Computing Assignment Grading Criteria

Demonstration of functionality (30 points)

Students should provide a clear and well-structured demonstration program, i.e., how to use your codes. Your program should take as input the data files you generated, e.g., registration records, and output weekly and monthly reports. Your program should also demonstrate how to update attributes and withdraw registrations. It is preferrable if students include a documentation to describe their code's use and functionality.

10 points: The program and codes are well-commented or well-documented. The registration records you generated must include all profession categories, age groups and all risk status, i.e., make them look realistic.

10 points: The weekly and monthly reports are correct and well-formatted. Your program should be able to output the weekly reports in order of name, profession category or age (you don't have to output multiple files for each possible order, but you have to show clearly how to do so, either by commenting or documenting your code).

10 points: The program clearly demonstrate how to update attributes and withdraw registrations.

Data Structures for Local Queues (10 points)

10 points: Correct implementation for the local queues.

Fibonacci Heap for the Central Queue (50 points)

5 points: The codes are well-commented (readable).

10 points: Correct implementation of Fibonacci Heap.

10 points: Correct implementation of priority rules.

5 points: The order from local queues is preserved. (Different local registries can be added to the central queue in random order)

10 points: The registration's attributes can be updated, and the priority is correctly reflected.

10 points: The registration in the central queue can be withdrawn. If those who have withdrawn register again, add additional two-week meeting time, unless they are in the medium or high-risk group.

lists for the Appointments (10 points)

10 points: Correct Implementation.