COMPUTER SCIENCE

Database Management System

Transaction & Concurrency Control

Problem due to concurrent execution

Lecture_5

Vijay Agarwal sir

01 Conflict & View Serializable

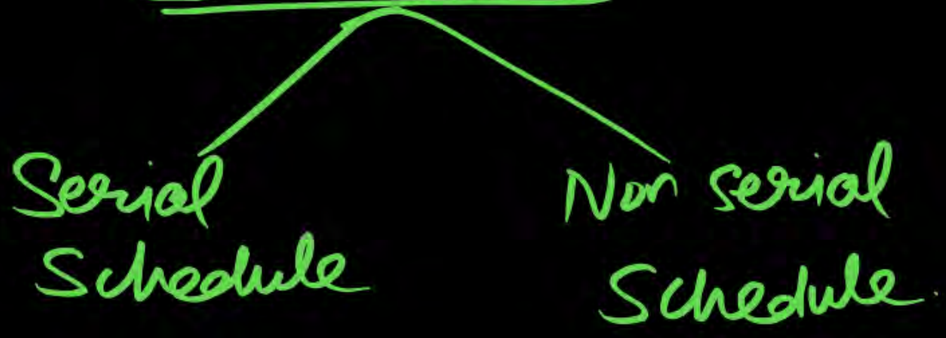
02 Problem due to Concurrent Execution

Transaction Concept

ACID Properties

Transaction State

Schedule



Serial Schedule

⇓

All serial schedule [n!]

are always Consistent.

Non serial schedule

may (or) may Not be Consistent.

⇓

Serializable

Total Number of Concurrent Schedule.

Total # of Serial Schedule.

Total Number of Non Serial Schedule.

Finding Total Number of concurrent Schedule

T_1	T_2
$R_1(A)$ $W_1(A)$	$R_2(B)$ $W_2(B)$

T_1	T_2
L_1 L_2	L_3 L_4

T_1	T_2
0 0	1 1

$L_1L_2L_3L_4$

$L_3L_4L_1L_2$

$L_1L_3L_2L_4$ (or) $L_1L_3L_4L_2$

$L_3L_1L_4L_2$ (or) $L_3L_1L_2L_4$

T_1	T_2
R(A) W(A)	R(B) W(B)

$S_1<T_1T_2>$

(1)

T_1	T_2
R(A) W(A)	R(B) W(B)

$S_2<T_2T_1>$

(2)

T_1	T_2
R(A) W(A)	R(B) W(B)

(3)

T_1	T_2
R(A) W(A)	R(B) W(B)

(4)

T_1	T_2
R(A) W(A)	R(B) W(B)

(5)

T_1	T_2
R(A) W(A)	R(B) W(B)

(6)

Total # Concurrent Schedule $= \frac{(n_1+n_2)!}{(n_1)!(n_2)!}$

$= \frac{(2+2)!}{(2)!(2)!} = \frac{4\times3\times2}{2\times2} = 6$

$T_1 \rightarrow n_1$ operation
2 operation

$T_2 \rightarrow n_2$ operation
2 operation

Total Concurrent = 6

2 Transaction
2! + 2 Serial Schedule.

Total non serial Schedule = Total Concurrent - Serial schedule(m!)

m: # of transaction

= 6 - 2

Serial = 2

Total non Serial = 4

NOTE:

