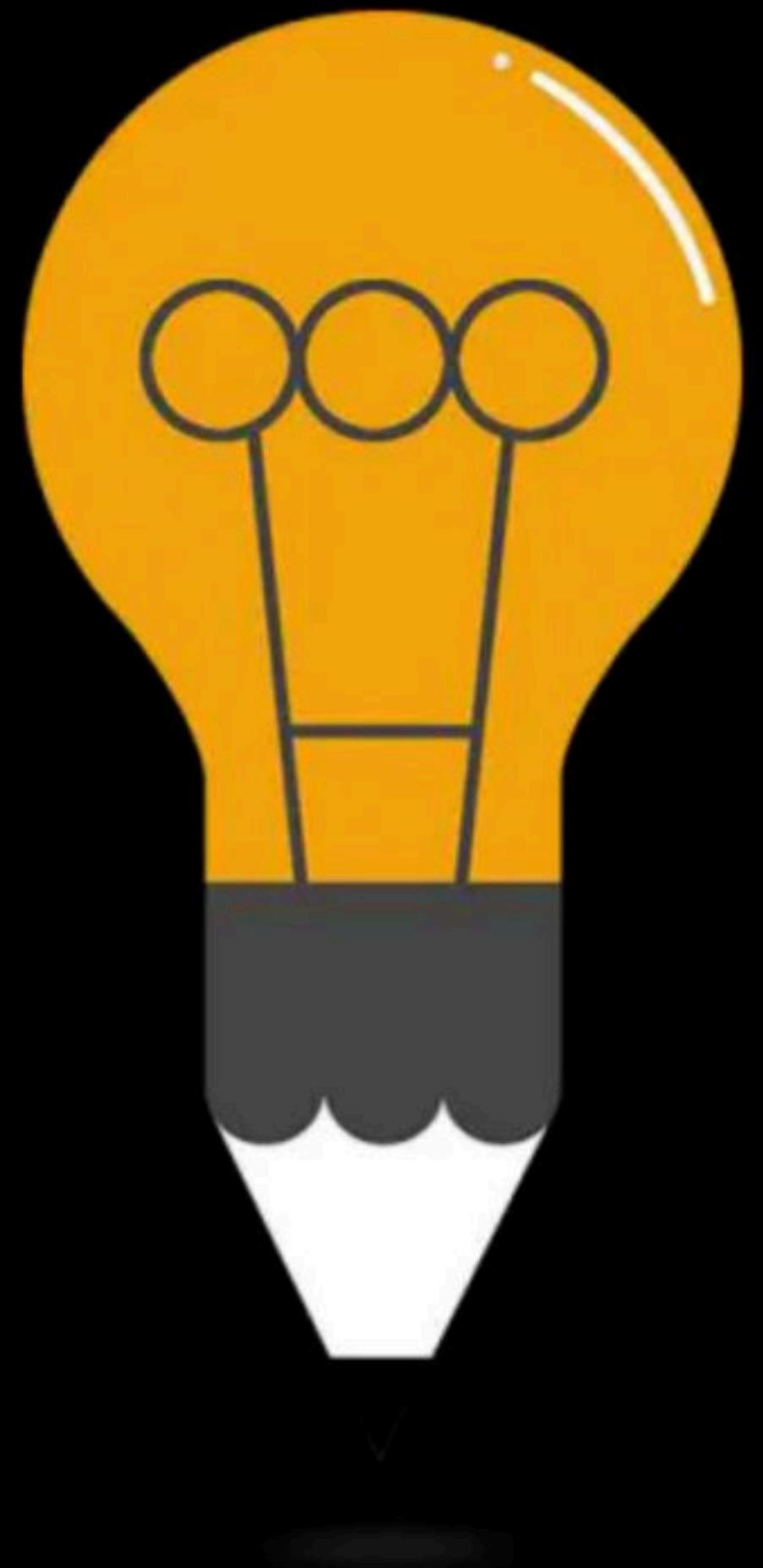




Transaction & Concurrency Control: Part II

Complete Course on Database Management System



DBMS **Serializability: Conflict & View**

By: **Vishvadeep Gothi**

Transaction

Transaction

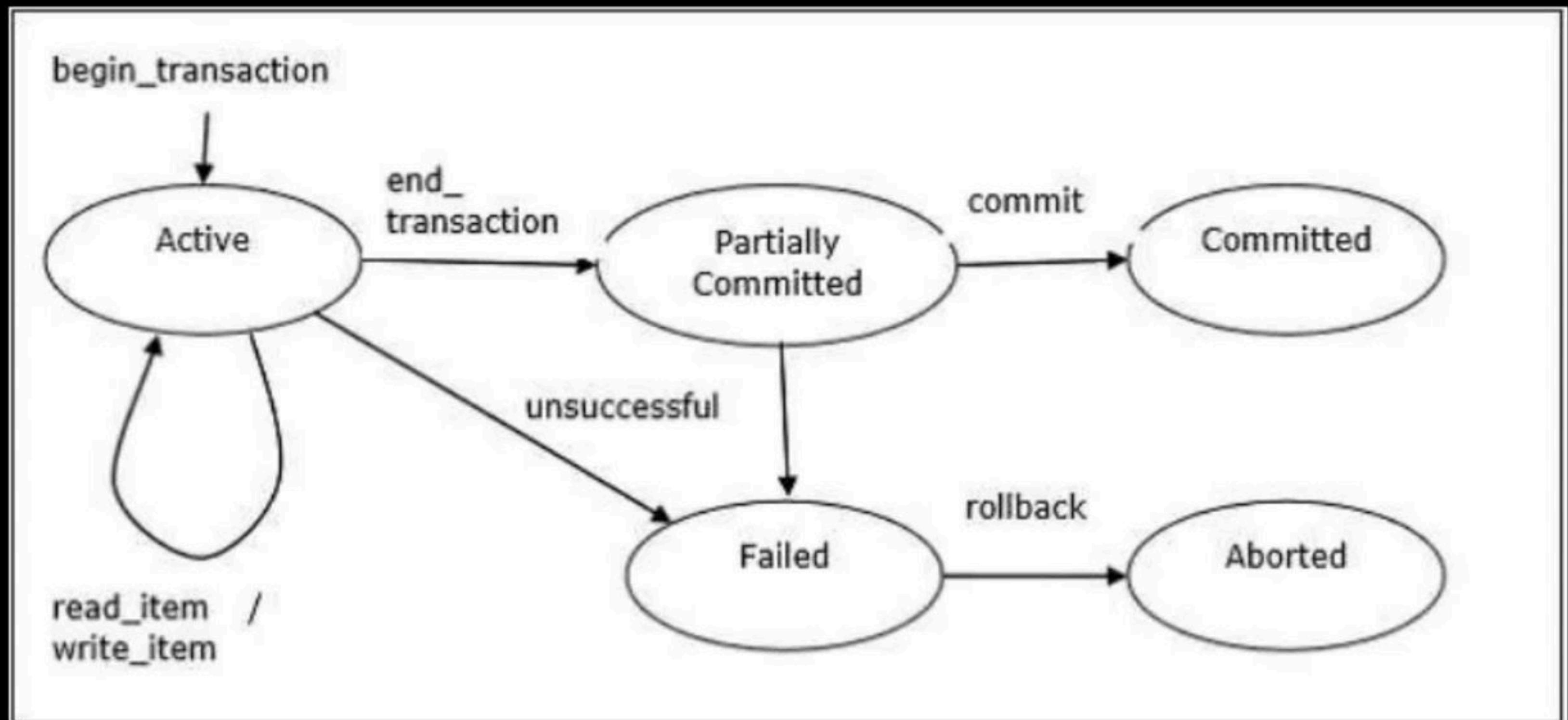
Logic unit of database which includes one or more database access operations

