CS344: Operating Systems Lab

Lab # 08 (1 Questions, 127 Points)
Held on 17-Oct-2023
Lab Timings: 09:00 to 12:00 Hours   Pages: 2
Submission: 12:00 Hrs, 17-Oct-2023
Instructor Dr. V. Vijaya saradhi
Head TAs Adithya Moorthy & Laxita Agrawal
Department of CSE, IIT Guwahati

a. This assignment is based on chapter 5, `Process Synchronization` in the book `Operating System Principles, Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne`.

b. In order to perform this assignment, understanding of system calls `ftok()`, `semget()`, `semop()`, `semctl()`, `fork()`, `shmget()`, `shmat()`, `shmctl()`, `shmdt()`, `shmctl()` are essential.

c. Read the attached resource for a understanding about system V semaphore implementation and system V shared memory implementation

d. Read chapter 5 on process synchronization. In particular readers-writers problem.

e. Carefully read the manual pages for the above system calls.

f. Read the system architecture given in the question which helps develop the solution.

Question 1: (127 points)

Readers-writers problem - Version control system implementation: Write C program as described below

a. (5 marks) Implement `sem_wait` function which will initialize `sembuf` structure with appropriate values and performs `semop`.

b. (5 marks) Implement `sem_signal` function which will initialize `sembuf` structure with appropriate values and performs `semop`.

c. (10 marks) Write a program `repository` to create a shared memory segment of required size. Initialize the shared memory with `Hello world!` program. That is a C program which prints `Hello world!`. Keep this program alive in sleep mode till the last three tasks are performed.

d. (38 marks) A writer should perform the following tasks

- (2 marks) Obtain a key
- (5 marks) Get the shared memory object
- (2 marks) Attach the shared memory object
- (2 marks) obtain another key
- (5 marks) Declare appropriate semaphores.
- (5 marks) Initialize the semaphores.

- (5 marks) Perform appropriate semaphore operations
- (5 marks) To the `Hello World!` program, residing in the shared memory, add one more output line `"Hello written by pid: <number>"` and write into the shared memory.
- (5 marks) Perform appropriate semaphore operations
- (2 marks) Detach the shared memory.

e. (49 marks) A reader program that should perform the following tasks:

   - (2 marks) Obtain a key
   - (5 marks) Declare appropriate semaphores.
   - (5 marks) Initialize the semaphores.
   - (2 marks) Obtain another key
   - (5 marks) Get the shared memory object
   - (5 marks) Attach the shared memory object
     i. (5 marks) Performing appropriate semaphore operations
     ii. (5 marks) Count number of readers
     iii. (5 marks) Print the program in the shared memory
     iv. (5 marks) Performing appropriate semaphore operations
   - (5 marks) Remove the semaphore
   - (5 marks) Detaches the shared memory object

f. (5 marks) Demonstrate that while a write is writing, no other writer should be allowed to write code.

g. (5 marks) Demonstrate that while a writer is writing, no reader should be allowed to access to read the code

h. (5 marks) Demonstrate that while a reader is reading, other readers can be able to read the code

i. For the three tasks above, write a help file `README.txt` file describing how you achieve task (f), (g) and (h) respectively.