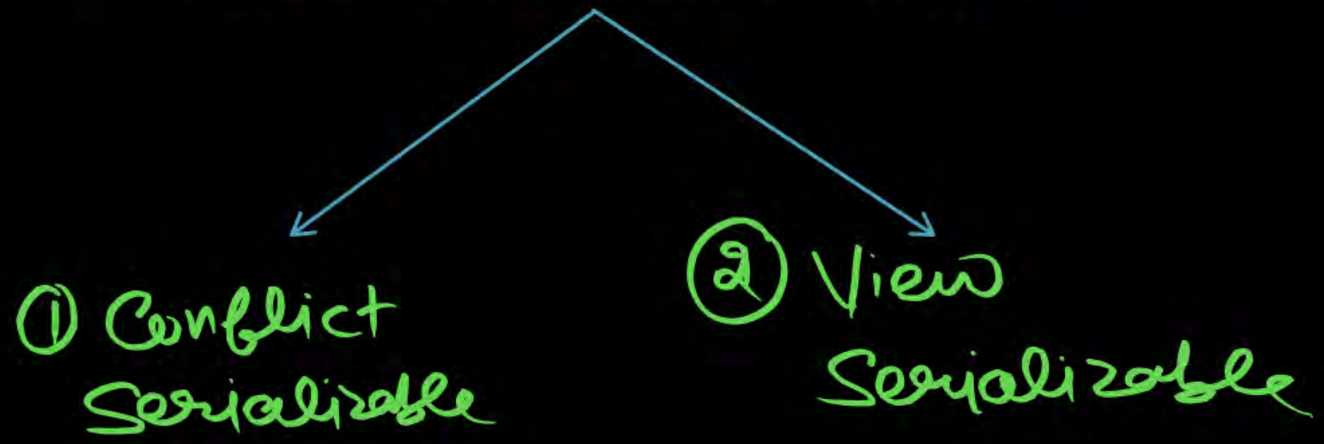
The Number of Concurrent schedule that can be formed Over 'm' transaction having n_1 n_2 n_3 ....n_m operation respectively

$$\text{Total \# of Concurrent Schedule} = \frac{(n_1+n_2+n_3+\cdots n_m)!}{(n_1!)(n_2!)(n_2!)\ldots(n_m!)}$$

$$\text{Total \# of Non Serial Schedule} = \frac{(n_1+n_2+n_3+\cdots n_m)!}{(n_1!)(n_2!)(n_2!)\ldots(n_m!)} - m!$$

$m!$ → Serial Schedule.

serializable Schedule



conflict serializable

① Concept Approach.

② Testing for Conflict Serializability

↳ Precedence Graph Method.

[CNC] ⇒ Cycle Not Conflict

③ Conflict Pair.

Serializability Order Concept.

if Schedule is Conflict Equivalance to a Any serial schedule then S is Conflict Serializable.

∴

MCQ

Suppose a database schedule S involves transaction T_1....., T_n. Construct the precedence graph of S with vertices representing the transactions and edges representing the conflicts. If S is serializable, which one of the following orderings of the vertices of the precedence graph is guaranteed to yield a serial schedule?

[GATE-2016-CS: 2M]

A Topological order

B Depth-first order

C Breadth-first order

D Ascending order of transaction indices

Ans (A).

MCQ

Which one of the following is NOT a part of the ACID properties of database transactions?

[GATE-2016-CS: 1M]

A Atomicity

B Consistency

C Isolation

D Deadlock-freedom

Ans (D).

Conflict Equivalent Schedule

Two schedule are said to be conflict equivalent, if all conflicting operations in both the schedules must be executed in the same order.

Q.

S_1: $R_1(x)$ $W_1(x)$ $R_2(y)$ $W_2(y)$ $R_1(y)$

S_2: $R_1(x)$ $W_1(x)$ $R_1(y)$ $R_2(y)$ $W_2(y)$

S_1

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

Conflict operation

$W_2(y) - R_1(y)$

OR

$W(y) - R(y)$: $T_2 \rightarrow T_1$

S_2

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

Conflict operation

$R_1(y) - W_2(y)$

OR

$R(y) - W(y)$: $T_1 \rightarrow T_2$

S_1 is Not Conflict to S_2.

Q.

S_1: $R_1(A)$ $W_1(A)$ $R_2(A)$ $W_2(A)$ $R_1(B)$ $W_1(B)$

S_2: $R_1(A)$ $W_1(A)$ $R_2(A)$ $R_1(B)$ $W_2(A)$ $W_1(B)$ (S2)

T_1	T_2
R(A)	
W(A)	
	R(A)
	W(A)
R(B)	
W(B)	

S_1

S_1: Conflict operation

$R_1(A) - W_2(A)$

$W_1(A) - R_2(A)$

$W_1(A) - W_2(A)$

OR

$R(A) - W(A)$: $T_1 \rightarrow T_2$

$W(A) - R(A)$: $T_1 \rightarrow T_2$

$W(A) - W(A)$: $T_1 \rightarrow T_2$

T_1	T_2
R(A)	
W(A)	
	R(A)
R(B)	
	W(A)
W(B)	

S_2

$R_1(A) - W_2(A)$

$W_1(A) - R_2(A)$

$W_1(A) - W_2(A)$

OR

$R(A) - W(A)$: $T_1 \rightarrow T_2$

$W(A) - R(A)$: $T_1 \rightarrow T_2$

$W(A) - W(A)$: $T_1 \rightarrow T_2$.

