Student ID: 30613043
Student Name: Grant Fullston
Unit Code: FIT2094
Applied Class No: A09

Comments for your marker:

(a)

TIME	TRANS	ACTION	A	B	C	D
0	**T1**	**UPDATE A**	X(T1)			
1	**T1**	**UPDATE B**		X(T1)		
2	**T2**	**READ C**			S(T2)	
3	**T2**	**READ D**				S(T2)
4	**T3**	**UPDATE A**	T3 wait T1			
5	**T2**	**UPDATE C**			X(T2)	
6	**T1**	**ROLLBACK**	X(T3)			
7	**T3**	**UPDATE C**			T3 wait T2	
8	**T2**	**UPDATE B**		X(T2)		
9	**T2**	**UPDATE A**	T2 wait T3			

- Does a deadlock exist in this transaction sequence? Yes there is a deadlock
- Explain why you came to this conclusion.

At the end of the sequece, we get into a loop where T3 is waiting on the T2 lock on element C, while T2 is waiting for the T3 lock on element A. In the case two transactions are waiting on eachother to update an element and commit or rollover, we get a deadlock, because neither transacton can continue to operate until the lock is released, which will only happen after completion of the transaction, which requires the element currently locked by the other.

Thus neither transaction is able to complete and we have a deadlock

(b)

TRL ID	TRX NUM	PREV PTR	NEXT PTR	OPERATION	TABLE	ROW ID	ATTRIBUTE	BEFORE VALUE	AFTER VALUE
101	601	Null	102	START	****START TRANSACTION				
102	601	101	103	INSERT	PRODUCT	1			ABC, 1205
103	601	102	104	INSERT	PART	1			A, 567
104	601	103	105	INSERT	PART	2			B, 98
105	601	104	106	INSERT	PART	3			C, 549
106	601	105	NULL	COMMIT	****END OF TRANSACTION				

…. add extra rows as needed