**CS 248 – Object-Oriented Programming and Data Structures**

**HW6: 100 points**

**Objective:**

The objective of this homework is to learn about:

1. Java class and Interface
2. Implement Queue for a real life problem
3. Learn how to bribe in restaurants (JUST KIDDING!!!)

**Problem Statement:**

The fancy new French restaurant *La Food* is very popular for its authentic cuisine and high prices. This restaurant does not take reservations. To help improve the efficiency of their operations, the Maitre De has hired you to write a program that simulates people waiting for tables. The goal is to determine the average amount of time people spend waiting for tables.

Your program will read in a list of event descriptions from a text file, one description per line.

* ***Arrival****:* A party has arrived to eat. Add them to the end of the list of waiting parties (a Queue) and tell them to wait at the bar (where strong drinks are served) until called. This event is described in the following format:

A  *t n b name*

Here A denotes arrival, *t* is the time of arrival (in minutes past opening time), *n* is the number of people in the party, *b* is the amount of the bribe, in Euros of course, and *name* is the name to call when the table is ready.

A party is then placed in the queue in the right place (ahead of everyone who gave a lesser bribe). Two parties that bribed the same amount are placed in order of arrival.

* ***Table****:* A table has become available; remove the party from the queue and seat them. This event is described in the following form:

T *t*

Here *t* is the time the table became available (again, in minutes past opening time),

* ***Quit****:* This is a sentinel event indicating the end of the input file. It has the following form:

Q

When the events in the file have been processed, compute and print the average waiting time per party. If there are still people waiting for tables, print a summary of who is still waiting.

**Notes:**

* **Submit only the Restaurant.java file**
* **Read from the file in.txt**

**Sample input output:**

**Input:**

A 3 3 0 Merlin

A 8 2 2 Arthur Pendragon

T 10

A 12 2 2 Sir Lancelot

T 15

A 17 3 5 The Green Knight

T 20

Q

**output**

\*\*\* Welcome to the La Food Restaurant Simulator \*\*\*

Please wait at the bar,

party Merlin of 3 people. (time=3)

Please wait at the bar,

party Arthur Pendragon of 2 people. (time=8)

Table for Arthur Pendragon! (time=10)

Please wait at the bar,

party Sir Lancelot of 2 people. (time=12)

Table for Sir Lancelot! (time=15)

Please wait at the bar,

party The Green Knight of 3 people. (time=17)

Table for The Green Knight! (time=20)

\*\* Simulation Terminated \*\*

The average waiting time per party: 2.6666666666666665

The average waiting time per person: 2.7142857142857144

The following parties were never seated:

party Merlin of 3 people.

**Input:**

A 3 3 0 Merlin

A 8 2 0 Arthur Pendragon

T 10

A 12 2 0 Sir Lancelot

T 15

A 17 3 0 The Green Knight

T 20

Q

**output**

\*\*\* Welcome to the La Food Restaurant Simulator \*\*\*

Please wait at the bar,

party Merlin of 3 people. (time=3)

Please wait at the bar,

party Arthur Pendragon of 2 people. (time=8)

Table for Merlin! (time=10)

Please wait at the bar,

party Sir Lancelot of 2 people. (time=12)

Table for Arthur Pendragon! (time=15)

Please wait at the bar,

party The Green Knight of 3 people. (time=17)

Table for Sir Lancelot! (time=20)

\*\* Simulation Terminated \*\*

The average waiting time per party: 7.333333333333333

The average waiting time per person: 7.285714285714286

The following parties were never seated:

party The Green Knight of 3 people.

**HW Grading:**

1. 10% - Follows style guidelines, including header and correct filename.
2. 40% - Compiles without errors
3. 50% - Correct implementation of all functions

**Bonus (5 points- capped at 100):**

* Compute the average waiting time per customer.