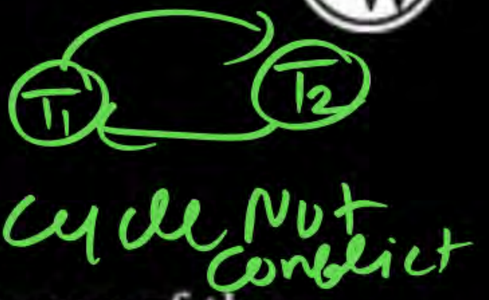
All Conflict operation in both S_1 S_2 are executed in the same order

So S_1 is Conflict equivalent to S_2.

Q.1

Consider a schedule of transactions T_1 and T_2:

T_1	RA			RC		WD		WB	Commit	
T_2		RB	WB		RD		WC			Commit



Here, RX stands for "Read(X)" and WX stands for "Write(X)". Which one of the following schedules is conflict equivalent to the above schedule?

[2020: 2 Marks]

A

T_1					RA	RC	WD	WB	Commit	
T_2	RB	WB	RD	WC						Commit

B

T_1				RA	RC	WD	WB		Commit	
T_2	RB	WB	RD					WC		Commit

C

T_1	RA	RC	WD	WB					Commit	
T_2					RB	WB	RD	WC		Commit

D

T_1	RA	RC	WD	WB					Commit	
T_2					RB	WB	RD	WC		Commit

Q.1

Consider a schedule of transactions T_1 and T_2:

T_1	RA			RC		WD		WB	Commit	
T_2		RB	WB		RD		WC			Commit

B.

Here, RX stands for "Read(X)" and WX stands for "Write(X)". Which one of the following schedules is conflict equivalent to the above schedule?

[2020: 2 Marks]

(A)

T_1					RA	RC	WD	WB	Commit	
T_2	RB	WB	RD	WC						Commit

(B)

T_1				RA	RC	WD	WB		Commit	
T_2	RB	WB	RD					WC		Commit

(C)

T_1	RA	RC	WD	WB					Commit	
T_2					RB	WB	RD	WC		Commit

(D)

T_1	RA	RC	WD	WB					Commit	
T_2					RB	WB	RD	WC		Commit

(a) $W_1(B) - R_2(B)$
$W_1(B) - W_2(B)$

Consider a schedule of transactions T_1 and T_2:

T_1	RA			RC		WD		WB	Commit	
T_2		RB	WB		RD		WC			Commit

Given Question

Conflict operations of Given Question

- ✓ $R_2(B) - W_1(B)$
- ✓ $W_2(B) - W_1(B)$
- → $R_1(C) - W_2(C)$
- ✓ $R_2(D) - W_1(D)$

$R(B) - W(B)$: $T_2 \rightarrow T_1$

OR

$W(B) - W(B)$: $T_2 \rightarrow T_1$

$R(C) - W(C)$: $T_1 \rightarrow T_2$

$R(D) - W(D)$: $T_2 \rightarrow T_1$

T_1					RA	RC	WD	WB	Commit	
T_2	RB	WB	RD	WC						Commit

$R_2(B) - W_1(B)$

$W_2(B) - W_1(B)$

$\Rightarrow$ $W_2(C) - R_1(C)$

$R_2(D) - W_1(D)$

OR

$R(B) - W(B)$: $T_2 \rightarrow T_1$

$W(B) - W(B)$: $T_2 \rightarrow T_1$

$W(C) - R(C)$: $T_2 \rightarrow T_1$

$R(D) - W(D)$: $T_2 \rightarrow T_1$

B

T_1				RA	RC	WD	WB		Commit	
T_2	RB	WB	RD					WC		Commit

$R_2(B) - W_1(B)$

$W_2(B) - W_1(B)$

$R_1(C) - W_2(C)$

$R_2(D) - W_1(D)$

OR

$R(B) - W(B)$: $T_2 \rightarrow T_1$

$W(B) - W(B)$: $T_2 \rightarrow T_1$

$R(C) - W(C)$: $T_1 \rightarrow T_2$

$R(D) - W(D)$: $T_2 \rightarrow T_1$

C

T_1	RA	RC	WD	WB					Commit	
T_2					RB	WB	RD	WC		Commit

✗ $W_1(B) - R_2(B)$

✗ $W_1(B) - W_2(B)$

✗ $W(B) - R(B) : T_1 \to T_2$ ✗

✗ $W(B) - W(B) : T_1 \to T_2$ ✗

✓ $R_1(C) - W_2(C)$

$R(C) - W(C) : T_1 \to T_2$

✗ $W_1(D) - R_2(D)$

$W(D) - R(D) : T_1 \to T_2$

Q.

