



## COMSATS University Islamabad, Lahore Campus

Block-C, Department of Computer Science

COMSATS University Islamabad, Lahore Campus 1.5KM Defence Road, Off Raiwind Road, Lahore

### Sessional – II Examination – Semester Spring 2021

Course Title:	Operating Systems			Course Code:	CSC322	Credit Hours:	3(2,1)
Course Instructor/s:	Nadeem Ghafoor Chaudhry			Programme Name:	BS Computer Science		
Semester:	5 <sup>th</sup>	Batch:	SP19-BCS	Section:	B,C	Date:	Wednesday, 5 <sup>th</sup> May 2021
<b>Time Allowed:</b>	<b>90 Minutes</b>			<b>Maximum Marks:</b>	<b>22</b>		
Student's Name				Reg. No.			

#### Important Instructions / Guidelines:

- Answer all questions.
- Do not give multiple answers for a question. Clearly cross out what you do not want me to read.
- Do not make multiple submissions.
- Follow the detailed instructions already posted on Google Classroom

Q1) [2.5+2.5 = 5 Marks] We discussed four conditions that must exist simultaneously for deadlocks to occur. We tried to invalidate each one of these four conditions to prevent deadlocks from taking place but came across Starvation and Low Resource Utilization and as new problems. Very briefly explain both with an example for each.

Q2) [ 1x5 = 5 Marks] Given below is the pseudocode to synchronize processes P1, P2, P3, P4, P5, and P6 by using three semaphores X, Y and Z that are initialized as follow: X=0, Y=0, Z= -2. The Operating System can schedule the processes which are ready to execute in any order. List down at least 5 possible orders of execution in which the processes can run.

P1	P2	P3	P4	P5	P6
Print P1 Signal(Z) Signal(Z)	Wait(Y) Print P2	Wait(X) Print P3 Signal(Y) Signal(Y)	Wait(Z) Print P4	Print P5 Signal(X)	Wait(Y) Print P6 Signal(Z)

Q3) [ 6 Marks] Does Peterson's Solution satisfies the Bounded-waiting and Progress requirements? If it does, then prove how, if not then state why not. Here is Peterson's Solution:

```
while (true){
    flag[i] = true;
    turn = j;
    while (flag[j] && turn == j)
        ;
    /* critical section */
    flag[i] = false;
    /* remainder section */
}
```

}

Q4) [4+2 = 6 Marks] Imagine there are following three processes in our system.

P0 – A file backup batch process

P1 – A user editing a video

P2 – An electricity bill generating and printing process

And there are four types of resources R1, R2, R3 and R4

The current scenario is as follows:

Max

R1	R2	R3	R4
4	2	3	1

Available

R1	R2	R3	R4
2	1	0	0

Allocation

	R1	R2	R3	R4
P0	0	0	1	0
P1	2	0	0	1
P2	0	1	2	0

Request

	R1	R2	R3	R4
P0	2	0	0	1
P1	1	0	1	0
P2	2	1	0	0

- Determine if the system is in a deadlock state or not. You must show all steps, just a YES/NO answer will not be accepted.
- Ignore the above scenario and imagine that all three processes are in deadlock. What will you do and why to recover from the deadlock?