Operating Systems Design - 19CS2106S, 19CS2106A

Home Assignment - CO-IV

Mode of submission: Post Handwritten scanned documents in LMS

Submission date: on or before 10th November 2020

1. Write a UNIX system program to Solve Producer Consumer problem using POSIX semaphores.
   Three conditions must be maintained by the code when the shared buffer is considered as a circular buffer:

   1. The consumer cannot try to remove an item from the buffer when the buffer is empty.

   2. The producer cannot try to place an item into the buffer when the buffer is full.

   3. Shared variables may describe the current state of the buffer (indexes, counts, linked list pointers, etc.), so all buffer manipulations by the producer and consumer must be protected to avoid any race conditions.

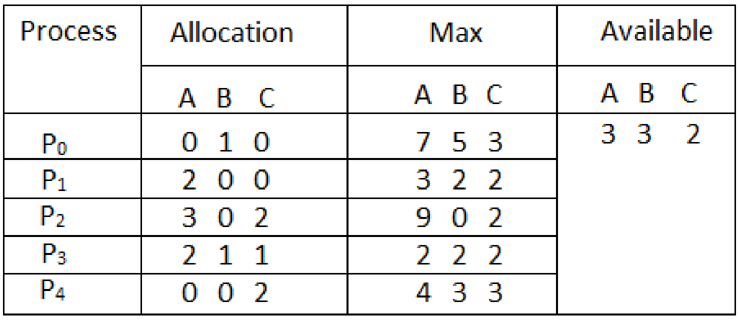
   Your solution using semaphores should demonstrates three different types of semaphores:

   1. A binary semaphore named mutex protects the critical regions: inserting a data item into the buffer (for the producer) and removing a data item from the buffer (for the consumer). A binary semaphore that is used as a mutex is initialized to 1. (Obviously we could use a real mutex for this, instead of a binary semaphore.)

   2. A counting semaphore named nempty counts the number of empty slots in the buffer. This semaphore is initialized to the number of slots in the buffer (NBUFF).

   3. A counting semaphore named nstored counts the number of filled slots in the buffer. This semaphore is initialized to 0, since the buffer is initially empty

2. Considering a system with five processes P0 through P4 and three resources types A, B, C. Resource type A has 10 instances, B has 5 instances and type C has 7 instances. Suppose at time t0 following snapshot of the system has been taken:



Process	Allocation	Max	Available
	A B C	A B C	A B C
P_0	0 1 0	7 5 3	3 3 2
P_1	2 0 0	3 2 2	
P_2	3 0 2	9 0 2	
P_3	2 1 1	2 2 2	
P_4	0 0 2	4 3 3	

   i. **What will be the content of the Need matrix?**
   ii. **Is the system in safe state? If Yes, then what is the safe sequence?**
   iii. **What will happen if process P_1 requests one additional instance of resource type A and two instances of resource type C?**

3. Write a Program using pthreads to demonstrate deadlock.
4. Solve Readers-Writers Problem using counter and 2 semaphores.