

Question 4. {12 marks, 6 marks each}

For each of the following schedules determine whether the schedule is conflict-serializable or view-serializable. Justify your answers.

a.

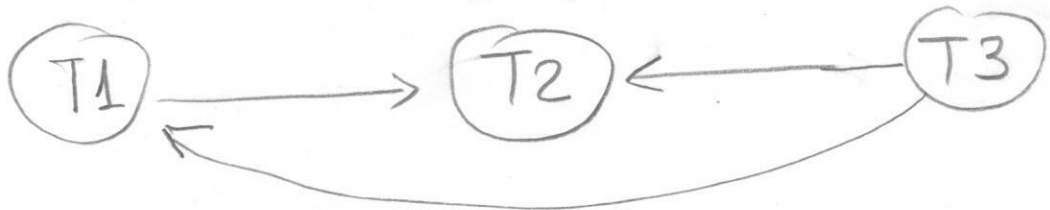
T1	T2	T3
		write(C)
	write(B)	
	read(B)	
		read(C)
write(C)		
	read(C)	

1 Is it conflict-serializable?

Yes

2 WHY?

Precedence graph :



Graph has no cycle.

1 Is it view-serializable?

Yes

2 WHY?

If it is conflict-serializable, it is also view serializable.

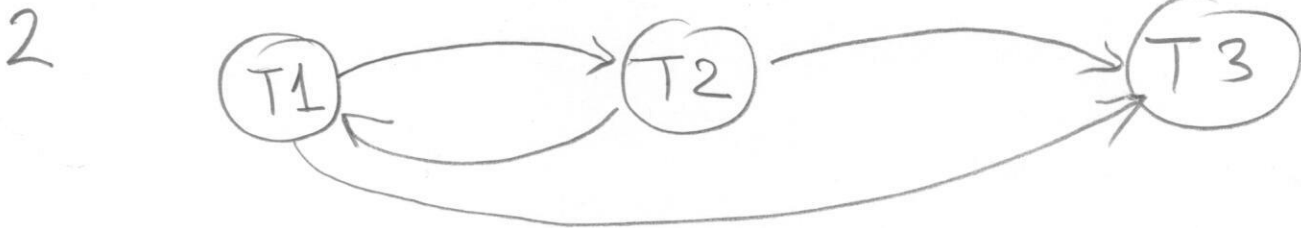
b.

T1	T2	T3
	read(C)	
write(C)	write(C)	
read(B)		write(C)

1 Is it conflict-serializable? NO

WHY?

Precedence graph:



Graph has a cycle

1 Is it view-serializable? Yes

WHY?

2 It is view-equivalent to the serial schedule:

T2, T1, T3

[T2 reads initial C, T1, reads initial B, T3 writes final C, in both schedules]

Question 5. {12 marks, 4 marks each}

Consider the following schedule involving three transactions T1, T2, and T3:

Operation#	T1	T2	T3
1	read(C)		
2			read(B)
3	read(A)		
4		read(C)	
5			write(B)
6	write(A)		
7		read(B)	
8			read(A)
9		write(C)	
10		write(B)	
11	commit		
12		commit	
13			commit

with no lock upgrades

- a. Can this schedule be produced by the **regular two-phase locking** protocol? Justify your answer. (Either show that the protocol is violated or show when the release phase for each transaction starts).

2 • Yes!

• Release phases start at:

T1: after 3 & before 4

T2: after 9

T3: after 6 and before 7

- b. Can this schedule be produced by the **strict two-phase locking** protocol? Justify your answer.

2 • No!

• T1 has to keep the lock for C or A to the end; T2 cannot read/write C.

- c. Is this schedule a **recoverable schedule**? Justify your answer.

2 • No!

• T2 reads B that is produced by T3, but T2 commits before T3 commits.

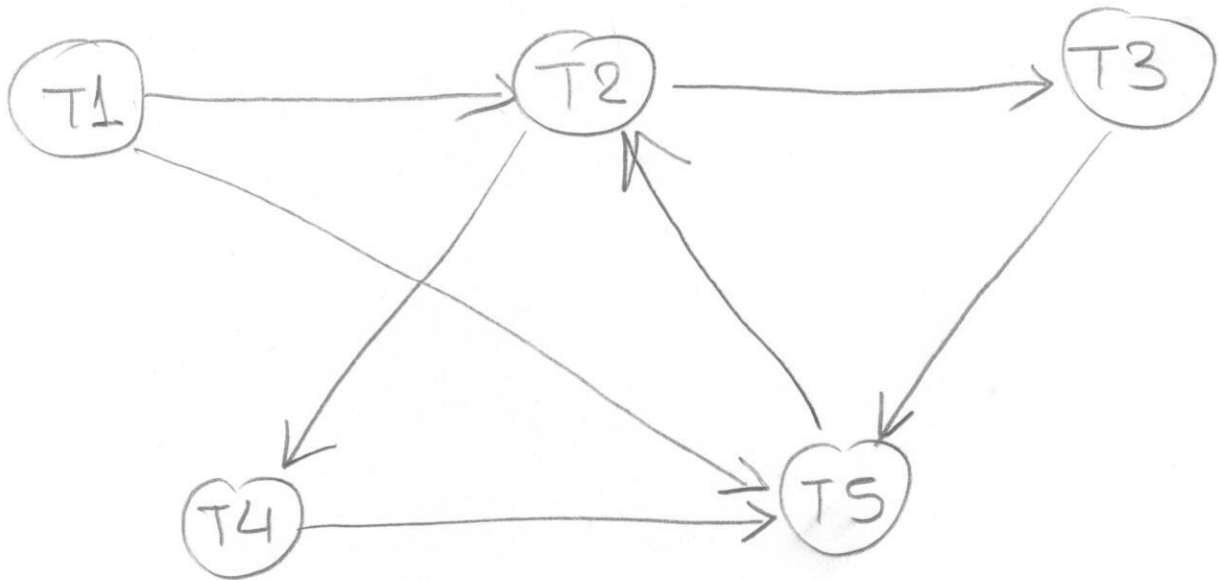
Q.6

{6 marks} Suppose that at some time during the execution of a database system we have the transaction scenario shown on the following table:

Transaction	Data items locked by transaction	Data items transaction is waiting for
T1	B	A, C
T2	C	D, E
T3	E, F	G
T4	D	A
T5	A, G	C

Determine whether a deadlock exists and justify your answer.

• Wait-for graph



• Graph has cycles.

2 Therefore, there is a deadlock.