States of Transaction

Commit

Rollback

States of Transaction



ACID Property

Schedule

Concurrency

Why Concurrency

Why Concurrency

- Improved throughput
- Resource utilization
- Reduced waiting time

Problems With Concurrency

- Recoverability problems
- Deadlock
- Serializability Issues

Dirty Read or Temporary Update Problem

T1	T2
R(X)	
X=X+2	
W(X)	
	R(X)
failed	

Phantom Read Problem

T1	T2
R(X)	
	R(X)
Delete(X)	
	R(X)

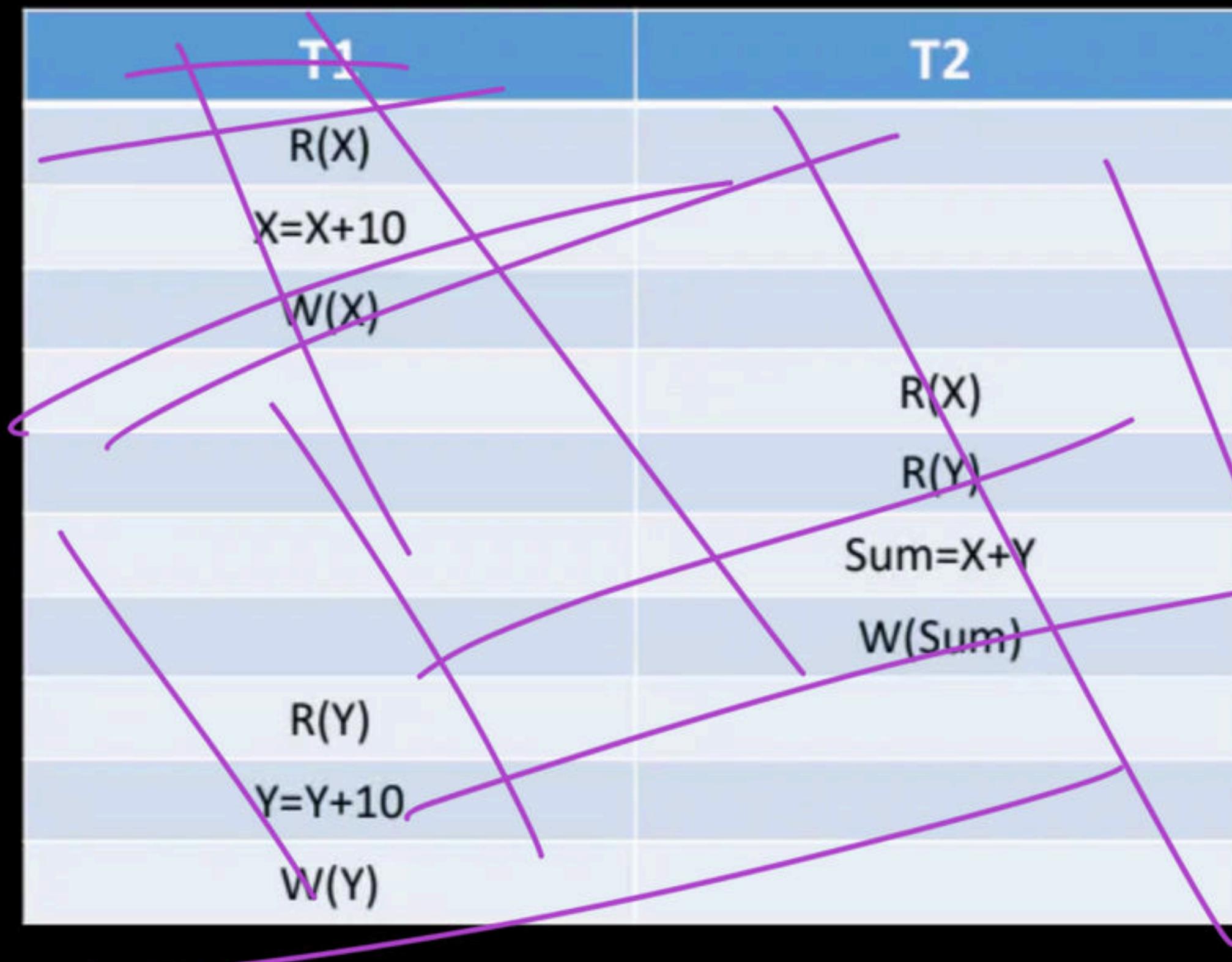
Unrepeatable Read Problem

T1	T2
R(X)	
	R(X)
W(X)	
	R(X)

Lost Update Problem

T1	T2
R(X)	
X=X+2	
W(X)	
	W(X)
	Commit
Commit	

Incorrect Summary Problem



Good vs Bad ~~Transactions~~

final
result
as expected

schedule
final result
not as expected

Serial vs Non-Serial Schedule



concurrent ↗

ex:-

	T_1	T_2
	$R(y)$	
	$y = y + 3$	
	$w(y)$	
		$R(x)$
		$x = x + 2$
		$w(x)$

ex

	T_1	T_2
	$R(y)$	
	$y = y + 3$	
	$w(y)$	
		$R(x)$
		$x = x + 2$
		$w(x)$

ex:-

	T_1	T_2
	$R(y)$	
	$y = y + 3$	
	$w(y)$	
		$R(x)$
		$x = x + 2$
		$w(x)$

Serializable Schedule

A concurrent schedule which can provide final output as a serial schedule.

	T ₁	T ₂
10	R(y)	
13	J = y + 3	
		R(x) 5
w(y)		
	x = x + 2 7	
w(x)		

$$x = \cancel{5} 7$$

$$y = \cancel{10} 13$$

serial schedule

↓

run T₁ then T₂

$$x = \cancel{5} 7$$

$$y = \cancel{10} 13$$

serial schedule:-

run T₂ then T₁

↓

$$x = \cancel{5} 7$$

$$y = \cancel{10} 13$$

Serializable Schedule

	T_1	T_2
$R(x)$		
$x = x + 5$		
$R(y)$		
$y = y + 10$		
$w(x)$		
$R(x)$		
$w(y)$		
$x = x * 2$		
$w(x)$		

$$x = \cancel{10} \cancel{15} 30$$

$$y = \cancel{5} 15$$

serial schedule :-

T_1 then T_2

$$x = \cancel{10} 15 30$$

$$y = \cancel{5} 15$$

serial schedule :-

T_2 then T_1

$$x = \cancel{10} 25$$

$$y = \cancel{5} 15$$

Serializable

Serializability :-

Method to prove that a concurrent schedule is serializable.

