2nd Assessment Report Template

JINGFENG ZHANG 50091040

1. Introduction

In our daily lives, there are many tasks to accomplish, and these tasks need to be completed in order based on their deadlines and priorities. Task scheduling programs help us arrange the completion sequence of tasks more efficiently and time-savingly.

Based on the priority and deadline input by the user, tasks are arranged in order of priority first, followed by the deadline for completion. The task scheduler comprehensively organizes the sequence of task completion. Additionally, users can adjust a task as needed and refresh the task sequence. Once a task is completed, users can delete it from the task sequence. The programming was achieved by the Priority Queue data structure e and Object-Oriented Programming principles. But this program still has some limitations. For instance, tasks with lower priority but earlier deadlines might be scheduled later, resulting in insufficient time for completion.

1. Task Class Design

1st paragraph: Explain the attributes of the Task class (6 points).

This class has three attributes: \_\_description,\_\_priority,\_\_deadline. They are used to describe the task's description, priority, and deadline.

2nd paragraph: Describe the methods of the Task class, their purpose, and implementation (10 points).

In the class, it has many methods. The \_\_init\_\_method receives some attributes and stores them. The get\_description method was created for the users to get the task’s description. The method was implemented by return its \_\_description attribute. The get\_priority and the get\_deadline are as same as the get\_decription. The set\_description method was created for the users to reset the task’s description. The method was implemented by seting its \_\_description attribute with its parameter. The set\_priority and the set\_deadline are as same as the set\_decription. The \_\_str\_\_ method was overrided to print the task’s information obviously.

3rd paragraph: Discuss encapsulation and how it is achieved in the Task class (4 points).

1. PriorityQueue Class Design

1st paragraph: Explain the choice of the priority queue data structure used in the PriorityQueue class (6 points).

2nd paragraph: Describe the attributes of the PriorityQueue class (4 points).

3rd paragraph: Explain the methods implemented for adding, removing, and peeking at tasks, along with their purpose and implementation (10 points).

4th paragraph: Discuss how sorting tasks based on deadlines as a secondary criterion is achieved (5 points).

1. Scheduler Class Design

1st paragraph: Explain how the Scheduler class utilises the PriorityQueue class (6 points).

2nd paragraph: Describe the attributes of the Scheduler class (4 points).

3rd paragraph: Explain the methods implemented for adding, removing, reordering, and executing tasks, along with their purpose and implementation (10 points).

4th paragraph: Discuss any design choices made in the Scheduler class and their reasoning (5 points).

1. Testing and Demonstration

1st paragraph: Describe the test program and main function used to demonstrate the Task Scheduler's functionality (8 points).

2nd paragraph: Provide sample input/output to show the Task Scheduler's performance (8 points).

3rd paragraph: Discuss any challenges faced during testing and how they were resolved (4 points).

1. Conclusion

1st paragraph: Summarise the project and its main features (2 points).

2nd paragraph: Reflect on the learning experience and the application of OOP concepts (2 points).

3rd paragraph: Suggest possible improvements or future enhancements (1 point).