
CS344: Operating Systems Lab

Lab # 09 (1 Questions, 91 Points)

Held on 31-Oct-2023

Lab Timings: 09:00 to 12:00 Hours Pages: 2

Submission: 12:00 Hrs, 31-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 dining philosophers' 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: (91 points)

Dining Philosophers - database transactions example: Write one single 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. (2 marks) Declare 5 database tables corresponding to the structure having the following members
 - name - character array of size 50 which holds table name
 - pid - an int array of size 10
- d. (2 marks) Obtain a key
- e. (5 marks) Declare 5 semaphores corresponding to 5 database tables
- f. (5 marks) Initialize all the semaphores
- g. (5 marks) Create shared memory corresponding to 5 database tables
- h. (5 marks) Attach the shared memory
- i. (5 marks) Initialize each table with table number.

- j. (5 marks) Create five children each corresponding to one philosopher
- k. Repeat the following tasks in 5 rounds. Figure below shows two rounds R_1 and R_2 . In each round philosopher's sits in different positions of the table as shown.
 - i. (5 marks) Each child (philosopher) picks two adjacent database tables (equivalent to picking two forks)
 - ii. (5 marks) While performing the above, acquire appropriate semaphores and prints the message "Transaction <pid> holding the semaphore".
 - iii. (5 marks) Perform the following:
 - Print "Transaction <pid>: operating on Table <number> - Table <number>".
 - Write the traction pid into the respective tables structure.
 - iv. (2 marks) Sleep for 10 seconds and releases the semaphores.
- l. (20 marks) Ensure that every philosopher write into the pid array of every table by following the round structure shown in the figure. That is, print the contents of every table at the end of program.
- m. (10 marks) Every philosopher eventually get to complete the transaction.