S_1: $R_1(A)$ $W_1(A)$ $R_1(B)$ $W_1(B)$ $R_2(A)$ $W_2(A)$ $R_2(B)$ $W_2(B)$

S_2: $R_1(A)$ $W_1(A)$ $R_2(A)$ $W_2(A)$ $R_1(B)$ $W_1(B)$ $R_2(B)$ $W_2(B)$

T_1	T_2
R(A)	
W(A)	...
R(B)	
W(B)	
	R(A)
	W(A)
	R(B)
	W(B)

Some of

$R_1(A) - W_2(A)$

$W_1(A) - R_2(A)$

$W_1(A) - W_2(A)$

$R_1(B) - W_2(B)$

$W_1(B) - R_2(B)$

$W_1(B) - W_2(B)$

Serial Schedule $\langle T_1\ T_2 \rangle$

T_1	T_2
R(A)	
W(A)	
	R(A)
	W(A)
R(B)	
W(B)	
	R(B)
	W(B)

Conflict Operation

$R_1(A) - W_2(A)$

$W_1(A) - R_2(A)$

$W_1(A) - W_2(A)$

$R_1(B) - W_2(B)$

$W_1(B) - R_2(B)$

$W_1(B) - W_2(B)$

Conflict Serializable

A schedule is said to be conflict serializable if it is conflict equivalent to a Any serial schedule.

Same conflicting operation order in C_1 & S_1

$\therefore$ Its $\{C_1\}$ conflict is conflict serializable.

$C_1 \equiv S_1$

↑ Conflict Equivalent

T_1	T_2
read(A)	
write(A)	
	read(A)
	write(A)
read(B)	
write(B)	
	read(B)
	write(B)

C_L

T_1	T_2
read(A)	
write(A)	
read(B)	
write(B)	
	read(A)
	write(A)
	read(B)
	write(B)

S_L

Conflict Equivalent [S_1 & S_2]

Note

Two Schedule S_1 & S_2 are said to be Conflict Equivalent If All Conflict operation Must be executed in the same order

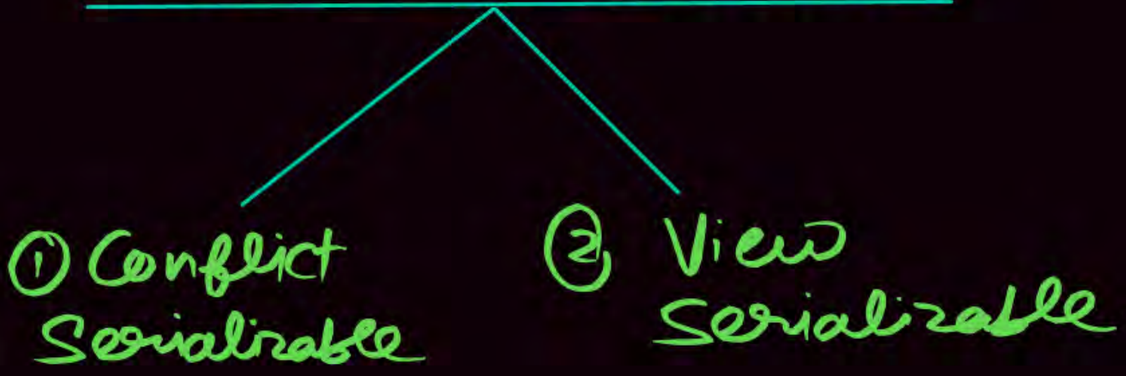
Note

If Schedule is Conflict Equivalent to Any of the Serial Schedule (of the Given Question) then Schedule is Conflict Serializable.

