I. INTRODUCTION

The second assignment aims to get students familiar with the practice of POSIX threads and semaphores. In this assignment, a control mechanism is going to be implemented for Boğaziçi (Bosphorus) Bridge to regulate the traffic flow.

According to 2006 records taken from General Directorate of Highways, approximately two hundred thousand vehicles cross Boğaziçi (Bosphorus) Bridge in a regular day. Since there is generally a heavy traffic flow on the bridge, some hard constraints have been being applied. One of them, the issue of regulating the pressure on the bridge piers and avoiding excess burden, has always have to be taken into consideration. For instance, maximum freight capacity the bridge can sustain is calculated as 6650 tons literally. For a secure transportation, however, a total of 2650 tons of freight is permitted to increase the availability and performance of the bridge. Since there is a bidirectional traffic, the flow on both sides has to be controlled to provide safety factors in terms of freight equilibrium. Distributing the pressure on the piers is not an easy question to be answered when there is a dense traffic on the way of the bridge, flowing from both European and Asian sides of the city.

In computer science point of view, if the vehicles crossing the bridge are considered as unique objects, solution for controlling traffic flow has to deal with some terms like atomicity, consistency, lastly but most importantly dynamism of variables/objects to improve memory utilization. In operating systems programming point of view, on the other hand, if these objects (vehicles crossing the bridge) are considered as unique threads running on the same environment (simply on a road), some mechanisms such as control threads, locks or semaphores (for example tollbooths on the bridge gates) must also be implemented to check the total freight in a simultaneous way.

II. DESIGN & IMPLEMENTATION (80 points)

In the following list, vehicle types having permission to cross the bridge are given. In fact, heavy transport vehicles are not permitted to cross the Boğaziçi (Bosphorus) Bridge. But according to our scenario, ten types of vehicles, including heavy transport vehicles, are listed with their properties and the bridge traffic allowance criteria.

<table>
<thead>
<tr>
<th>Vehicle Type</th>
<th>Vehicle ID</th>
<th>Weight (tons)</th>
<th>Passenger Capacity</th>
<th>Speed (kph)</th>
<th>Bridge Allowance</th>
<th>Traffic Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heavy truck</td>
<td>0</td>
<td>60.00</td>
<td>5</td>
<td>20</td>
<td>6</td>
<td>4%</td>
</tr>
<tr>
<td>Truck</td>
<td>1</td>
<td>50.00</td>
<td>4</td>
<td>30</td>
<td>10</td>
<td>5%</td>
</tr>
<tr>
<td>Haulage lorry</td>
<td>2</td>
<td>35.00</td>
<td>3</td>
<td>35</td>
<td>15</td>
<td>5%</td>
</tr>
<tr>
<td>Lorry</td>
<td>3</td>
<td>25.00</td>
<td>2</td>
<td>35</td>
<td>20</td>
<td>6%</td>
</tr>
<tr>
<td>Coach</td>
<td>4</td>
<td>20.00</td>
<td></td>
<td>40</td>
<td>40</td>
<td>5%</td>
</tr>
<tr>
<td>Public Bus</td>
<td>5</td>
<td>15.00</td>
<td>100</td>
<td>40</td>
<td>40</td>
<td>10%</td>
</tr>
<tr>
<td>Minibus</td>
<td>6</td>
<td>10.00</td>
<td>20</td>
<td>50</td>
<td>50</td>
<td>12%</td>
</tr>
<tr>
<td>Jeep</td>
<td>7</td>
<td>5.00</td>
<td>5</td>
<td>80</td>
<td>60</td>
<td>13%</td>
</tr>
<tr>
<td>Car</td>
<td>8</td>
<td>2.00</td>
<td>5</td>
<td>70</td>
<td>200</td>
<td>34%</td>
</tr>
<tr>
<td>Motocycle</td>
<td>9</td>
<td>0.50</td>
<td>2</td>
<td>50</td>
<td>400</td>
<td>6%</td>
</tr>
</tbody>
</table>

In our scenario, each vehicle in any type is going to be controlled by a total of six computers in tollbooths located on bridge gates, three each for both European and Asian sides. These computers in our implementation, however, are going to be unique threads which continuously control the current allowances, total freight, total passenger number on the bridge and LR equilibrium limit.

Since the maximum freight allowed is 2650 tons, as stated in the introduction part, these six threads assumed to be located on the bridge gates are going to ensure the current freight is always under the limit permitted. Besides, by counting the passenger capacity for each vehicle, these control threads will simultaneously calculate the total passengers on the bridge.
In an extreme emergency situation like bridge collapse, touch wood before, providing care from rescue ships of coast guard in Istanbul will not be sufficient if the number of passengers is higher than 6000. Hence, these threads should also ensure the current number of passengers on the bridge is less than 6000. On the other hand, since there is a bidirectional traffic, the flow on both sides (left and right sides) has to be controlled to provide safety factors in terms of freight equilibrium. The control threads have to be responsible for ensuring that LR equilibrium limit is maximum 20 tons, meaning that total freight of the traffic on the right side can be maximum 20 tons greater or less than of the traffic on the left side, or vice versa.

