Semaphores **SCOMP-PL4**

1. Implement a program that creates 8 new processes.

   • Each process reads 200 Numbers from a text file ("Numbers.txt") and write those numbers to the file "Output.txt";

   • At any time only one child should be able to read/write into the files (one process in each of the files);

   • At the end, the father process should print the content of the file "Output.txt".

2. Change the last program in a way that the child process reads from the file "Numbers.txt" by order of its process number (1 to 8). Use 8 semaphores to synchronize between the child processes.

3. Implement a program that should behave as follows:

   • Guarantee mutual exclusion to the shared memory;

   • Add, to the shared memory, a text line with the following information: "I'm the Father - with PID X", where X should be replaced by the PID of the process;

   • Wait N seconds (from 1 to 5 seconds);

   Test the solution by executing several concurrent instances of your code.

   Note: Consider that you have a shared memory area with 50 strings, each string having 80 characters.

4. Change the last question by adding the following features:

   - Add an option to "remove" (free) one string from the shared memory;

   - Your program should wait at most 12 seconds to access the shared memory, warning the user if that is not possible.

5. Implement a program that creates a new process. Use semaphores to synchronize their actions.

   • The child process writes on screen the message "I'm the child";

   • The father process writes on scree the message "I'm the father";

   • The child process should be (always) the first to write the message on screen.

6. Change the program above in order to write 15 messages, alternating between father and child.

7. Consider the following child processes operations:

Process 1	Process 2	Process 3
`printf("Sistemas ");`	`printf("de ");`	`printf("Computadores -");`
`printf("a ");`	`printf("melhor ");`	`printf("disciplina! ");`

Add operations on semaphores such that the result printed is: "Sistemas de Computadores – a melhor disciplina!".

8. Consider the two following processes:

   P1: while true do: print "S"

   P2: while true do: print "C"

   Add semaphores such that at any moment the number of S or C differs by at most one. The solution should allow strings such as: "SCCSSCCSCS"

9. Consider the two following processes:

Process 1	Process 2
`buy_chips();`	`buy_beer();`
`eat_and_drink();`	`eat_and_drink();`

Add semaphores such that will ensure that neither of P1 or P2 eat and drink until both the beer and chips are acquired!

10. A company wants to develop an application which allows to consult and insert personal data records (Number, Name, Address), for that is needs to develop the following programs:

    • Consult – This application allows you to consult a user's personal data. To do this, you must ask the user for his or her identification number.

    • Insert – Allows you to enter a user's data in the shared memory area.

    • Consult_All – Lists all data in the shared memory area. You should only display the records that contain valid data.

    Notes:

    - The application should support several Consults and Inserts in parallel.

    - Consider that there is a shared memory region that has the capacity to store 100 records.

11. Assume a train has 200 passengers capacity and has three doors. Use semaphores to develop a solution that ensures:

    - the train will never be overloaded;

    - the passengers will not collide with each other when getting on or getting off the train.

12. Implement a program that simulates the selling of tickets. You should simulate the customers, using semaphores to synchronize the access to shared memory area for data exchange. Consider also the following:

    - Only the seller has access to the tickets, only he knows the number of the next ticket.
    - The clients should be served by their arrival order, blocking until getting their ticket.
    - The client should print the number of the ticket.
    - The behavior of each client and seller should be developed by separate programs.
    - Assume that each client needs a random number of seconds (1-10) to be served.
    - You should try to execute several clients concurrently.

13. Implement a program that starts by creating two new processes. Each of the child processes will have the role of producer, while the father will act as a consumer. Between them there a circular buffer capable of handling 10 integers on a shared memory area. Each of the producer writes increasing values into the buffer, which should be printed by all. Assume that only 30 values are exchanged and that a semaphore controls the buffer access:

    • producer blocks if the buffer is full;

    • producer only writes after guaranteeing mutual exclusion;

    • consumer blocks until there are no new data to read.

14. Implement the following set of programs:

    • **Reader**, reads a string to a shared memory area:

    – The readers do not change the shared memory area, therefore several readers can in parallel access the shared memory area;

    – Each reader should print a string and the number of readers at the time.

    – Readers can only access shared memory when there are no writers;

• **Writer**, writes on the shared memory area:

– Writes its PID and current time;

– Writers have priority over readers;

– Only one writer can access the shared memory area at the same time;

– Each writer prints the number of writers as also the number of readers.

Execute several readers and writers at the same time to test your software.

15. Consider a show room and a set of visitors. Each visitor is simulated by an infinity loop in which he enters the show room and leaves after a while. Each show in a room starts and ends at defined times, the visitors can enter and leaving the room at those times. Each room supports 5 visitors if more want to enter the room then they have to wait.

16. Consider a bridge with only one lane that permits traffic in both East-West directions. A car can only enter the bridge if the bridge is not occupied by cars going in the opposite direction. If that is the case, the car must wait until the bridge allows traffic in the other direction. Assume that a car takes 30 seconds to cross the bridge. The bridge's current allowed direction changes whenever the last car finishes its crossing.

17. Consider a movie theater with a maximum capacity of 300 spectators. There are three types of spectators: VIP, Special and Normal. While there are free seats, clients enter the movie theater according to their arrival order. Whenever the movie theater is full, the entrance of the spectators is conditioned to the exit of other spectators. In this case, priority should be given to waiting VIP, then Special, and only then Normal spectators (by that order).