- ① Conflict serializability
- ② View —||—

Conflict Serializability

→ find conflicts & then based on it prove that
a given schedule is conflict serializable or not

Conflict

2 Database ^ statements are conflict statements if and only if:
all following conditions are satisfied \Rightarrow

- ① Both statements should be in 2 different transactions
- ② Both statements should access same data item
- ③ one of them should be a write operation.

Conflict Equivalent Schedules

 S_1

T1	T2
R(X)	W(X)
	R(Y)
R(Z)	W(Z)
R(Y)	

 S_2

T1	T2
R(X)	W(X)
R(Z)	
R(Y)	R(Y)
R(Y)	W(Z)



2 schedules having
same conflicts
in same order.

$$S_1 \xrightarrow{\text{conflict}} S_2$$

Conflict Equivalent Schedules

S_1

T1	T2
R(X)	
	R(Y)
	R(Z)
W(Y) ←	
	W(Z) ←
R(Z) ←	

S_2

T1	T2
	R(Y)
R(X)	
W(Y) ←	
	R(Z)
R(Z) ←	
	W(Z) →

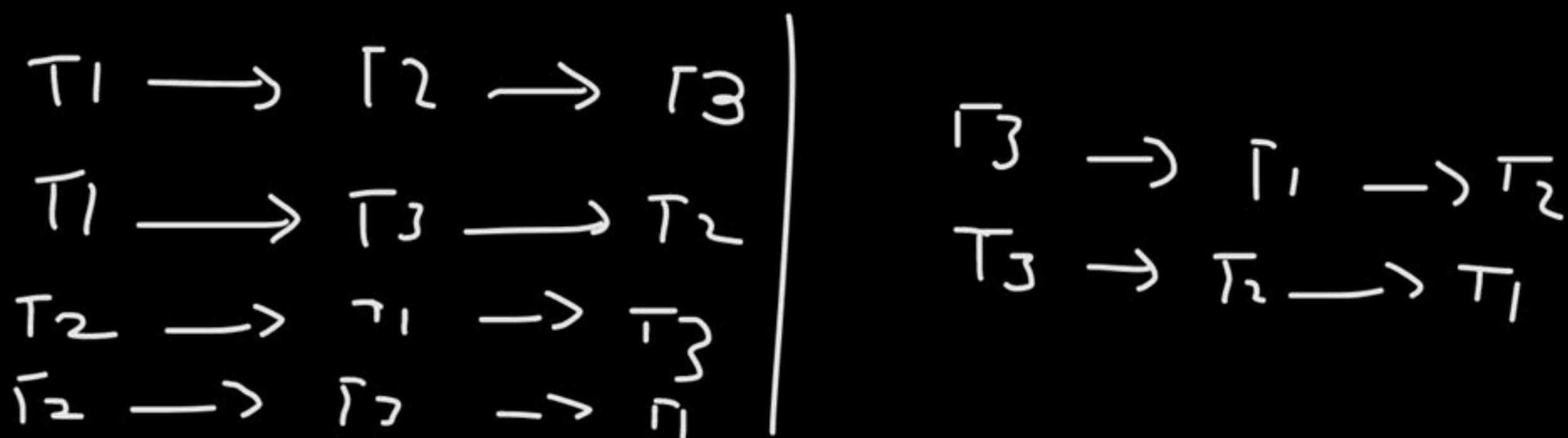
$R(Y)$

$S_1 \cancel{\parallel} S_2$
conflict

Conflict Serializability

Given schedule S is conflict serializable if it is conflict equivalent to S' .

where S' is a serial schedule.

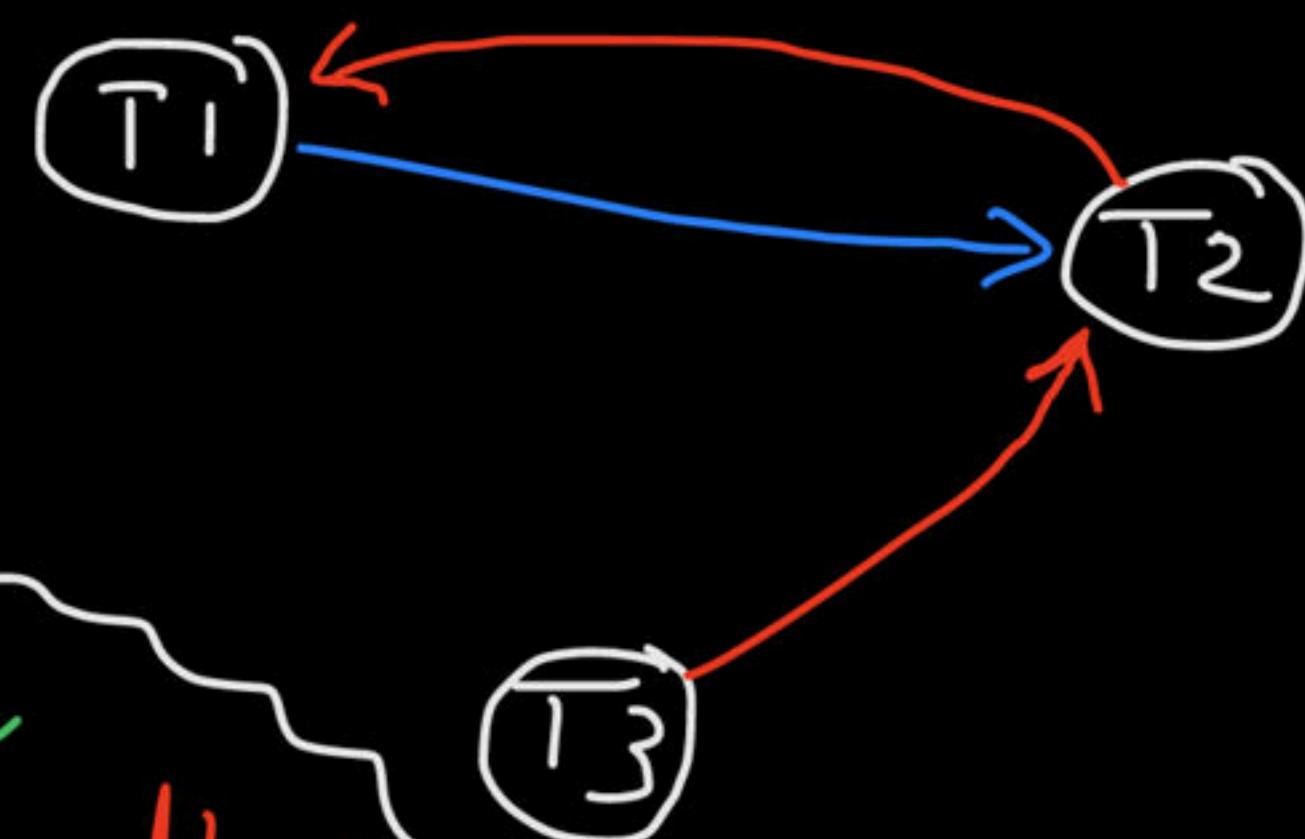


Conflict Serializability

T1	T2	T3
R(X)		
	W(X)	
		R(Y)
		W(Y)
R(Y)		

