

Part 2: Conflict Serializability (20 points)

Consider the following three transactions and schedule (time goes from top to bottom). Is this schedule conflict-serializable? Show why or why not.

T1	T2	T3
R(A)		
W(A)		
		R(A)
		W(A)
	R(A)	
R(B)		
		R(B)
W(B)		
		W(B)
	R(B)	
	commit	
commit		
		commit

As above, we can arrange the schedules as follows:

$R_1(A); W_1(A); R_3(A); W_3(A); R_2(A); R_1(B); R_3(B); W_1(B); W_3(B); R_2(B)$

There are edges exist in following pair of actions:

$R_1(A)W_3(A)$

$W_1(A)R_3(A)$

$W_1(A)W_3(A)$

$W_1(A)R_2(A)$

$W_3(A)R_2(A)$

$R_1(B)W_3(B)$

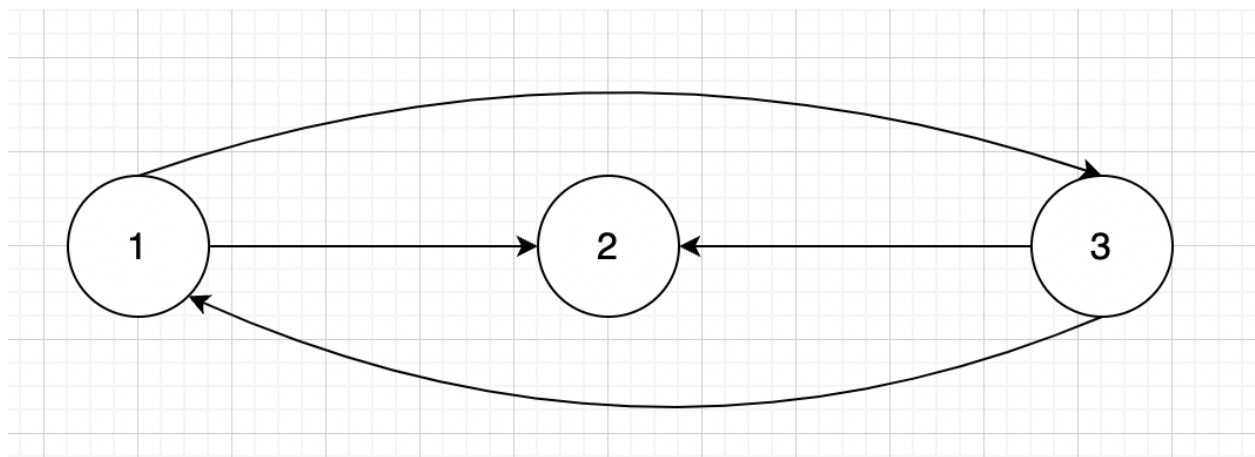
$R_3(B);W_1(B)$

$W_1(B)W_3(B)$

$W_1(B)R_2(B)$

$W_3(B)R_2(B)$

So, we can get edges graph as follows:



No, this schedule is not conflict-serializable since there is a cycle inside the graph. The graph is not acyclic.