Register Number									

Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam – 603 110

(An Autonomous Institution, Affiliated to Anna University, Chennai)

Department of Computer Science and Engineering

Continuous Assessment Test – 3

Question Paper

Degree & Branch	B.E. CSE				**Semester**	IV
Subject Code & Name	UCS1404 Database Management Systems				**Regulation: 2018**	
Academic Year	2020-2021	**Batch**	2019-2023	**Date**	**26.04.2021**	**FN 11.00 to 11.40 AM**
Time: 40 Minutes	**Answer All Questions**				**Maximum: 20 Marks**	

Part – B Answer all the questions (2×10 = 20 Marks)

<K3>	1. Consider the three transactions T_1, T_2 and T_3 and the schedule S_1 as given below: T_1: $r_1(A)$, $w_1(A)$, $r_1(C)$, $w_1(C)$ T_2: $w_2(A)$ T_3: $r_3(B)$, $w_3(B)$, $r_3(C)$, $w_3(C)$ S_1: $r_3(B)$, $\mathbf{r_1(A)}$, $\mathbf{w_2(A)}$, $w_3(B)$, $w_1(A)$, $r_3(C)$, $w_3(C)$, $r_1(C)$, $w_1(C)$ a) Draw the precedence graph for S_1 and state whether the schedule is serializable or not. Why or why not? (1) b) Now swap the operations in S_1 that is highlighted in bold and consider as S_2. Check whether the schedule S_2 is *conflict serializable* through the swapping of operations. If so, give the equivalent serial schedule(s). (6) c) Is the schedule S_2 is view serializable or not? Justify. If so, determine the equivalent serial schedule(s). (3)	**<CO4>**
	(OR)	
<K3>	2. Consider the three transactions T_1, T_2 and T_3 and the schedule S_3 as given below: S_3: $w_1(A)$, $r_1(B)$, $r_1(C)$, $w_3(B)$, $w_1(B)$, $w_3(C)$, $r_2(C)$, $w_2(B)$, $w_2(C)$, $r_3(A)$ a) Is this schedule conflict serializable ? (Use precedence graph or swapping of operations). Why or why not? If so, give the equivalent serial schedule. (4) b) Is this schedule view serializable? Why or why not? If so, give the equivalent serial schedule. (6)	**<CO4>**
<K2>	3. Consider the two transactions T 1 and T 2 : T1 : r 1 (Y); r 1 (X); w 1 (X); T2 : r 2 (X); w 2 (X);	**<CO4>**

<table>
<tr><td></td><td>Assume that the schedule to be generated from the above transactions must use two-phase locking protocol (2PL) using shared / exclusive locks. The schedule does not allow upgradation / degradation of locks.
a) Without changing the order of operations in transaction, write a serializable schedule that follows basic 2PL. (4)
b) Now consider the modified version of transaction T2 as T2'' :
T2'' : r 2 (X); r 2 (Y); w 2 (X);
and add the operation commit at the end of each transaction T1 and T2'' . Write a serializable schedule that implements strict 2PL. (6)
.</td><td></td></tr>
<tr><td></td><td>(OR)</td><td></td></tr>
<tr><td><K2></td><td>4. Consider the two sample transactions T3 and T4:
<table><tr><th>T3</th><th>T4</th></tr><tr><td>read_item(X);
X:=X – N;
write_item(X);
read_item(Y);
Y:=Y + N;
write_item(Y);</td><td>read_item(X);
X:= X + M;
write_item(X)</td></tr></table>
a) Apply basic two-phase locking protocol to the above transactions. Prove that, if every transaction in a schedule follows the two-phase locking protocol, the schedule is guaranteed to be serializable. (5)
b) Now apply strict two-phase locking protocol to the above transactions. Prove that strict two-phase locking guarantees strict schedules. (5)</td><td><CO4></td></tr>
</table>

Prepared By P.Mirunalini B.Senthilkumar	Reviewed By	Approved By
Course Coordinator	PAC Team	HOD