Precedence graph :- (directed graph)

vertices / nodes \Rightarrow transactions
edges \Rightarrow conflicts



cycle form

$T_1 \rightarrow T_2 \rightarrow T_1$

not conflict serializable

if cycle present
in graph \Rightarrow Not conflict
serializable

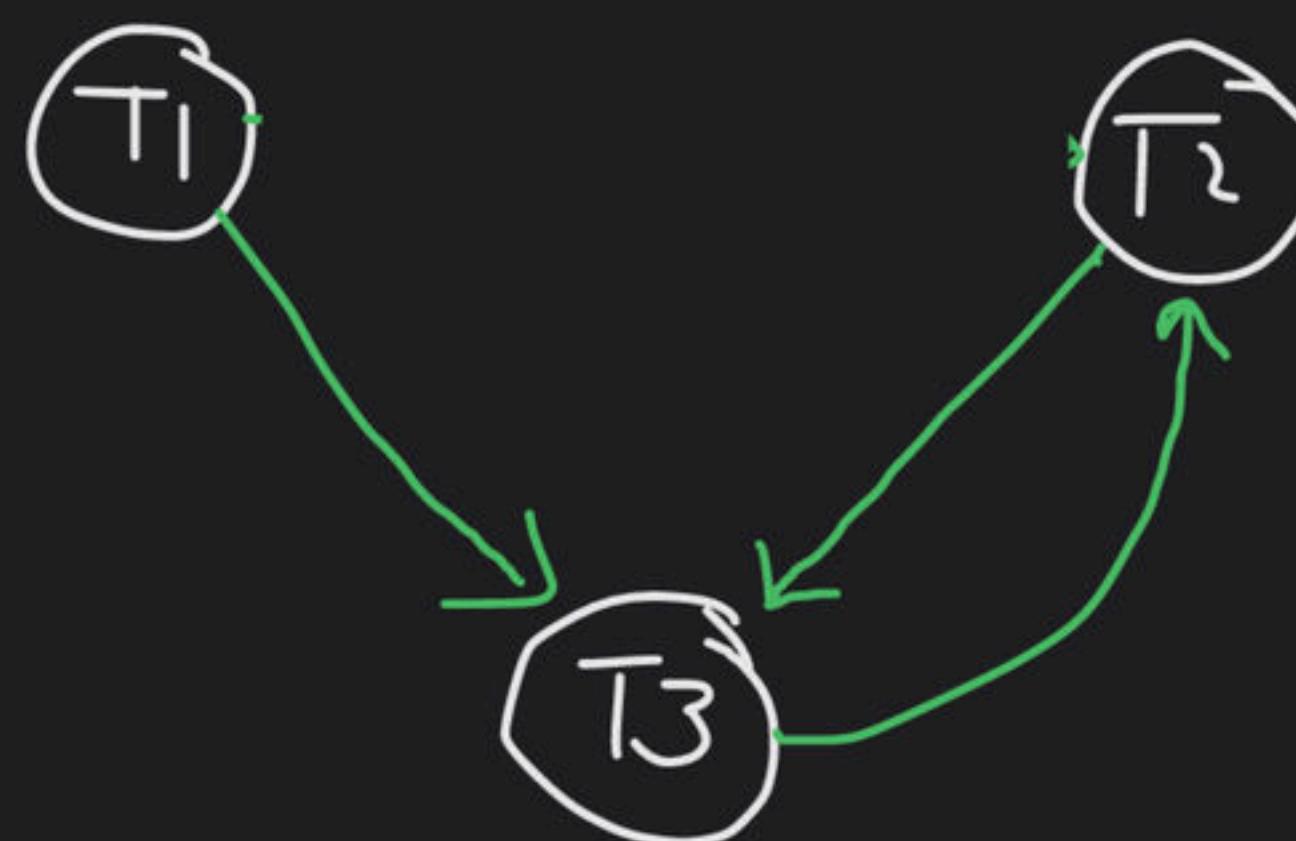
Conflict Serializability



T1	T2	T3
R(X)		
	W(X)	
		R(Y)
	W(Y)	
R(Y)		

	T_1	T_2	T_3
$R(a)$			
$w(b)$			
$R(a)$			
$w(a)$			
$R(q)$			
$w(b)$ <td></td> <td></td> <td></td>			
$R(c)$			
$w(c)$			
$R(c)$			
$R(q)$			

S_1 is conflict serializable or not?

