Module 10 Homework - Priority Queue Implementation for Task Management

Overview

We will implement a priority queue to manage the scheduling of tasks in an operating system. Tasks with higher priority should be executed first. You are provided with a skeleton code for three classes `Entry`, `MaxHeap`, and `TaskManager`. You will complete the implementation of the `Entry` class and `MaxHeap` class.

Requirements:

1. `Entry` Class Implementation:
    - Implement the comparison methods in the provided `Entry` class to compare entries based on priority.
2. `MaxHeap` Class Implementation:
    - Implement methods to insert entries, remove the maximum priority entry, maintain the heap property, and change the priority of existing tasks.
    - Implement heapifying methods to ensure the heap property is maintained after insertion, removal, and priority change operations.
3. `TaskManager` Class Implementation:
    - The `TaskManager` class includes methods to add tasks, remove tasks with the highest priority. Use the `MaxHeap` class methods to manage the task queue efficiently.

Examples

Several examples of behavior shown below.

```
# Creating entries
entry1 = Entry(priority=5, process_id="Process 1")
entry2 = Entry(priority=3, process_id="Process 2")
entry3 = Entry(priority=7, process_id="Process 3")
entry4 = Entry(priority=8, process_id="Process 4")
entry5 = Entry(priority=2, process_id="Process 5")
entry6 = Entry(priority=6, process_id="Process 6")
entry7 = Entry(priority=4, process_id="Process 7")
# Initializing MaxHeap
max_heap = MaxHeap()

# Adding entries to the max heap
max_heap.put(entry1)
max_heap.put(entry2)
max_heap.put(entry3)
max_heap.put(entry4)
```

```
    max_heap.put(entry5)
    max_heap.put(entry6)
    max_heap.put(entry7)
    # Changing priority of a process
    max_heap.change_priority(process_id="Process 1", new_priority=10)

    # Changing priority of a process
    max_heap.change_priority(process_id="Process 3", new_priority=1)

    # Removing processes until heap is empty
    while max_heap:
        print(f"Removing max priority process: {max_heap.remove_max()}")
    # Expected output:
    # Removing max priority process: Process 1
    # Removing max priority process: Process 4
    # Removing max priority process: Process 7
    # Removing max priority process: Process 6
    # Removing max priority process: Process 2
    # Removing max priority process: Process 5
    # Removing max priority process: Process 3
```

Imports

No imports allowed on this assignment, with the following exceptions:

- Any modules you have written yourself.

Grading

This assignment is 100% auto graded.

Submission

Submit the following files:

- `TaskManager.py`

Students must submit individually to Gradescope by the posted due date (Tuesday, April 16th at 11:59pm) to receive credit.