Briefly, the constraints below should be taken into consideration:

- Total number of passengers on the bridge cannot be higher than 6000.
- Total weight of the vehicles passing from the bridge cannot be higher than 2650 tons.
- The total weight sustained on both traffic ways (left and right) can be maximum 20 tons greater from each other.

Students are wanted to use macros to define these constraints like in the following example, so changing the values becomes easier. Testing the simulation with different, particularly with small constraint values will be a helpful approach in order to understand activities being controlled by the threads. Students are encouraged to play with these values in order to control the simulation in a requested way.

```c
#define TOTPASS 6000
#define TOTFRE 2650
#define LREQLIM 20
```

There must be some other control mechanisms that dynamically keep track of the vehicles crossing the bridge. For each vehicle type, a bridge allowance number is given in the list, indicating that maximum permitted number of that vehicle type on the bridge travelling at the same time. The synchronization between the control threads to check maximum allowance for each type of vehicles have to be achieved by semaphore implementation. For instance, there can be at most 6 heavy trucks on the bridge; meaning that with a semaphore implementation the seventh heavy truck is going to be waited until one among six heavy trucks leaves the bridge. Students are wanted to use macro definitions as shown in the following example for initial values of the semaphores going to be implemented and encouraged to play with these values in order to control the simulation in a requested way.

```c
#define MAXHEAVYTRUCK 6
#define MAXTRUCK 10
#define MAXHAULAGE 15
#define MAXMOTOCYCLE 400
```

Students have to consider that traffic never ends, for both sides. Creating a traffic flow for both sides is needed. The design issue for the traffic functions has to be considered thoroughly before implementation. For a better simulation, each of six control threads assumed to be located in the tollbooths on the gates of the bridge has to control each vehicle waiting to cross the bridge in a proper time, which is wanted to be determined by the students. The frequencies for each vehicle type in the traffic have to be taken into consideration. Besides, the time elapsed for each vehicle type during bridge crossing have to be calculated in order to simulate the traffic flow in a right manner. If the distance between the tollbooths located in Asia and Europe sides is measured as 1500 meters, according to the speed formula above, the time elapsed to pass the bridge for a heavy truck is 270 seconds whereas it is 67.5 seconds for a jeep. For a better representation of the program simulation, these values have to be converted into appropriate values. Scaling these real time values to appropriate ones is needed; so that an accelerated simulation demonstration becomes available. These scaled values can be altered easily by the utilization of macros, so that students can adjust the speed of their demonstrations.
In the demonstration part of the simulation, the current situation of the traffic on the bridge is wanted to be displayed and updated in several periods. Clarity and simplicity while showing most of the data being accessed/altered is expected. More creative ideas will get higher points.

In a brief manner, what wanted from the students in this assignment are listed:

- Traffic creation for both European and Asian sides (20 points)
  - Frequencies for each vehicle type in the traffic have to be taken into account.
  - Time elapsed for each vehicle type to cross the bridge must be controlled with a thread independent from control threads.
  - Time elapsed as if make a vehicle pass from the tollbooth for each control thread has to be added into the simulation.
- Control threads (3 located on Asian gates, 3 on European gates) which are responsible for ensuring:
  - Traffic constraints with mutexes (10 points)
    - Current total freight
    - Current number of passengers on the bridge
    - Current LR equilibrium limit
  - Bridge allowances for each vehicle type with semaphores (30 points)
- A clear, simple but wellrounded display for the demonstration of the simulation (20 points)

III. REPORT (20 points)

Students are wanted to write a report file that expresses the design, implementation and execution flow of their program including what they understood from threads, mutexes and semaphores. Functions, variables, mutexes, semaphores used have to be expressed briefly. The report must also include how the time units are converted and the algorithms designed for the two traffic creation functions. (15 points)

Answer all the following questions to get 5 more points. There is no wrong answer.

- Did you find this assignment worth to spend time in order to understand and learn the subject?
- What is the difficulty level of this assignment in your opinion? (Easy / Medium / Hard)
- How many effective hours/days did you spend to complete this assignment?
- Do you think the topics covered in the lab was sufficient to work on this assignment? If not, what was missing? (Do not forget that understanding and finding out how to use some new commands/programs is a part of the assignment.)
- How many assignments have you had, from other courses you have been taking, within the same time slot with this assignments? What are they?

ACKNOWLEDGMENTS

- Writing clean, readable code and using comments are recommended.
- Including the student name and ID in the top of the code, as comments is required.
- There will be only two files (hw2.c and the report) that will be submitted.
- The C file (hw2.c) and the report will be zipped as a tar file has a name of the student number, e.g. 240702030.zip.
- The homeworks will be sent to the mail address cse331-submit@cse.yeditepe.edu.tr
- No late submissions will be accepted!
- Students are welcomed to ask questions, come up with new ideas about the homework, but reading the instructions explained here carefully, also studying from the course book are highly recommended to have a general understanding before asking questions. Playing with the program implementation is encouraged. More creative ideas will get higher points.

Best regards,
Okan Türkeş