assignees with that name.

**Search Assignee with no task Assigned**

**Shift Tasks of one assignee to another Assignee**

**Delete Assignee:** If an assignee has not yet completed their tasks, they cannot be deleted.

**Add Task:** Users should be able to add new tasks to the system, providing information such as task description, priority level, and the assignee ID (team member).

**Complete Task:** Tasks should be marked as completed when they are finished. Users should be able to update the task status accordingly.

**Print Task Queue:** The system should provide an option to display the list of tasks, sorted in ascending order of priority.

**Find Highest Priority Task:** Users should be able to find and display the task with the highest priority (lowest priority value) without removing it from the queue.

**Find Task by Assignee:** The system should allow users to search for tasks assigned to a specific team member and display them.

**Count Total Tasks:** Users should be able to obtain a count of the total number of tasks in the system.

**Update Task Priority:** The system should allow users to update the priority level of a specific task without removing it from the queue.

**Clear Completed Tasks:** Users should be able to remove all completed tasks from the system, keeping only pending tasks.

**Display All Completed Tasks:** The system should provide a feature to display all completed tasks, which can be helpful for generating progress reports.

**Search for Task by Priority Range:** Users should be able to search for tasks within a specified priority range and display them.

## **2. RED-BLACK Tree Implementation:**

The red-black tree should store task records. Each node in the red-black tree represents a task in the system. The node structure of the task tree includes the following fields:

**Task ID:** A unique identifier for each task.

**Description:** A text description of the task.

**Priority Level:** An integer value representing the task's urgency, with lower values indicating higher priority.

**Assignee ID:** Every Assignee should have a unique ID.

Another tree would be created to store the details of Assignee, which would include:

**First Name**

**Last Name**

**Address**

**Date of birth:** in the format of day/month/year

**Assignee ID:** This will be auto-generated.

Include an In-Order traversal Function of Red-Black trees. Which should also display the color of each node. The tree would be created according to the priority of the tasks in the case of the task tree and the Assignee ID in the case of the Assignee tree.