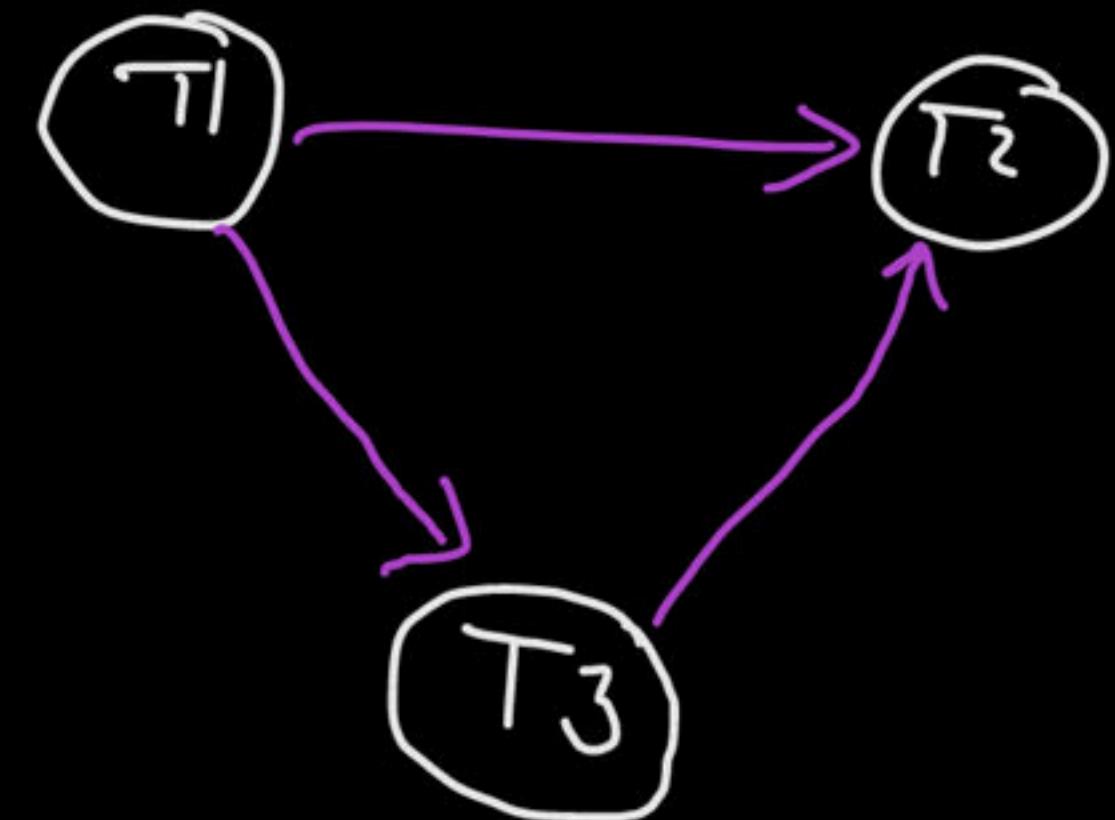


cycle in graph \Rightarrow not conflict serializable

Conflict Serializability

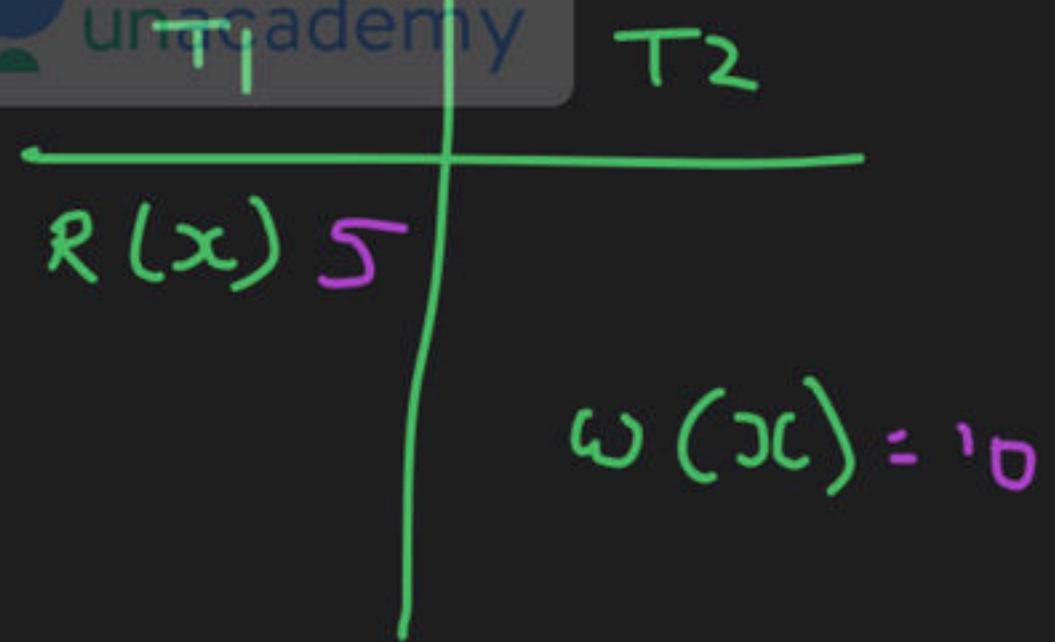
S1

T ₁	T ₂	T ₃
Read(x)		
	Read(Y)	
		Read(Y)
	Write(Y)	
Write(X)		
		Write(X)
	Read(X)	
	Read(X)	

