laSalle
UNIVERSITAT RAMON LLULL

Operating Systems 2023-2024

Session 9: Semaphores II 14/12/23

Goals

In previous sessions, we worked on the concept of threads and forks for parallelism and concurrency of tasks. The problem with working with shared resources and multi-process systems is that two processes try to access the same resource simultaneously, thus generating unpredictable or random results. On the other hand, sometimes the correct sequencing of tasks (synchronization) must also be guaranteed. The goal of this session is to introduce Operating Systems students to one of the most common synchronization and mutual exclusion mechanisms: **semaphores**. With the development of this session, it is intended that the student clarify the theoretical concepts of the use of semaphores and solve different problems.

Motivation

More specifically, with this session, the student will exercise:

- Semaphore creation (*SEM_constructor, SEM_init*).
- Control and destruction of semaphores (*SEM_destructor*).
- Operations on semaphores (*SEM_signal, SEM_wait*).
- Our own library semaphore.h.
- Use of the mutex library to use semaphores with threads.

Previous documentation

To complete this session, it is recommended to read the following references:

SALVADOR, J. (2014). *Programació en UNIX per a pràctiques de Sistemes Operatius*.

STEVENS, W. R. & RAGO S.A. (2008). Advanced Programming in the UNIX Environment, 2nd edition.

TANENBAUM, A. S. (2009). Modern Operating Systems, 3rd Edition.

F1

Tomorrow is the Monaco GP, and in today's practice, the pit crew has to show off as their season is at stake.

The pit crew consists of 3 sub-teams: Bolts Team, Fuel Team, Tires Team.

The functions of each team are as follows:

Bolts Team

As the name of the team indicates, they are in charge of removing and inserting the bolts, so the tires can be replaced.

Fuel Team

As the name of the team indicates, they are in charge of opening the gas tank, which has a secure anchoring system, filling the tank and closing it (it has an easy lock).

Tires Team

As the name of the team indicates, they are in charge of removing the tires and putting the new ones on so that the Bolt Team can put the bolts back on.

The procedure is as follows.

Both the Bolts Team and the Fuel Team start at the same time. That is, the Bolts Team starts by removing the bolts and the Fuel Team starts by opening and refilling the tank. Once the bolts have been removed, the Tires Team acts by removing and then putting on the new tires and finally the Bolts Team puts the bolts back on.

laSalle UNIVERSITAT RAMON LLULL

Operating Systems 2023-2024

Session 9: Semaphores II 14/12/23

Throughout the season, we have been measuring the times of the teams to make estimations and we know that:

Removing bolts	2 – 1 s
Putting in bolts	2 – 1 s
Opening tank	1s
Refilling tank	4 – 7 s
Removing tires	2 – 1 s
Putting on tires	2 – 1 s

Each team works independently, so we already give you a hint that each one will be a different thread. At the start of any action, a message of what it's being carried out appears on the screen (i.e. *Removing bolts*). Note that there are teams that depend on others (Bolts Team <-> Tires Team), as they cannot place the tires if the bolts have not been removed for example.

Once the team has completed the task, they shout out what they have accomplished along with how long it took (i.e. Bolts out!!!). The time will be a random number between the estimations specified before.

At the end, a count is made of all the time the driver has stayed in the pits. To calculate this time, the following schemes are used to clarify how the hole Pit Team works.

Bolts Team & Tires Team

|

Removing bolts

|

Removing tires

|

Putting on tires

|

Putting in bolts

Fuel Team

|

Opening tank

|

Refilling tank

Total time = Bolts Team + Tires Team

Bolts Team & Tires Team

|

Removing bolts

|

Removing tires

|

Putting on tires

|

Putting in bolts

Fuel Team

|

Opening tank

|

|

|

|

|

Refilling tank

Total time = Fuel Team

The whole Pit Team has to be tested in 2 scenarios:

1. One pilot is coming to pit stop.
2. Two pilots are coming to pit stop.

There is a file containing the name of the 2 pilots, with the following format.

NamePilot\nNamePilot\n

We will specify the file and the number of pilots on the arguments of the program as follows, so depending on the number, one or both names of the file will need to be read, because before starting the simulation, the name of the actual pilot will be displayed on the screen.

Execution Examples

```
                        ~>./sem2 drivers.txt 2
[Max Verstappen]

[Bolts TEAM]    * Removing bolts *
[Fuel  TEAM]    * Opening tank *        [1 s]
[Tires TEAM]    * Removing tires *
[Bolts TEAM]    Bolts out!!!            [1 s]
[Fuel  TEAM]    * Refilling tank *
[Tires TEAM]    Tires removed!!!        [1 s]
[Tires TEAM]    * Putting on tires *
[Tires TEAM]    Tires on!!!             [2 s]
[Bolts TEAM]    * Putting in bolts *
[Fuel  TEAM]    Tank refilled!!!        [4 s]
[Bolts TEAM]    Bolts on!!!             [1 s]
-----------------------------------------------
Boxes time [Max Verstappen]             [5 s]



[Sergio Pérez]

[Bolts TEAM]    * Removing bolts *
[Fuel  TEAM]    * Opening tank *        [1 s]
[Fuel  TEAM]    * Refilling tank *
[Bolts TEAM]    Bolts out!!!            [2 s]
[Tires TEAM]    * Removing tires *
[Tires TEAM]    Tires removed!!!        [1 s]
[Tires TEAM]    * Putting on tires *
[Tires TEAM]    Tires on!!!             [1 s]
[Bolts TEAM]    * Putting in bolts *
[Fuel  TEAM]    Tank refilled!!!        [4 s]
[Bolts TEAM]    Bolts on!!!             [1 s]
-----------------------------------------------
Boxes time [Sergio Pérez]               [5 s]
```

```
                                ~>./sem2 drivers.txt 1
[Max Verstappen]

[Bolts TEAM]    * Removing bolts *
[Fuel  TEAM]    * Opening tank *          [1 s]
[Fuel  TEAM]    * Refilling tank *
[Tires TEAM]    * Removing tires *
[Bolts TEAM]    Bolts out!!!              [2 s]
[Tires TEAM]    Tires removed!!!          [3 s]
[Tires TEAM]    * Putting on tires *
[Tires TEAM]    Tires on!!!               [2 s]
[Fuel  TEAM]    Tank refilled!!!          [6 s]
[Bolts TEAM]    * Putting in bolts *
[Bolts TEAM]    Bolts on!!!               [2 s]
----------------------------------------------
Boxes time [Max Verstappen]               [7 s]
```

laSalle
UNIVERSITAT RAMON LLULL

Operating Systems 2023-2024

Session 9: Semaphores II 14/12/23

Considerations

- It is mandatory to implement it using threads.
- It must be ensured that at the end of the execution all the processes/resources are cleaned up properly.
- The use of global variables is not allowed, except for those that you think are essential.
- All inputs and outputs must be done using file descriptors. **The use of printf, scanf, FILE*, getchar or similar is NOT allowed.**
- **The use of the functions "system", "popen", or from the same family is NOT allowed**.
- **You must compile using -Wall, -Wextra, and -lpthread flags**
- **Any deliverable containing warnings or errors will be directly discarded.**
- At the start of the .c you must include a comment with your logins, names, and surnames.
- **For submitting the session, you must hand over a file "S9.c", and deliver it through eStudy platform.**