Cairo University
Faculty of Computers & Information.
Operating Systems 1 Course
Third Year
Dr. Khalid Wassif
2018/2019

Assignment #2
Java Synchronization & Threading

It is required to simulate a petrol station program using Java threading and semaphore

Description

The petrol station should be designed to have n pumps; only one client can be served on a single pump at a time validating the following rules.

- The petrol station is initially empty.
- If a client arrives (print a message that a client has arrived) and if a free pump exists, the client should
    - Occupy pump
    - Get served.
    - Pay
    - Leave

Note: these actions will be represented by printed messages, such that there is a random waiting time between the printed messages when a client arrives, gets served, pays and leaves

- If a client arrives and all pumps are occupied, he must wait until one of the currently available clients finishes his service and leave.
- After finishing a client finishes his service, a client leaves and one of the waiting clients (if exist) will enter that pump.

Program Input:

- N: Number of pumps inside the petrol station.
- TC: total number of clients with their names (ex: C1, C2, C3…)

Program Output:

The execution order of the Clients' threads and the printed messages of each client

Bonus (2 grades): Implement a GUI for your simulation

Your GUI should show the simulation using boxes, images, colors…etc. not just making a text area and printing messages there as it will not be graded

Example:

Inputs:

```
- What is number of Pumps? 2
- Number of Clients : 4
- Clients' names: C1 C2 C3 C4
```

Output: (Note: output depends on the order of the executions of the threads)

```
- C1 arrived
- C2 arrived
- Pump 1: C1 Occupied
- Pump 2: C2 Occupied
- C4 arrived and waiting
- C3 arrived and waiting
- Pump 1: C1 Being Served
- Pump 1: C1 Paying
- Pump 2: C2 Being Served
- Pump 1: C1 Leave
- Pump 1 : C4 Occupied
- Pump 2: C2 Paying
- Pump 1 : C4 Being Served
- Pump 2: C2 Leave
- Pump 2: C3 Occupied
….
```

Submission instructions:

1. **Submission deadline date 24/11 on Acadox. NO LATE SUBMISSIONS ALLOWED**
2. **The assignment is submitted in group of maximum 3 students and minimum 2 students.**
3. **Team members MUST be from the same group**
4. **Discussion time will be decided by your TA.**
5. **All group members should understand the code delivered, anyone fails to answers will cause the whole group members to be deduced in grades**