Conflict serializable

1. By Concept
2. By Testing Method
   - Precedenc graph method
3. Conflict Pair
4. Conflict Equivalent to Any serial Schedule

serializable schedule



A Schedule is Serializable either if it is Conflict Serializable (or) View Serializable (or) Both.

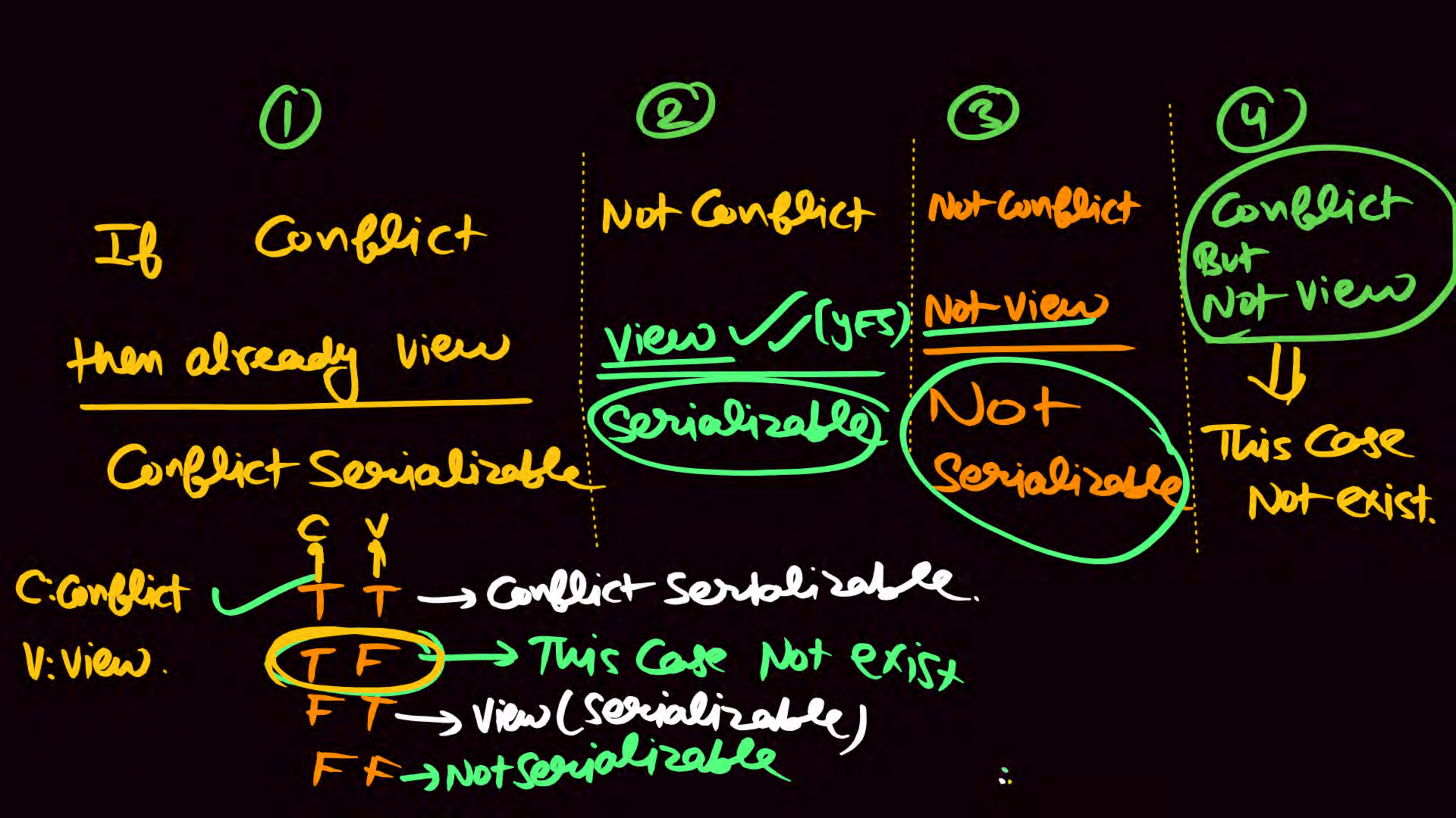
Note

If Schedule is Conflict Serializable then already its View Serializable also but Vice-versa Not True.

Note If Schedule is Not Conflict Serializable then it May (or) May Not View Serializable.

Note If Schedule is View Serializable then Schedule is Serializable.

