CS 610(Sec 851)
Prog Assignment 1

Shah Rahim
sr278
sr278@njit.edu

Instructions:

This application is bundled in an executable jar file making it easier to execute. However, it does require:

1. Java installed where this jar file will be run (the afs server does have java installed)
2. **Two** arg parameters: (duration of the check in, and average arrival time)

So the program will be run like:

java -jar assignment_1.jar 20 3

where 20 is the duration of the check in and 3 is the average arrival rate.

Executions:

When executing the jar file correctly, the program will first run **option 1** based on the input parameters. **NOTE:** average service rate = (average arrival rate * 5). There is a single queue and 5 stations processing the queue. The program will iterate a time counter and each time counter, it will display the current status, sleep, and then repeat. Some main statistics the program outputs per time increment include:

1. Total Duration
2. Num Passengers Arrival -- per time increment
3. Max Queue Length
4. Current Queue Length
5. Max wait time
6. Avg wait time
7. The Queue state
8. Each passengers wait time in the queue
9. Each stations state including current passenger, time left for service, and % busy

The program randomly generates an arrival amount, service time per passenger..etc through Java's gaussian library in the Random util class.

```
private static Integer getGaussian(Integer aMean){
    return (int)(aMean + fRandom.nextGaussian() * VARIANCE);
}
```

So, for every time interval the arrivals/service times are completely random based on the average input. Meaning, they will all average out to the input average, but have a nice distribution. The output will look like so:

OPTION 1 Simulation is about to begin

==========
==========

Total Duration: 1

Num Passengers Arrival: 5
Max Queue Length: 5
Current Queue Length: 0
Max wait time: 1
Avg wait time: 1

-->

Station 1 handling Passenger-0 time remaining: 12 Percent Busy: 100%
Station 2 handling Passenger-1 time remaining: 16 Percent Busy: 100%
Station 3 handling Passenger-2 time remaining: 7 Percent Busy: 100%
Station 4 handling Passenger-3 time remaining: 12 Percent Busy: 100%
Station 5 handling Passenger-4 time remaining: 18 Percent Busy: 100%

Total Duration: 2

Num Passengers Arrival: 4
Max Queue Length: 5
Current Queue Length: 4
Max wait time: 1
Avg wait time: 1

[Passenger-8 (Wait: 1)] --> [Passenger-7 (Wait: 1)] --> [Passenger-6 (Wait: 1)] -->
[Passenger-5 (Wait: 1)] -->

Station 1 handling Passenger-0 time remaining: 11 Percent Busy: 100%

Station 2 handling Passenger-1 time remaining: 15 Percent Busy: 100%
Station 3 handling Passenger-2 time remaining: 6 Percent Busy: 100%
Station 4 handling Passenger-3 time remaining: 11 Percent Busy: 100%
Station 5 handling Passenger-4 time remaining: 17 Percent Busy: 100%

Total Duration: 3

Num Passengers Arrival: 1
Max Queue Length: 5
Current Queue Length: 5
Max wait time: 1
Avg wait time: 1

[Passenger-9 (Wait: 1)] --> [Passenger-8 (Wait: 2)] --> [Passenger-7 (Wait: 2)] -->
[Passenger-6 (Wait: 2)] --> [Passenger-5 (Wait: 2)] -->

Station 1 handling Passenger-0 time remaining: 10 Percent Busy: 100%
Station 2 handling Passenger-1 time remaining: 14 Percent Busy: 100%
Station 3 handling Passenger-2 time remaining: 5 Percent Busy: 100%
Station 4 handling Passenger-3 time remaining: 10 Percent Busy: 100%
Station 5 handling Passenger-4 time remaining: 16 Percent Busy: 100%

This will continue until the queue is empty and the services are empty. Next, the program will run **Option 2 - Round Robin** on the same input args. There are five stations and each station has its own queue. The output will display the queue # along with its station # and the current status. The goal is that the program is sequentially output the state of the queues. The output will look like so:

