CSE/IT Batch : Hinglish

Operating System

Process Synchronization Part - 2

DPP-02

[MSQ]

1. Consider the following code shared among three process P_1, P_2 and P_3.

```
void incr( )
   {
    lockAcq(lockP);
     a++;
     lockRele(lockP);
     lockAcq(lockQ); /* Here we used a different lock
      a++;
     lockRel(lockQ);
   }
```

If 3 processes are executing the above block of code concurrently, what will be the values of "a" at the end NOTE that a is initialized to zero initially.

(a) 6 (b) 4
(c) 3 (d) 5

[MSQ]

2. Choose the correct statements about Bakery algorithm

(i) Bakery algorithm sustains FCFS property.
(ii) Bakery algorithm provides a solution to C.S problem which means the algorithm provides solution when there are N processes that can access C.S.
(iii) This algorithm can be used in 2 process mutual exclusion problem.
(iv) None

(a) (i), (ii) & (ii) (b) (ii) & (iii)
(c) (iii) & (i) (d) None

[MSQ]

3. Consider the following solution to C.S problem using atomic Exch (exchange) instructions Exch function exchanges the data of two variables atomically.

```
void Exch(boolean *p, boolean *q)
   {
       boolean temp = *p;
          *p = *q;
          *q = temp;
    }
boolean lock = FALSE;
process pi;
do
  {
   /* Each process has a local Boolean variable key*/
      key = TRUE;
      while (lock = = TRUE);
      Exch(& lock, & key);
      /* critical section */
      lock = FALSE;
      /* remainder section */
    } while (TRUE);
```

(i) It ensures ME and progress.
(ii) It ensures ME and B.W.
(iii) It ensures B.W.
(iv) It ensures all the three, ME, progress and B.W.

(a) (i) only (b) (ii) only
(c) only (iii) (d) (i) & (iii)

[MSQ]

4. Choose the correct statements from the following:

(i) lock variables violates independence of speed.

(ii) strict alternation violates "progress" goal for IPC.

(iii) peterson's solution satisfies all the three major goals for IPC.

(iv) none

(a) (i) & (ii)

(b) (ii) & (iii)

(c) (i) (ii) & (iii)

(d) (iv)

[MCQ]

5. The hardware solution which provides mutual exclusion is

(a) Test and set instructions

(b) Peterson's solution

(c) Dekker's algorithm

(d) None

[MCQ]

6. If the output of a computation depends on the speed of the processes involved, then it is said to be a

(a) deadlock

(b) time lock

(c) cycle stealing

(d) race condtion

Answer Key

1. **(a, b, d)**
2. **(a)**
3. **(a)**
4. **(c)**
5. **(a)**
6. **(d)**

Hints and solutions

1. (a, b, d)

Lets suppose, P_1 acquires lock P[lockAcq(lockP), so, it will increment "a" to 1. Now P_1 releases lock P and acquires lockQ, process P_2 gains lockP, Now process P_1 and process P_2 increases the value of "a" simultaneously to '2'. P_1 release lockQ and P_2 releases lockP. P_3 acquires lock P and process P_2 acquires lockQ. P_3 and P_2 increments the value of "a" to 3 simultaneously. P_3 and P_2 both releases the lock and at last process P_3 acquires lock Q and increments the value of "a" to 4 and releases to lock. Hence value of "a" will be 4.

In case, P_1 and P_2 don't increment the value of "a" at a time. So "a" will become 3 and also value of "a" will become 5 after all the locks are releases by P_1, P_2 and P_3. And also if no processes tries to increment the value of "a" simultaneously then "a" will be 6.

2. (a)

All the given statements are true except (iv).

3. (a)

M.E is ensured, Since Each function is atomic, only one process be able to execute Exch code at a time and key is the local variable for each processes even if both processes makes the value of key = TRUE simultaneously, because of atomic Exch function value of lock becomes true and key becomes false for the first process and first process is able to get into C.S. Second process reads the value of lock and key as true.

Hence 2nd process will get stuck in while statement for that Exch function does nothing.

2. B.W is violated as a process can enter many times in C.S.
3. Progress is guaranteed between processes since the processes can enter into the critical region independently and hence there is no deadlock situation.

4. (c)

Strict alternation violates "progress" goal for IPS (a process executing non C.S region may prevent other processes from getting out of its busy loop and into its C.S).

5. (a)

As peterson's solution and Dekker's algorithms are software based solution to C.S problem.

Hence these option are eliminated. Test and set instruction is a hardware solution to a C.S problem.

6. (d)

Here speed means ordering of the process. So the race condition occurs when the state of a system depends on the ordering of the processes.