Note If Schedule is Not View Serializable then Schedule is Not Serializable.


①
②
③
④
If Conflict
then already View
Conflict Serializable
Not Conflict
View ✓ (VFS)
Serializable
Not Conflict
Not View
Not Serializable
Conflict But Not View
This Case Not exist.
C: Conflict
V: View.
C V
T T → Conflict Serializable.
T F → This Case Not exist
F T → View (Serializable)
F F → Not Serializable
∴

View Equivalence [S & S']

On Each Data Item 3 Condition Must be Satisfied.

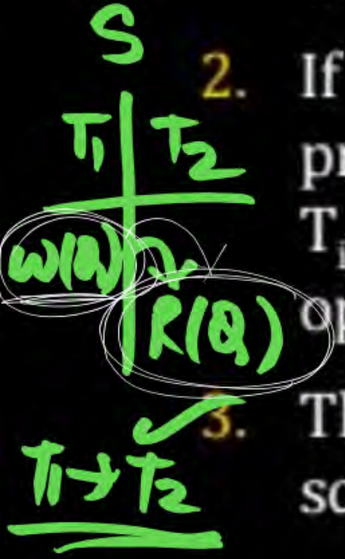
- ① Initial Read
- ② Final write
- ③ Write-Read(sequence)
  (Updated-Read)

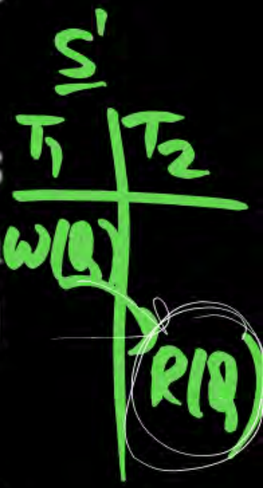
} executed
Must be in same order in S, S'.

View Serializability

Let S and S' be two schedules with the same set of transactions. S and S' are **view equivalent** if the following three conditions are met, for each data item Q,

1. If in schedule S, transaction T_i reads the initial value of Q, then in schedule S' also transaction T_i must read the initial value of Q.
2. If in schedule S transaction T_i executes **read**(Q), and that value was produced by transaction T_j (if any), then in schedule S' also transaction T_i must read the value of Q that was produced by the same **write**(Q) operation of transaction T_j.
3. The transaction (if any) that performs the final **write**(Q) operation in schedule S must also perform the final **write**(Q) operation in schedule S'.





Serializable objective ⇒ (Correct Result) Consistent Result along with Concurrent Execution.

View Serializability

- **View Serializable Schedule:** View equivalent serial schedule.
- **View Equivalent:** S_1 and S_2 said to be view equivalent.

  Only if

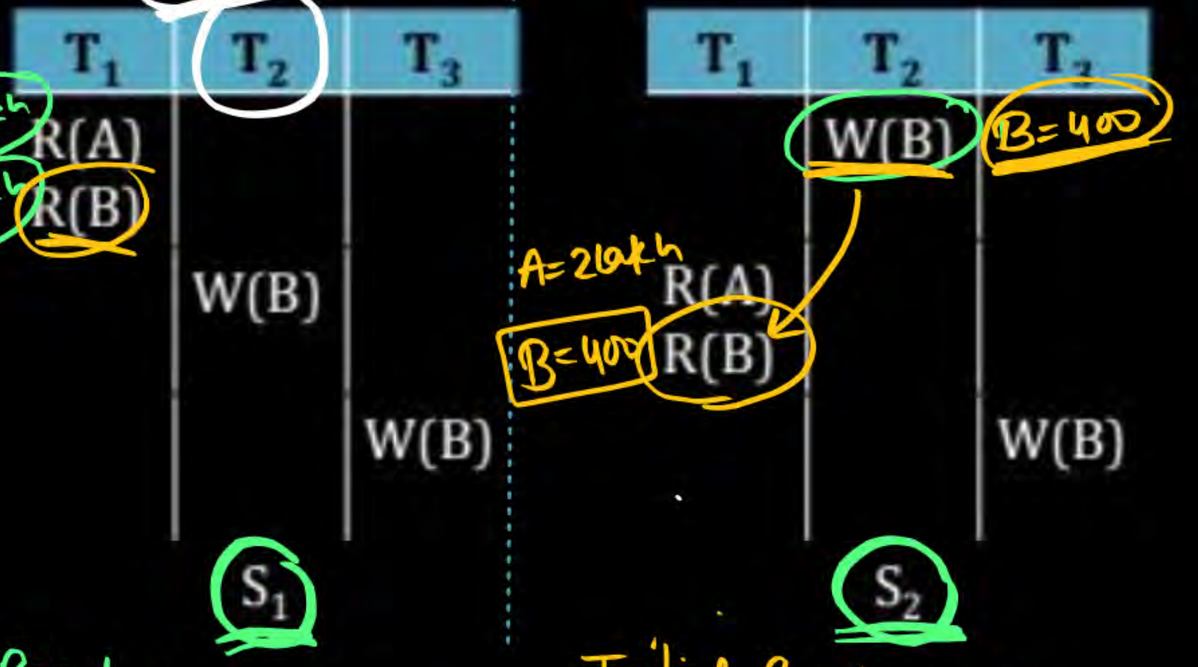
  [1] initial reads of S_1 and S_2 should be same.

