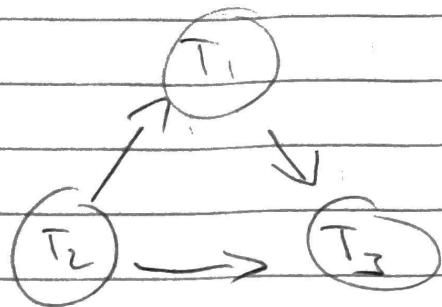


HW7

i.



ii. Conflict Serializable

serial schedules:

~~T₂, T₃, T₁~~
~~T₂, T₁, T₃~~

iii. No, there are several locks that occur during the shrinking phase. ex.

lock in T₂ W(D) and T₃ R(D),

lock in T₂ W(E) and T₃ R(E)

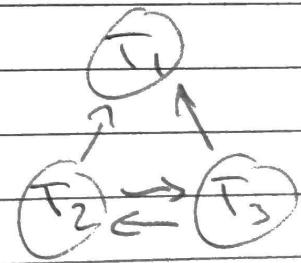
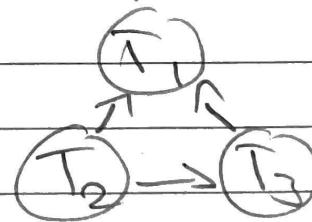
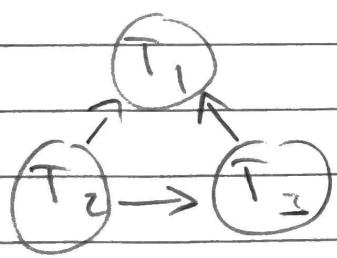
lock in T₁ W(B) and T₃ R(B)

2	T ₁	T ₂	T ₃	
X(X)				
R(X)	S(Y)			not 2PL
R(Y)	S(Y)			
	R(Y)			(schedule on next page)
W(Y)				
W(X)		W(X)		
U(X)				
	R(X)			
	W(X)			

2.

 T_1 T_2 T_3 $X(x)$ $R(x)$ $S(x)$ $R(y)$ $S(y)$ $R(y)$ $W(x)$ $U(x)$ $X(x)$ $W(x)$ $U(x)$ $U(y)$ $X(y)$ $W(y)$ $X(x)$ $R(x)$ $W(x)$ $U(x)$ $U(y)$ $X = \text{exclusive lock}$ $S = \text{shared lock}$ $U = \text{unlock}$

S1: T ₁	T ₂	T ₃	S2: T ₁	T ₂	T ₃	S3: T ₁	T ₂	T ₃
r(A)					r(C)		r(A)	
w(A)				r(A)			r(C)	
	((C))			w(A)			w(A)	
w(B)				w(E)			w(A)	
	w(A)			w(A)		w(B)		
	w(L)		r(A)		w(A)	w(C)		
r(A)			r(B)			r(A)		
((B))			w(B)			r(B)		
w(A)					w(C)	w(A)		
w(B)						w(B)		



i. S1 & S2 are conflict equivalent

$S1: T_1$	T_2	T_3	$S2: T_1$	T_2	T_3	$S3: T_1$	T_2	T_3
$s(A)$					$s(c)$		$s(A)$	
$r(A)$					$r(c)$		$r(A)$	$s(c)$
$x(A)$				$sca)$				(cc)
$w(A)$				$r(A)$				$x(A)$
	$s(c)$			$x(A)$				$w(A)$
	$r(c)$			$w(A)$				$u_x(+)$
	$x(B)$			$x(B)$				$u_s(-)$
	$w(B)$			$w(B)$				$x(B)$
	$u(A)$			$u(A)$				$w(B)$
	$u(A)$			$u_x(A)$				$u_s(A)$
	$u(B)$	$x(A)$		$u_x(B)$	$x(A)$			$u_x(A)$
		$w(A)$			$w(A)$			$u_x(B)$
		$y(C)$			$u_x(A)$			$x(C)$
		$w(C)$			$u_s(C)$			$w(C)$
		$u_s(C)$		$s(A)$				$x(C)$
		$u_x(A)$		$r(A)$				$x(C)$
		$u_x(C)$		$s(B)$				
				$r(B)$				
				$x(A)$				
				$w(A)$				
$s(A)$								
$r(A)$								
$s(B)$								
$x(A)$								
$w(A)$								
$x(B)$								
$w(B)$								
$u_s(A), u_s(B), u_x(A), u_x(B)$				$u_x(B)$				