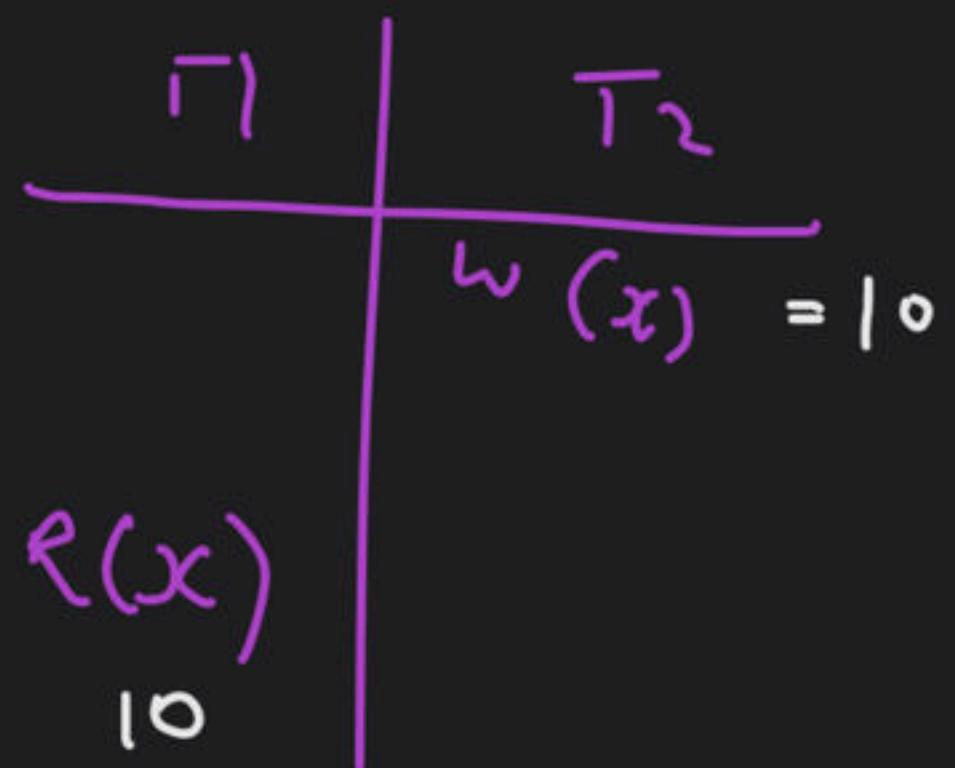


no cycle
 conflict serializable

T₁ → T₃ → T₂



$$x = \not\approx 10$$

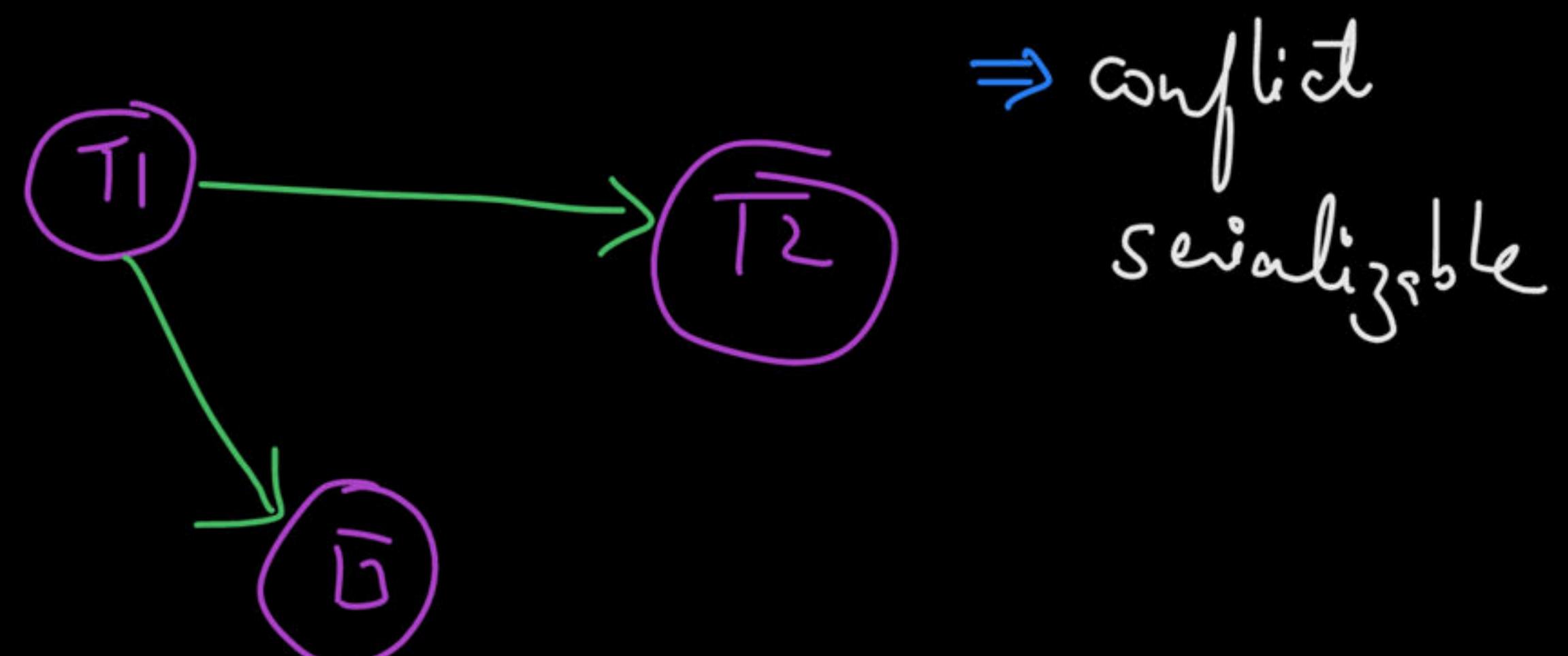


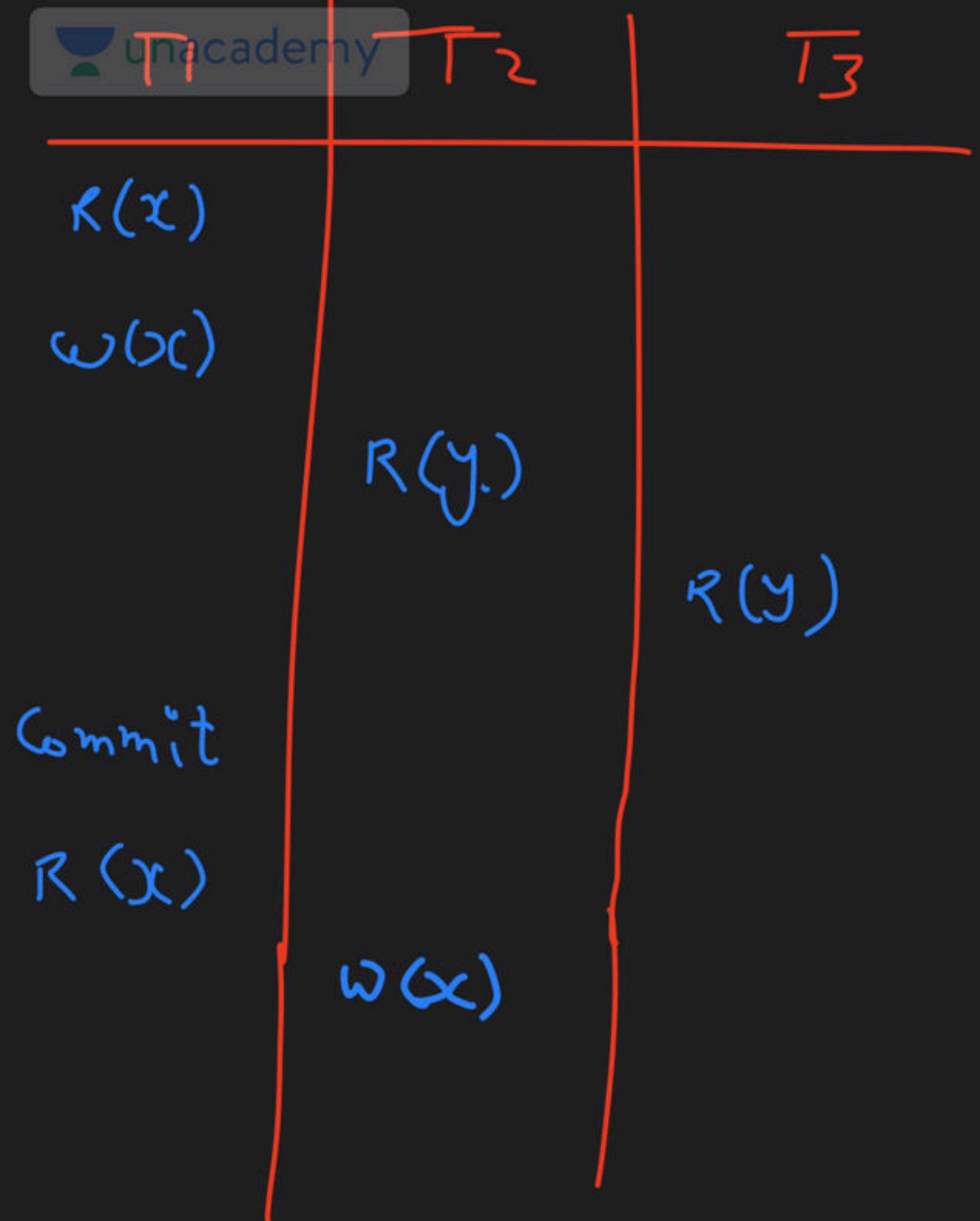
$$x = \not\approx 10$$

Conflict Serializability

T1	T2	T3
R(X)		
	W(X)	
	Commit	
		W(X)
		W(Y)
	Commit	
R(Y)		
Commit		

No checking after commit for conflict





Conflict Serializability

Consider the following schedules for transactions T1, T2, T3 AND T4.

R1A, W1B, R2B, R3C, W1A, R4A, R2C, W4A, W3B, R4B, W4C

The given schedule is serializable or not?

