CSE/IT   Batch : Hinglish

Operating System
Process Synchronization Part - 3 | DPP-03 |

[MCQ]

1. Producer-Consumer can also be solved by using.
   (a) monitors
   (b) semaphores
   (c) both (a) and (b)
   (d) none

[MSQ]

2. The problems which are solved using semaphores are
   (i) Mutual exclusion
   (ii) Race condition
   (iii) Process synchronization
   (iv) None
   (a) (i) & (ii)
   (b) (i) & (iii)
   (c) (i) & (ii) & (iii)
   (d) None

[MSQ]

3. Processes x_1 and x_2 uses flag_critical in the following function to achieve mutual exclusion. Assuming flag_critical is initialized FALSE initially.

```
get_access
  {
     if(flag_critical = =FALSE)
      {
         flag_critical = TRUE;
         critical_section();
       flag_critical = FALSE;
      }
    {
```

Consider the following statements:
   (i) The above routine may lead to deadlock.
   (ii) it is possible for processes x_1 and x_2 to access critical section concurrently.
   (a) (i) is true and (ii) is false
   (b) (ii) is true (i) is false
   (c) Both (i) and (ii) are True
   (d) Both (i) and (ii) are false

[MSQ]

4. Consider the code given below, used by the processes x_1 and x_2 to access critical section.
The initial value of shared boolean variable P and Q are false

```
        x1                    x2
   while(P = = Q);      while(P! = Q);
   <critical section>   <critical section>
   P = !(Q);            P = Q;
```

Select the true statements from the following:
   (i) Process x_2 can go into critical section just after one entry by process x_1 into its critical section.
   (ii) Mutual exclusion is not ensured.
   (iii) Process x_1 can go into critical section many times without single entry of x_2 into its critical section.
   (iv) none of the above
   (a) (i) & (ii)
   (b) (ii) & (iii)
   (c) only (i)
   (d) none (iv)

[MCQ]

5. Match the following statements

List I	List II
A. Critical section	1. Ensuring that only one process can execute C.S.
B. Synchronization	2. atomic operation are used to ensure co-operation between processes.
C. Mutual exclusion	3. Section of code that only one process can access at once.

Matches:

	A	B	C
(a)	1	2	3
(b)	3	2	1
(c)	2	3	1
(d)	1	3	2

Answer Key

1. (c)
2. (b)
3. (b)

4. (c)
5. (b)

Hints and Solutions

1. (c)
- There will be a fixed sized buffer and producer produces the items and enters then into the buffer and consumer removes the item from the buffer and consumes them, when solved with semaphore.
- Mutex ensure that at any particular point of time only producer or consumer is accessing the buffer achieving M.E.

2. (b)

Semaphores are used to solve the problem of synchronization between the processes to achieve mutual exclusion.

3. (b)

(ii) is true because, both the processes x_1 and x_2 can access critical region concurrently because of
if (flag_critical = = FALSE).

x_1, x_2 can execute the above condition Simultaneously and can enter C.S without leading to deadlock.

4. (c)

Process x_1 cannot go into critical section multiple times without entry of x_2.

5. (b)
- Synchronization uses p() and v() operation.
- Critical section is a section of code that only one process can access at a time.
- Mutual exclusion ensures that only one process can execute CS at any time.