OPTION 2 ROUND_ROBIN Simulation is about to begin

==========
==========

Total Duration: 1

Num Passengers Arrival: 4
Queue: 1

Max Queue Length: 1
Current Queue Length: 0
Max wait time: 1
Avg wait time: 1

-->

Station 1 handling Passenger-1 time remaining: 17 Percent Busy: 100%

Queue: 2

Max Queue Length: 1
Current Queue Length: 0
Max wait time: 1
Avg wait time: 1

-->

Station 2 handling Passenger-2 time remaining: 13 Percent Busy: 100%

Queue: 3

Max Queue Length: 1
Current Queue Length: 0
Max wait time: 1
Avg wait time: 1

-->

Station 3 handling Passenger-3 time remaining: 13 Percent Busy: 100%

Queue: 4

Max Queue Length: 1
Current Queue Length: 0
Max wait time: 1
Avg wait time: 1

-->

Station 4 handling Passenger-4 time remaining: 13 Percent Busy: 100%

Queue: 5

Max Queue Length: 0
Current Queue Length: 0
Max wait time: 0
Avg wait time: 0

-->

Station 5 handling null time remaining: null Percent Busy: 0%

This round robin policy will continue the same way as option one where the queue is empty The next one is **SHORTEST**:

OPTION 2 SHORTEST Simulation is about to begin

==========
==========

Total Duration: 1

Num Passengers Arrival: 3
Queue: 1

Max Queue Length: 1
Current Queue Length: 0
Max wait time: 1
Avg wait time: 1

-->

Station 1 handling Passenger-1 time remaining: 15 Percent Busy: 100%

Queue: 2

Max Queue Length: 1

Current Queue Length: 0
Max wait time: 1
Avg wait time: 1

-->

Station 2 handling Passenger-2 time remaining: 10 Percent Busy: 100%

Queue: 3

Max Queue Length: 1
Current Queue Length: 0
Max wait time: 1
Avg wait time: 1

-->

Station 3 handling Passenger-3 time remaining: 18 Percent Busy: 100%

Queue: 4

Max Queue Length: 0
Current Queue Length: 0
Max wait time: 0
Avg wait time: 0

-->

Station 4 handling null time remaining: null Percent Busy: 0%

Queue: 5

Max Queue Length: 0
Current Queue Length: 0
Max wait time: 0
Avg wait time: 0

-->

Station 5 handling null time remaining: null Percent Busy: 0%

Last is the **RANDOM** policy:

OPTION 2 RANDOM Simulation is about to begin

==========
==========

Total Duration: 1

Num Passengers Arrival: 6
Queue: 1

Max Queue Length: 2
Current Queue Length: 1
Max wait time: 1
Avg wait time: 1

[Passenger-6 (Wait: 1)] -->

Station 1 handling Passenger-3 time remaining: 21 Percent Busy: 100%

Queue: 2

Max Queue Length: 1
Current Queue Length: 0
Max wait time: 1
Avg wait time: 1

-->

Station 2 handling Passenger-4 time remaining: 9 Percent Busy: 100%

Queue: 3

Max Queue Length: 0
Current Queue Length: 0
Max wait time: 0
Avg wait time: 0

-->

Station 3 handling null time remaining: null Percent Busy: 0%

Queue: 4

Max Queue Length: 2
Current Queue Length: 1
Max wait time: 1
Avg wait time: 1

[Passenger-5 (Wait: 1)] -->

Station 4 handling Passenger-2 time remaining: 10 Percent Busy: 100%

Queue: 5

Max Queue Length: 1
Current Queue Length: 0
Max wait time: 1
Avg wait time: 1

-->

Station 5 handling Passenger-1 time remaining: 16 Percent Busy: 100%

All of these outputs will continue until the simulation is complete. In the submission of this project, there are **three** output log files where each log file contains the simulation for all options/policies for a given duration/avg arrival rate. They are named **test1.log, test2.log** and **test3.log**.

test1.log inputs inputs → ***java -jar assignment_1.jar 20 3***

test2.log inputs inputs → ***java -jar assignment_1.jar 10 2***

test3.log inputs inputs → ***java -jar assignment_1.jar 15 3***