Initial Balance (value) of
A: 2 lakh
B: 5 lakh

Rahul
Mohit

Example

400 Rs



Initial Read on A: T_1, B: T_1

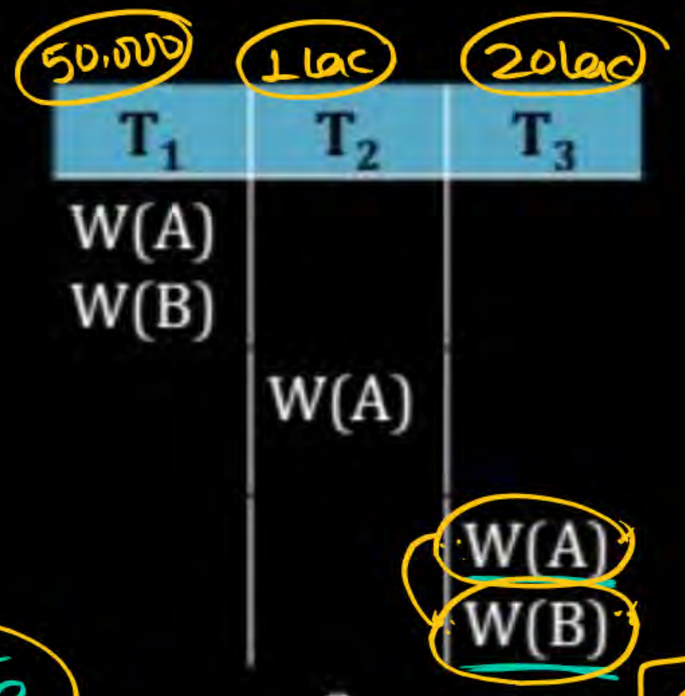
$S_1 \neq S_2$

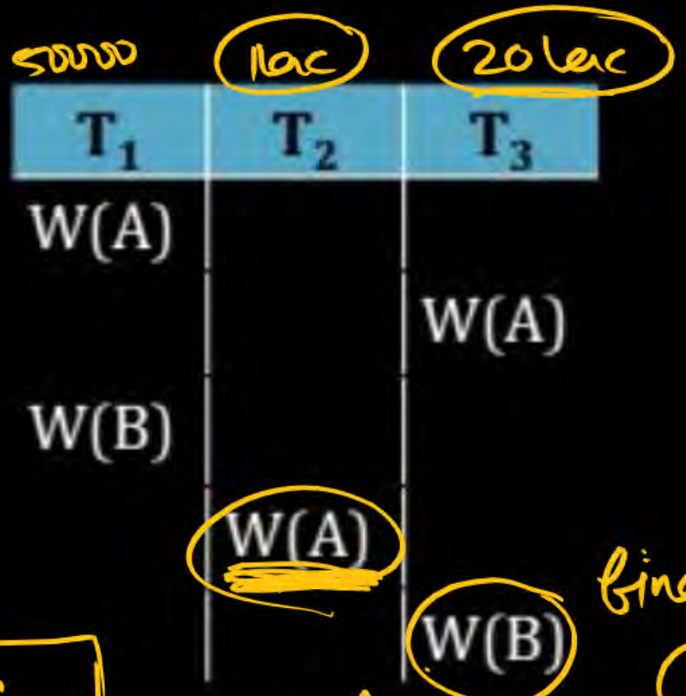
Initial Read on A: T_1, B: Not T_1

Initial Read on each Data Item Must be same.

