Project 3
CSC 311: Data Structures, Fall 2017

Department of Computer Science

California State University, Dominguez Hills

(Due: November 27, 2017, 11:59 PM PDT)

A. Objectives

1. Implement a priority queue using heaps (20 points)
2. Use methods to do separate and repetitive work. The main method should not have more than 20 lines of code (10 points)
3. Implement the project correctly. (60 points)

10 points will be awarded for use of meaningful identifiers, consistent indentation, explanatory comments in your code and properly formatted output.

B. Description

We would like to implement boarding of passengers in an airplane. Passengers arrive belonging to different classes in the cabins, can be elderly, can have infants with them, or belong to the armed services. Each of the passenger is provided a priority depending on the conditions described. A passenger with a higher priority (indicated by a small number) is boarded before passengers with lower priority, ignoring the order in which they arrive; hence using a normal queue would not be sufficient. Using a priority queue to keep track of the order of boarding is useful in this scenario.

Construct the program using the following guidelines:

1. Create a Java Class for passengers. The data fields in the Class should include First Name, Last Name, Date of Birth, Boarding category and Priority.
2. Simulate incoming passengers with varied degree of boarding class and priority. Higher boarding class should be reflected by lower priority. Range priorities from 1 to 20. For example, "Traveling with infant" is priority 1, "Elderly" is priority 5, "Armed Services" is priority 10, "First Class" is priority 15 and "Economy class" is priority 20.
3. Add the passengers to the priority queue when they arrive based on the priority.
4. If there are more than one passenger with the same priority, service is first come first serve.
5. Implement the program to handle 15 passengers over a simulated time period of 10 minutes.
6. Assume it takes 20 seconds to board any passenger. Once a passenger is been attended, other passengers have to wait even the ones that arrive with higher priority.
7. Print out the details of the passengers in the order in which they arrived as well as the order in which they were boarded.

C. Constraints

1. Implement all code in Java programming language
2. The project is due by November 27, 2017, 11:59 PM PDT, using Blackboard.
3. This is not a group project. Copying code from others or using an unfair means is strictly not allowed and plagiarism charges will be imposed on students who do not follow this.
4. If students find code somewhere else that they want to include in the program, they need to cite the reference as well.
5. Include print screens of the execution of the program as a separate pdf. All the options must be included to show the complete execution of the program.
6. Upload all files using a single zip file; dont use other compressed format such as rar.