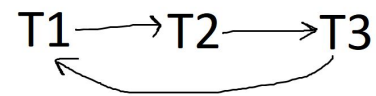


CSCI235-Assignment 2

TASK 2:



- (1) Serialization graph scheduler would abort the transaction (T1) due to the order of conflicting operations as observed in the serialization graph (right). As it is cyclical transactions are aborted by scheduler.

T1	T2	T3
read(x)		
write(x, x+1)		
	read(y)	
	write(x, y+1)	
		read(z)
		write(x, z+1)
read(z)		
write(z, x+2)		
abort //because creates cyclical conflict serializable graph		

- (2) 2PL Scheduler would cause a deadlock as seen below.

T1	T2	T3
Lock (x) read(x)		
write(x, x+ 1)		
	Lock (y) read(y)	
	lock(x) wait	
		lock(z) read(z)
		lock(x) wait
lock(z) wait		

At this point deadlock has occurred because all transactions are waiting for locks to be unlocked before further processing can occur. In a database system **deadlock is eliminated either through wait for graph or through a timeout.**

- (3) Timestamp ordering scheduler:

T1	T2	T3	X	Y	Z
timestamp(t1)					

read(x)			X:T1		
write(x, x+ 1)					
	timestamp(T2)				
	read(y)			Y:T2	
	write(x, y+1)		X:T1:T2		
		timestamp(T3)			
		read(z)			Z:T3
		write(x, z+1)	X:T1:T2:T3		
read(z)					Z:T3:T1
write(z, x+2)					
abort					

Aborted because timestamp order is conflicted on data Z. ($T3 > T1$ is not correct)