CS & IT ENGINEERING

Operating System

Process Synchronization / Coordination

DPP 07 (Discussion Notes)

By- Anjnee Bhatnagar ma'am

01 Question

02 Discussion

Q.1

What is a compulsory step before using semaphore?

[MCQ]

A. Deciding final value of semaphore

B. Initialization of semaphore

C. Defining number of operations to be performed

D. All of the above.

Q.2 Consider the following statements, which of the following is correct?

[MCQ]

A. Semaphore are atomic in nature and implemented in user's mode.

B. Semaphore are atomic in nature and implemented in kernel mode.

C. Semaphore are non-atomic in nature and implemented in user's mode.

D. Semaphore are non-atomic in nature and implemented in kernel mode.

. Reading + Updating + Writing

X=10 + X=X+100 + X=110

R U W

Variable: int x = 10 / 110

Q.3

If a semaphore's value is "– 3", then what does magnitude of "–ve" value indicate ____?

[MCQ]

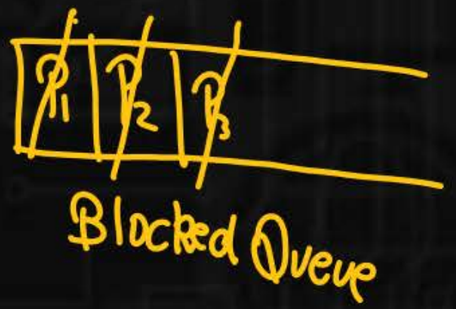
A. Number of successful up operation.

B. Number of successful down operation.

C. Number of blocked processes.

D. Number of unblocked processes.

S = -3



Q.4

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

[MCQ]

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

1. X_1 2. X_2

X_1 is preempted

false

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.

Consider the following statement:

(i) The above routine may lead to deadlock. ✗

(ii) It is possible for processes x1 and x2 to access critical section concurrently. ✓

1. M.E
2. Hold & Wait
3. Circular Wait
4. No preemption.

Q.5

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

[MCQ]

x_1	x_2
while (P == Q);	while (P! = Q);
<critical section>	<critical section>
P = Q;	P = !(Q);

Initialized Value.

P = Q = False

P = True

False

Select the true statements from the following:

(i) Process x_1 can go into critical section just after one entry by process x_2 into its critical section.

(ii) Mutual exclusion is not ensured.

$x_2 x_1 x_2 x_1 x_2 x_1 x_2 \ldots$

(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. (i), (ii) & (iii)

Q.6

Consider the two function P_i and P_j that share a variable Q with an initial value '3' execute concurrently:

[NAT]

P_i()	P_j()
{ 3	{
1. R = Q * 2; := 6	3. S = Q + 1; := 4
2. Q = R;	4. Q = S;
} Q=6	} Q=4

- 3,4,1,2 ⇒ Q = 8
- 1,2,3,4 ⇒ Q = 7
- 3,1,2,4 ⇒ Q = 4
- 1,3,4,2 ⇒ Q = 6

Initial Q = ~~3~~ 6

What are the different possible value for variable Q at the end of execution of both process P_i and P_j?

Q = 4, 6, 7, 8

4

Q.7

Match the following statements

[MCQ]

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-3, B-2, C-1

	A	B	C
A.	1	2	3
B.	3	2	1
C.	2	3	1
D.	1	3	2

Q.8

Let S be a binary semaphore variable. Let S = 1 initially. Assume that no blocked processes exist in the system. The following operations are performed on semaphore S.

6 P, 8 V, 12 P, 11 V, 19 P

The number of blocked processes after executing these operations are 19.

[NAT]

S = 0

S = 1

S	~~1~~ 0	~~0~~ 1	~~1~~ 0	0	0
P/V	6P	8V	12P	11V	19P
Blocked Processes	5	0	11	0	19


THANK YOU GW
SOLDIERS !