

CS 162 Section 10

1.

For parts (a-d), consider the following schedule of three transactions. Commit abbreviated “com”

Operation	1	2	3	4	5	6	7	8	9	10	11	12
T1:	R(C)		R(A)				W(A)			com		
T2:				W(C)				R(A)	W(B)		com	
T3:		R(A)			R(A)	W(B)						com

Draw the dependency graph for this schedule. Be sure to list the object(s) (A, B, or C) that is (are) the cause of each dependency on each edge.

Is this schedule conflict-serializable? If so, list a serial ordering of the transactions that would produce an equivalent schedule. If not, state why not.

True/False: This schedule of read and write operations could be generated by a system following the regular **2PL** (two phase locking) protocol.

True/False: This schedule of read and write operations could be generated by a system following the **Strict 2PL** protocol. (Circle one)

In general, is Strict 2PL is more likely to encounter deadlocks than regular 2PL?

State **Why** or **Why Not**.

