

* Serializability

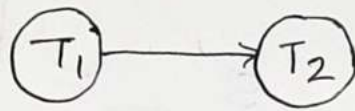
- Schedule - collection of transactions

There are 2 transactions in schedule T_1 & T_2
 and if T_1 has Read(A), write(A) ...
 T_2 has Read(A), write(A)

S

T_1	T_2
Read(A) write(A)	Read(A) write(A)

S is serial schedule.



$T_1 \rightarrow T_2$

fig. Precedence graph

T_1 complete its execution first
 then T_2 complete its execution.

If serial schedule is given, then there is
 no need to serialize the schedule.

S_1

T_1	T_2
Read(A)	Read(A)
write(A)	write(A)

S_1 is parallel schedule.



$T_1 \rightarrow T_2 \rightarrow T_1$

fig. precedence graph for S_1

As S_1 is parallel schedule, then we can
 serialize the S_1 schedule.

There are 2 ways for serialization

①

serializability \rightarrow Conflict View

\rightarrow Find a clone of parallel schedule that clone should be serial schedule.

* Conflict serializability

\rightarrow conflict equivalent

S	
T ₁	T ₂
R(A)	
w(A)	

S'	
T ₁	T ₂
R(A)	
w(A)	
R(B)	

R(A)
w(A)

R(B)

non conflict pair

whether $S \equiv S'$?

R(A) R(A) } Non conflict pair

w(A) R(A)
R(A) w(A)
w(A) w(A) } conflict pair

R(B) R(A)
w(B) R(A)
R(B) w(A)
w(A) w(B) } non conflict pair

we have to first check

S, S' schedules

\rightarrow check for adjacent non conflict pair.

S	
T ₁	T ₂
R(A)	
w(A)	
R(B)	

R(A)
w(A)

S	
T ₁	T ₂
R(A)	
w(A)	
R(B)	R(A)
	w(A)

non conflict pair.

\leftarrow In this way parallel schedule is converted into equivalent serial schedule.

* precedence graph - conflict serializability

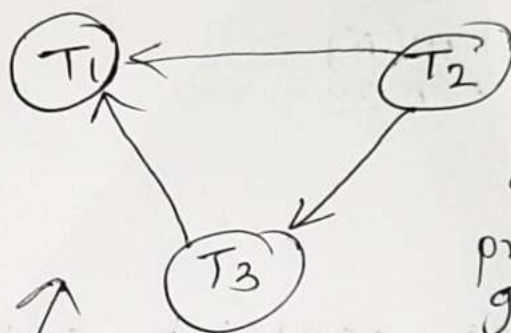
EX.

	S		
	T ₁	T ₂	T ₃
R(x)	R(x)		R(y) R(x)
		R(y) R(z)	w(y)
		w(z)	
R(z) w(x) w(z)			

R(x)
conflict
is
w(x)
check
for

check conflict pairs in
other transactions and
draw edges.

precedence graph.



check in
precedence
graph, ..

whether any
loop or cycle exists

No loop or
No cycle

This means this is conflict serializable
schedule. It is serializable and
consistent schedule.

Refer precedence graph, for finding the
sequence of transactions.

→ check indegree of each vertex.

$$\text{ind}(T_1) = 2$$

$$\text{ind}(T_2) = 0$$

$$\text{ind}(T_3) = 1$$

So, pick T₂ as it has 0
indegree

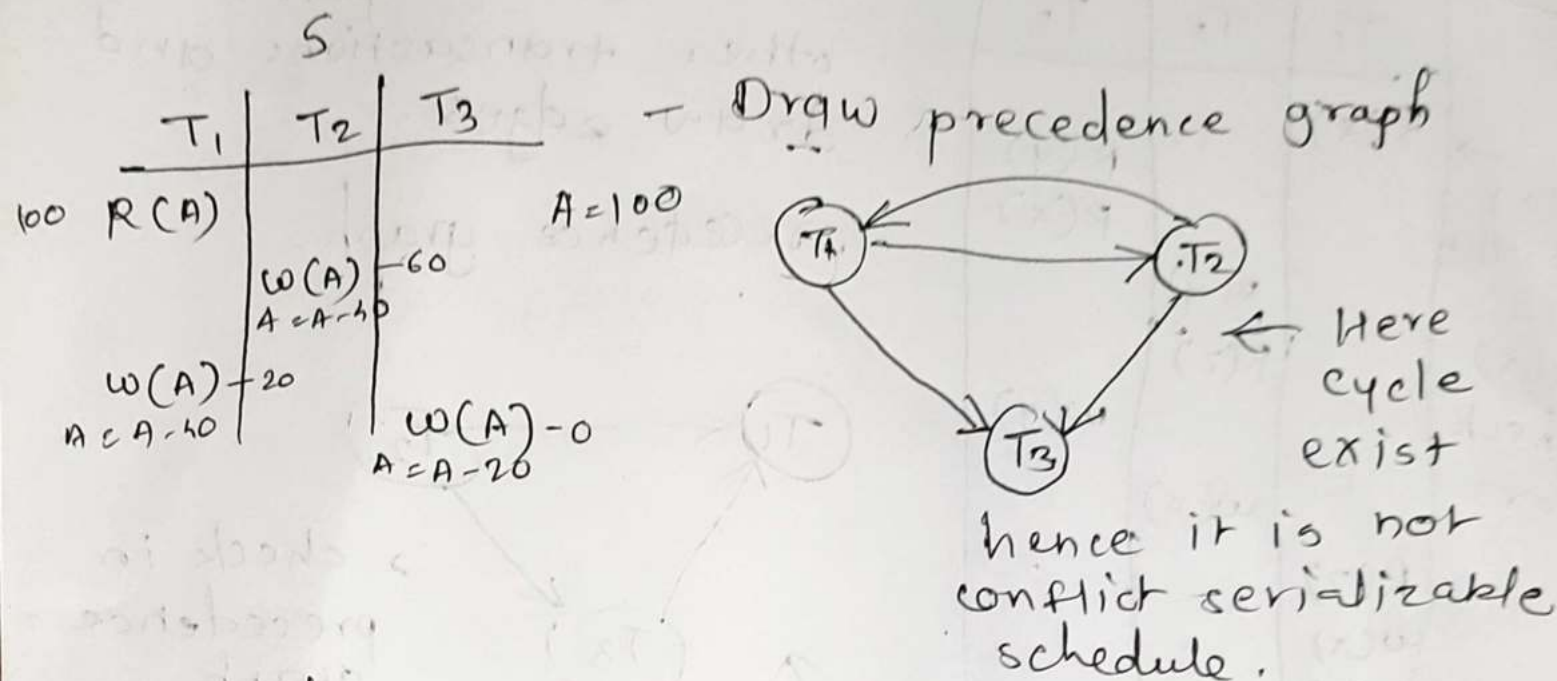
T₂ → T₃ → T₁

This is the sequence of equi serializable
schedule.

Then apply conflict serializability method.

→ check & replace conflict equivalent.

Q. check whether schedule is conflict serializable or not?



so when conflict serializable schedule is not given then we can use view serializable method.

S₁

	T ₁	T ₂	T ₃
100 R(A)			
		W(A) - 60 A = A - 40	
			W(A) - 0 A = A - 20
	W(A) - 20 A = A - 40		

A = 100

for both the S & S₁ schedules, as the operations gives same o/p. so we can say that they are equivalent not equal.

S & S₁ is are not conflict serializable schedule. They are view serializable schedule.

$$\boxed{S \equiv S_1}$$

↑
view equivalent