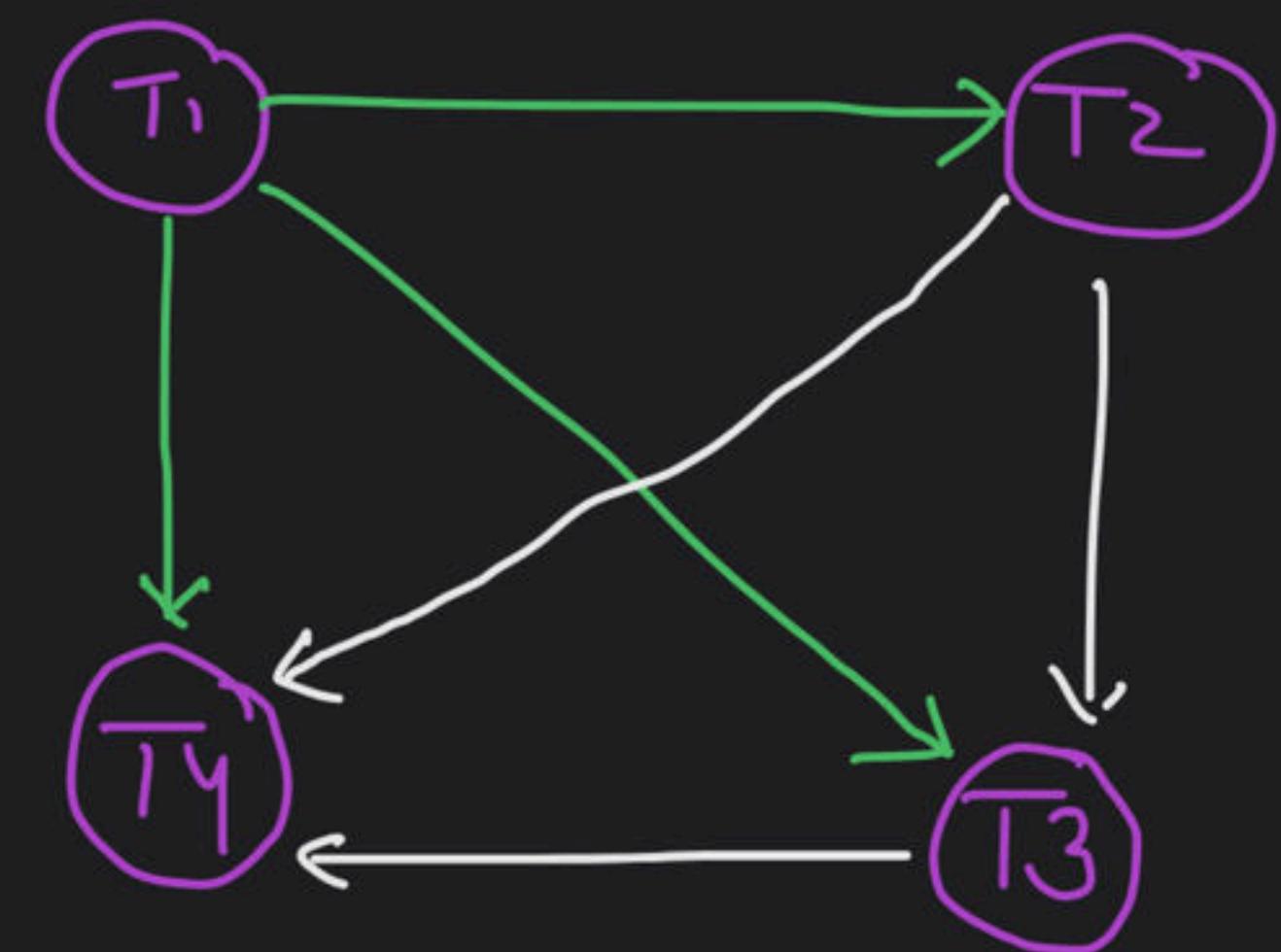
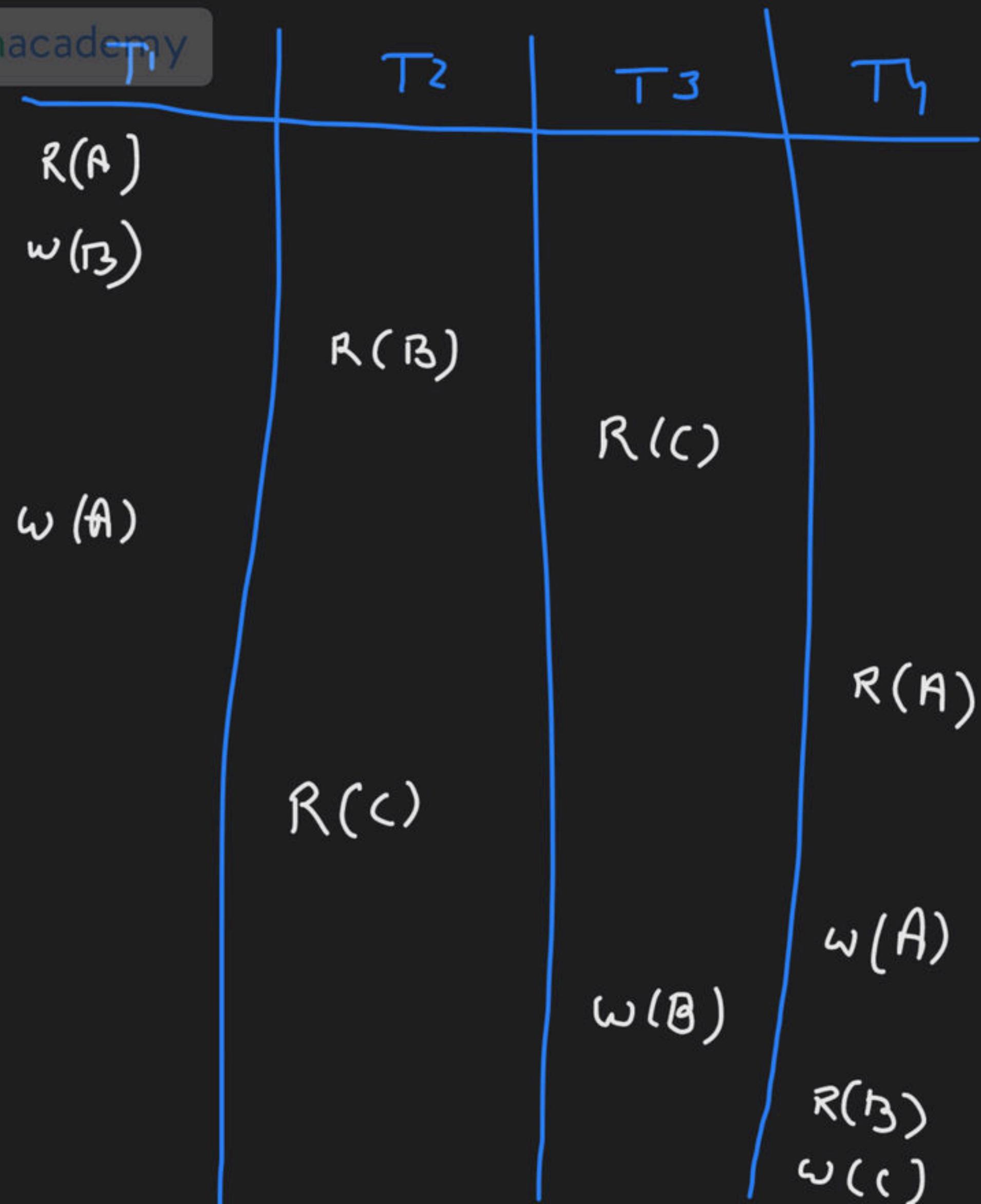
Yes.