ii. S1 is in 2PL

iii. S2: T ₁	T ₂	T ₃	S3: T ₁	T ₂	T ₃
		S(C)		S(A)	
		r(C)		r(A)	
S ₂ & S ₃ are now: in 2PL	S(A)			S(C)	
	r(A)			r(C)	
	X(A)			X(A)	
	W(A)			W(A)	
	X(C)			X(C)	
	W(B)			W(C)	
	U _s (A)	V(A)	X(A)	U _s (A), U _s (C)	
	U _x (A)	W(A)	W(A)	U _x (C)	
	U _r (B)	X(C)	X(B)		
		W(C)	W(B)		
		U _x (A)	S(A)	U _s (A)	
		U _s (A)	r(A)	U _x (A)	
		U _x (L)	S(B)	U _x (B)	
			r(B)		
			X(A)		
			W(A)		
			X(B)		
			U(B)		
			U _s (A)		
			U _x (A)		
			U _s (B)		
			U _x (B)		

4.	T_1	T_2
	$X(Y)$	
	$X(X)$	
	$R(Y)$	
	$W(Y)$	
	$U_X(Y)$	$X(Y)$
		$R(Y)$
		$W(Y)$
	$R(X)$	
	$W(X)$	
	$U_X(X)$	$X(X)$
		$R(X)$
		$W(X)$
		$U_X(Y)$
		$U_X(X)$

This schedule can be 2PL because all locks are acquired by T_1 at the start, the lock for data item B is released in between, then data item X is in an exclusive lock acquired later, allowing 2PL.

Deadlock:

T₁

X(X)

T₂

X(Y)

X(Y) ~~E~~

deadlock, wait execute
further

X(X)

R(Y)

W(Y)

R(Y)

W(Y)

R(X)

W(Y)

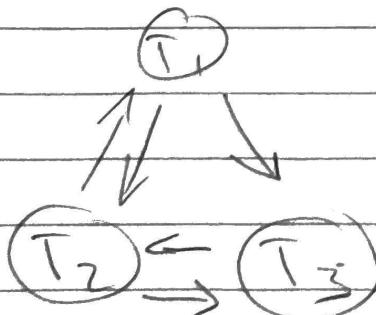
R(X)

W(X)

S.

 T_1 T_2 T_3 $r(A)$ $A = A + S$ $r(A)$ $A = A - 10$ $r(A)$ $A = A + 3$ $w(A)$ $v(A)$

Conflict

 $w(A)$ $r(B)$ $B = B - 10$ $r(B)$ $B = B + 5$ $r(B)$ $B = B - 2$ 

A cycle exists, so the given schedule is not equivalent

6 1.) SI: T_1 | T_2 | T_3 | T_4

$r(Bw\delta)$			
	$r(PBR)$		
		$r(CoC)$	
$w(Bu\delta)$			
	$w(Bw\delta)$		
	$w(PBR)$		
			$r(PBR)$
			$w(PBR)$
$w(Bu\delta)$			
		$w(CoC)$	

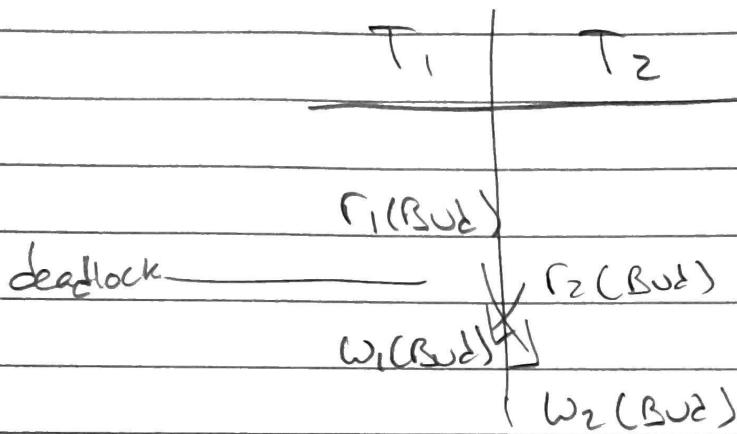
yes,
Conflict Serializable

S2: T_1 | T_2 | T_3 | T_4

$r(Bw\delta)$			
	$r(Bu\delta)$		
$w(Bu\delta)$			
	$w(Bw\delta)$		
	$r(PBR)$		
		$w(CoC)$	
			$r(PBR)$
$w(PBR)$			$w(PBR)$

S2 has cycles \therefore not serializable

2. The loops at $R_1(Bud) \rightarrow W_2(Bud)$ and $R_2(Bud) \rightarrow W_1(Bud)$ create a deadlock, so not granted under 2PL



3. let reading = holding , writing = drinking .
If T_1 holding the Budweiser, then
 T_2 holding the Budweiser again , then
 T_1 drinks some of the Bud , then
 T_2 drinks some of it . Here,
write write or drink drink conflict
occurs . This conflict might not
keep track of inventory properly .