Rubric for Project #3 (100 points total)

Zip file name and project name are identical, and follow the format below "lastname_project3" (5 points).

Project compiles and runs as submitted (5 points)

- Project contains all the required files (0-3 points)
    - event class, two queue classes (waiting queue and event queue), a class containing the driver program.
- Other files including report file (report.txt), and sample run (0-2 points)

Project documentation (8 points)

- Name and project description (0-3 points)
- Class and Method descriptions (0-3 points)
- in-line documentation (0-2 points)

Program style (3 points)

- Meaningful names for variables, methods, and classes (0-1 point)
- Proper indentation (0-1 point)
- Code is readable- includes blank line(s) throughout the code (0-1 point)

Implementation requirements (7 points)

- Two generic queue classes (3 points)
- Asks the user for simulation parameters (2 points)
- Accommodates for infinite waiting queue size (2 points)

Screen output is complete and correct (10 points)

- % time modems were busy (3 points)
- Average wait time in the waiting queue (3 points)
- Number of users left in the waiting queue (2 pts)
- Displays a message when the waiting queue is full (2 points)

Simulation result is written neatly to the report file (10 points)

- A header line that labels each column (2 points)
- A single line of output per simulation containing 8 columns (3 Points)
- Each line shows input parameters as well as the % of time modems were busy, average wait time, and number of customers left in waiting queue when the simulation ends (5 points)

Event class (10 points)

- Event has an ID, type, and time of occurrence (3 points)
- Event class implements Comparable interface and overrides compareTo (5 pts)
- Event class has other getter and setter methods (2 points)

Event queue class is generic (10 points)

- Event queue is implemented using an ordered singly linked list (4 points)

- Event queue is a priority queue and not a FIFO (6 points)
    - Queue method offer adds based on the time of event (3 points)
    - Methods remove, poll, peek, and element are supported (3 points)

Waiting queue class is generic (10 points)

- Waiting queue is implemented using a circular array (4 points)
- Accommodates both finite and infinite queue (2 points)
- Queue methods (offer, remove, poll, etc.) are implemented correctly (4 pts)

Program is robust and doesn't crash for bad input (3 points)

Program works correctly (19 points)

- Event queue is initialized with dial-in events prior to simulation (5 points)
- Number of dial-in events and connection times are based on Poisson (2 points)
- Simulation is implemented correctly (12 points)
    - All the events scheduled for a given time unit are handled properly (2 pts)
    - Modems are freed and allocated correctly (2 points)
    - Waiting time and modem busy time are computed correctly (4 points)
    - Results are displayed and written to the report file at the right time (4 pts)