Assignment (Transaction Management)

Done By : Mohammed Aadil (IIT2018179)

Q1) Initial values of A and B is zero (given) and consistency requirement is A=0 ∨ B=0

a. Only 2 possible permutations are possible $\mathbf{T_1T_2}$ and $\mathbf{T_2T_1}$:

Case 1 :

	A	**B**
Initial	0	0
After T_1	0	1
After T_2	0	1

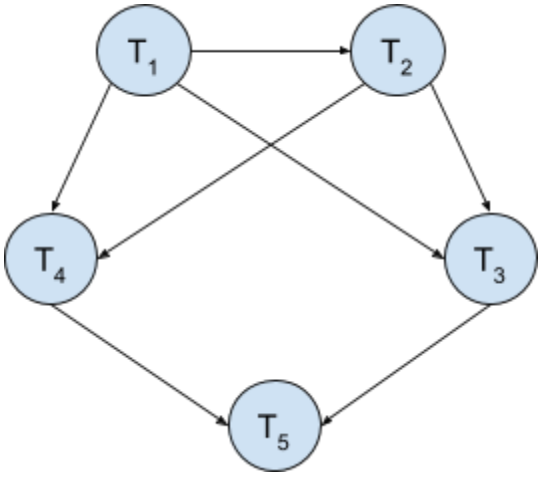
Consistency is met as : 0 v 1 = 1 (True)

Case 2 :

	A	**B**
Initial	0	0
After T_2	1	0
After T_1	1	0

Consistency is met as : 1 v 0 = 1 (True)

b. Procedure Graph :



T_1	T_2
read(A)	
	read(B)
	read(A)
read(B)	
If A = 0 **then** B = B+1	
	If B = 0 **then** A = A+1
	write(A)
write(B)	

c. No serializable schedule is possible even by parallel execution of the transactions. Suppose we start with :

(i) T1 **read**(A) -> Irrespective of the steps of T2 => **B = 1**

(ii) T1 completes -> Then T2 read(B) will give a value of 0 -> T2 completes => **A=1**

Thus B = 1 ∧ A = 1 -> ¬ (A = 0 ∨ B = 0)

Similar result is obtained if we start with T_2 **read**(B).

Q2)

a. Lock and unlock instructions :

T_{31} :
lock-S(A)
read(A)

lock-X(B)
read(B)
If A = 0 **then** B := B + 1
write(B)

unlock(A)
unlock(B)

T_{32} :
lock-S(B)
read(B)

lock-X(A)
read(A)
If B = 0 **then** A := A + 1
write(A)

unlock(B)
unlock(A)

b. Yes, the execution of these transactions result in a deadlock.

T_{31}	T_{32}
lock-S(A)	
	lock-S(B)
	read(B)
read(A)	
lock-X(B)	
	lock-X(A)

Now we are in a deadlock.