CSE218 Queues and Priority Queues HW3

1. Implement a queue to store strings using an Array List.

2. Implement a priority queue to store numbers of Double type using an Array List.

3. Create a **GUI** program using **JavaFX** to simulate student enrollment into a class. For a particular class, the class size limit is 5. This means the first 5 students registered will automatically be accepted into the class. The remaining applicants, if any, will be wait-listed.
   As an example, the user enters the information of 7 applicants. With a click of a button, the program will display two text areas. One text area contains the name, phone number, and registration date and time of the first five students that are automatically accepted into the class. The other text area contains the same information of the remaining two students in the wait list. Use a Java queue API named **ArrayBlockingQueue** for this problem.

4. Use the priority queue API **PriorityQueue** to implement another wait list like the one in the previous problem. This time, however, students are admitted based on their GPAs rather than the chronological order their data were entered. The higher the GPA, the more likely the student gets admitted. Enter information (name, phone, GPA) of 7 students and show the five students automatically accepted and the remaining two students in the wait list, all based on their GPAs.

5. Use JavaFX to implement a user sign-in and sign-up program. It must have a sign-in screen and a sign-up screen.

   The sign-in screen allows the user to enter a username and password for an existing account. If the username and password are correct, the program displays a message to indicate success. Otherwise, the program displays a message to indicate failure.

   The sign-up screen allows the user to create a new user account. The user may do so by providing a username, a password, and an email address. The username must be unique.

   All the user accounts must be stored in a data structure of your choice (an array or array list) to allow both sign-in and sign-up.

   The sign-in screen may have a text field and a password field for the user to enter his/her username and password, respectively. In addition, it may also have a sign-in button and a sign-up button. The sign-in button, when clicked, check the validity of an existing account. The sign-up button, when clicked, opens the sign-up screen to allow the user to create and store a new user account. On the sign-up screen, you may also have a **Back** button so the user may click it to get back to the sign-in screen to sign in.