For code 1 $[T_1 \rightarrow 1a \text{ and } T_2 \rightarrow 1b]$

T_1	T_2
R(x)	R(x)
W(x)	W(x)

Conflict Serializable

T_1	T_2
R(x)	
W(x)	
	R(x)
	W(x)

Non - Conflict Serializable

T_1	T_2
R(x)	
	R(x)
W(x)	
	W(x)

Now, trying to solve the conflicts using locks:

T_1	T_2
e.lock(x)	
R(x)	
W(x)	
unlock(x)	
	e.lock(x)
	R(x)
	W(x)
	unlock(x)

T_1	T_2
S.lock(x)	
R(x)	
	S.lock(x)
	R(x)
unlock(x)	
e.lock(x)	
W(x)	
unlock(x)	
	e.lock(x)
	W(x)
	unlock(x)

For code 2 [T₁ for 2a and T₂ for 2b]

T₁	T₂
R(x)	R(x)
W(x)	W(x)
R(y)	R(y)
W(y)	W(y)
	R(z)
	W(z)

Conflict Serializable

T₁	T₂
R(x)	
W(x)	
	R(x)
	W(x)
R(y)	
W(y)	
	R(y)
	W(y)
	R(z)
	W(z)

Non Conflict Serializable

T₁	T₂
R(x)	
W(x)	
	R(x)
R(y)	
	W(x)
	R(y)
W(y)	
	W(y)
	R(z)
	W(z)

Now, trying to solve the conflicts using locks

T₁	T₂
e_lock(x)	
R(x)	
W(x)	
unlock(x)	
	e_lock(x)
	R(x)
	W(x)
	unlock(x)
e_lock(y)	
R(y)	
W(y)	
to unlock(y)	
	e_lock(y)
	R(y)
	W(y)
	unlock(y)
	e_lock(z)
	R(z)
	W(z)
	unlock(z)

T₁	T₂
e_lock(x)	
R(x)	
W(x)	
unlock(x)	
	S_lock(x)
	R(x)
S_lock(y)	
R(y)	
	unlock(x)
	S_lock(x)
	W(x)
	unlock(x)
	S_lock(y)
	R(y)
e_lock(y)	unlock(y)
W(y)	
unlock(y)	
	e_lock(y)
	W(y)
	unlock(y)
	e_lock(z)

T₁	T₂
	R(z)
	W(z)
	unlock(z)