Conflict serializable

R, w \Rightarrow Read / write

number \Rightarrow Transaction number

A, B, C \Rightarrow Data item



$\rightarrow T_1 \rightarrow T_2 \rightarrow T_3 \rightarrow T_4$

serializability sequence

View Equivalence

1. Who is Reading first
2. Who is reading from other
3. Who is writing last

View Equivalence

View Equivalence

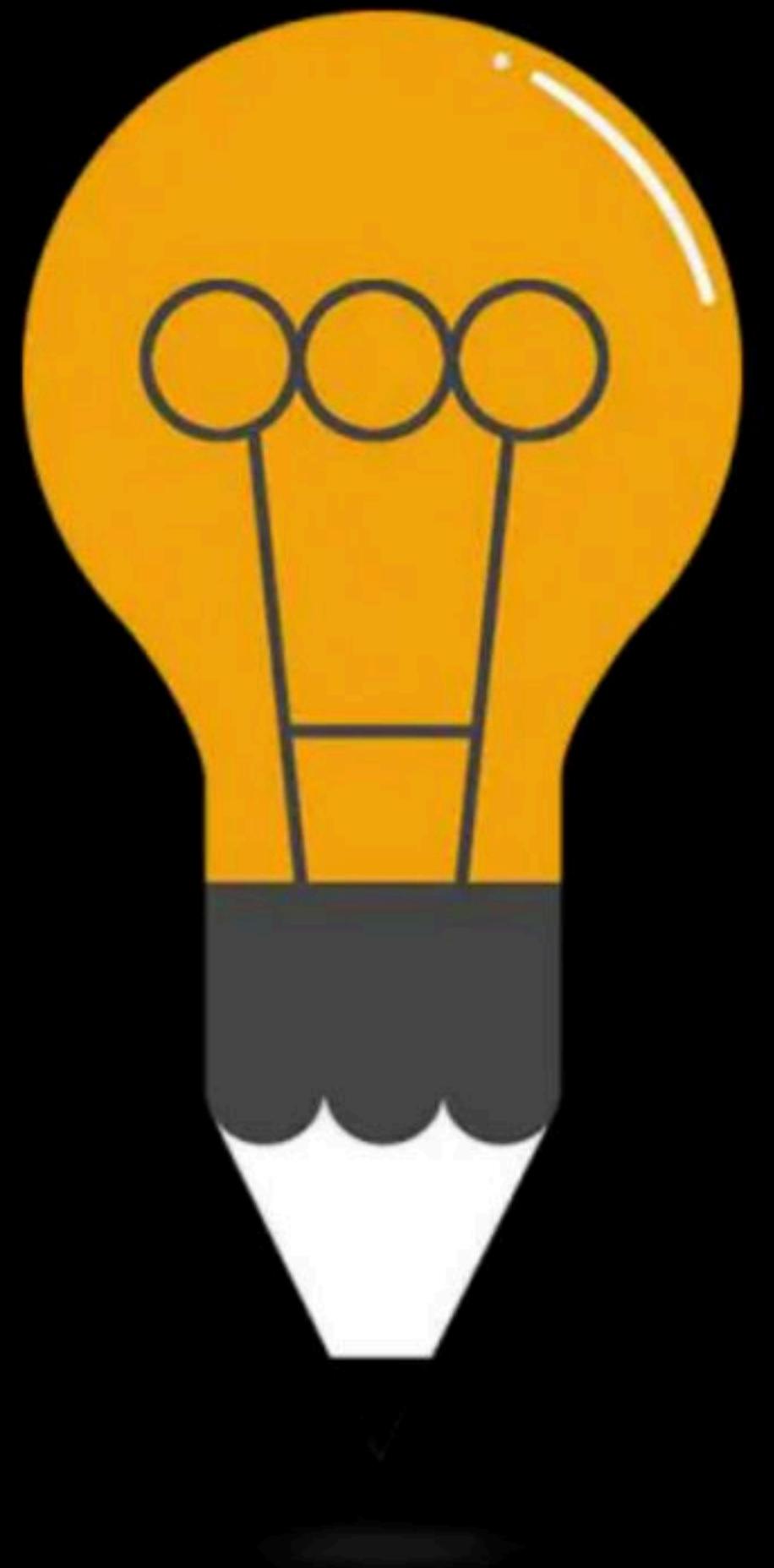
T1	T2	T3
R(X)		
W(X)		
	R(X)	
		W(X)
	W(X)	

T1	T2	T3
R(X)		
	R(X)	
W(X)		
	W(X)	
		W(X)

View Equivalence

T1	T2	T3
R(X)		
	W(X)	
	R(Y)	
		W(Y)
	R(Z)	
W(Z)		
		W(Z)

T1	T2	T3
R(X)		
	W(X)	
	R(Y)	
		R(Z)
W(Z)		
		W(Y)
		W(Z)



DPP Transaction

By: Vishvadeep Gothi

Question 1

The given schedules are conflict serializable or not?

1. 2RA, 2WA, 3RC, 2WB, 3WA, 3WC, 1RA, 1RB, 1WA, 1WB
2. 3RC, 2RA, 2WA, 2WB, 3WA, 1RA, 1RB, 1WA, 1WB, 3WC
3. 2RA, 3RC, 3WA, 2WA, 2WB, 3WC, 1RA, 1RB, 1WA, 1WB
4. 2RX, 3WX, 3Commit, 1WY, 1Commit, 2RY, 2WZ, 2Commit

Question 2 gate-2014

Consider the transactions T_1 , T_2 , and T_3 and the schedules S_1 and S_2 given below.

- $T_1 : r_1(X); r_1(Z); w_1(X); w_1(Z)$
- $T_2 : r_2(Y); r_2(Z); w_2(Z)$
- $T_3 : r_3(Y); r_3(X); w_3(Y)$
- $S_1 : r_1(X); r_3(Y); r_3(X); r_2(Y); r_2(Z); w_3(Y); w_2(Z); r_1(Z); w_1(X); w_1(Z)$
- $S_2 : r_1(X); r_3(Y); r_2(Y); r_3(X); r_1(Z); r_2(Z); w_3(Y); w_1(X); w_2(Z); w_1(Z)$

Which one of the following statements about the schedules is **TRUE**?

- A. Only S_1 is conflict-serializable.
- B. Only S_2 is conflict-serializable.
- C. Both S_1 and S_2 are conflict-serializable.
- D. Neither S_1 nor S_2 is conflict-serializable.

Question 3 gate-2021

Let $r_i(z)$ and $w_i(z)$ denote read and write operations respectively on a data item z by a transaction T_i . Consider the following two schedules.

- $S_1 : r_1(x)r_1(y)r_2(x)r_2(y)w_2(y)w_1(x)$
- $S_2 : r_1(x)r_2(x)r_2(y)w_2(y)r_1(y)w_1(x)$

Which one of the following options is correct?

- A. S_1 is conflict serializable, and S_2 is not conflict serializable
- B. S_1 is not conflict serializable, and S_2 is conflict serializable
- C. Both S_1 and S_2 are conflict serializable
- D. Neither S_1 nor S_2 is conflict serializable

Happy Learning.!