View Serializability

2. Final updations for every data item should be same in S_1 and S_2





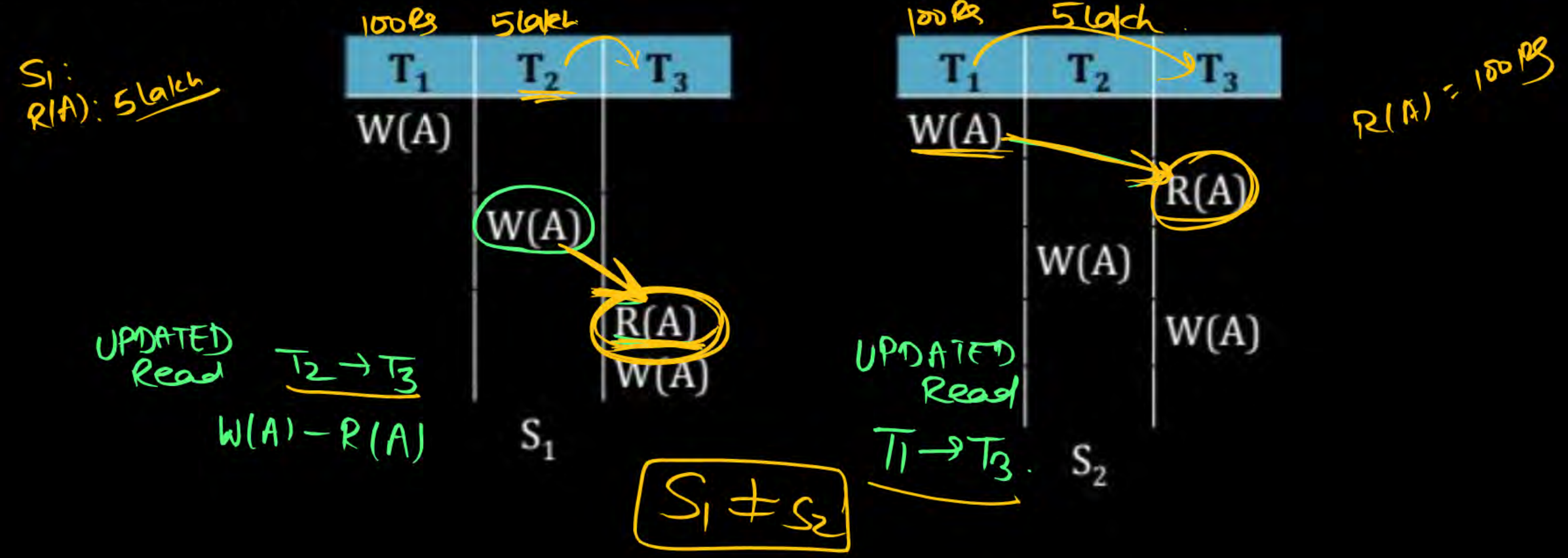
Final Write On A: T_3 B: T_3

$S_1 \neq S_2$

Final write on A: T_2 (Not T_3) B: T_3

View Serializability

3. Write-Read sequence should also be equal. (Updated Reads should be same



Q.2

S1

T_1	T_2
R(A)	
W(A)	
	R(A)
	W(A)
R(B)	
W(B)	
	R(B)
	W(B)

T_1 T_2 Conflict Serializable $\langle T_1\ T_2 \rangle$

& View Conflict serializable

① Initial Read
On A: T_1
B: T_1

② final Write
On A: T_2
B: T_2

③ Updated Read (Write - Read Sequence)
A: $T_1 \to T_2$
B: $T_1 \to T_2$

S2

T_1	T_2
	R(A)
	W(A)
	R(B)
	W(B)
R(A)	
W(A)	
R(B)	
W(B)	

Initial Read
On A: T_2
B: T_2

$S' \langle T_2\ T_1 \rangle$

S_1 is View equivalent to S_2

S3

T_1	T_2
R(A)	
W(A)	
R(B)	
W(B)	
	R(A)
	W(A)
	R(B)
	W(B)

① Initial Read
On A: T_1
B: T_1

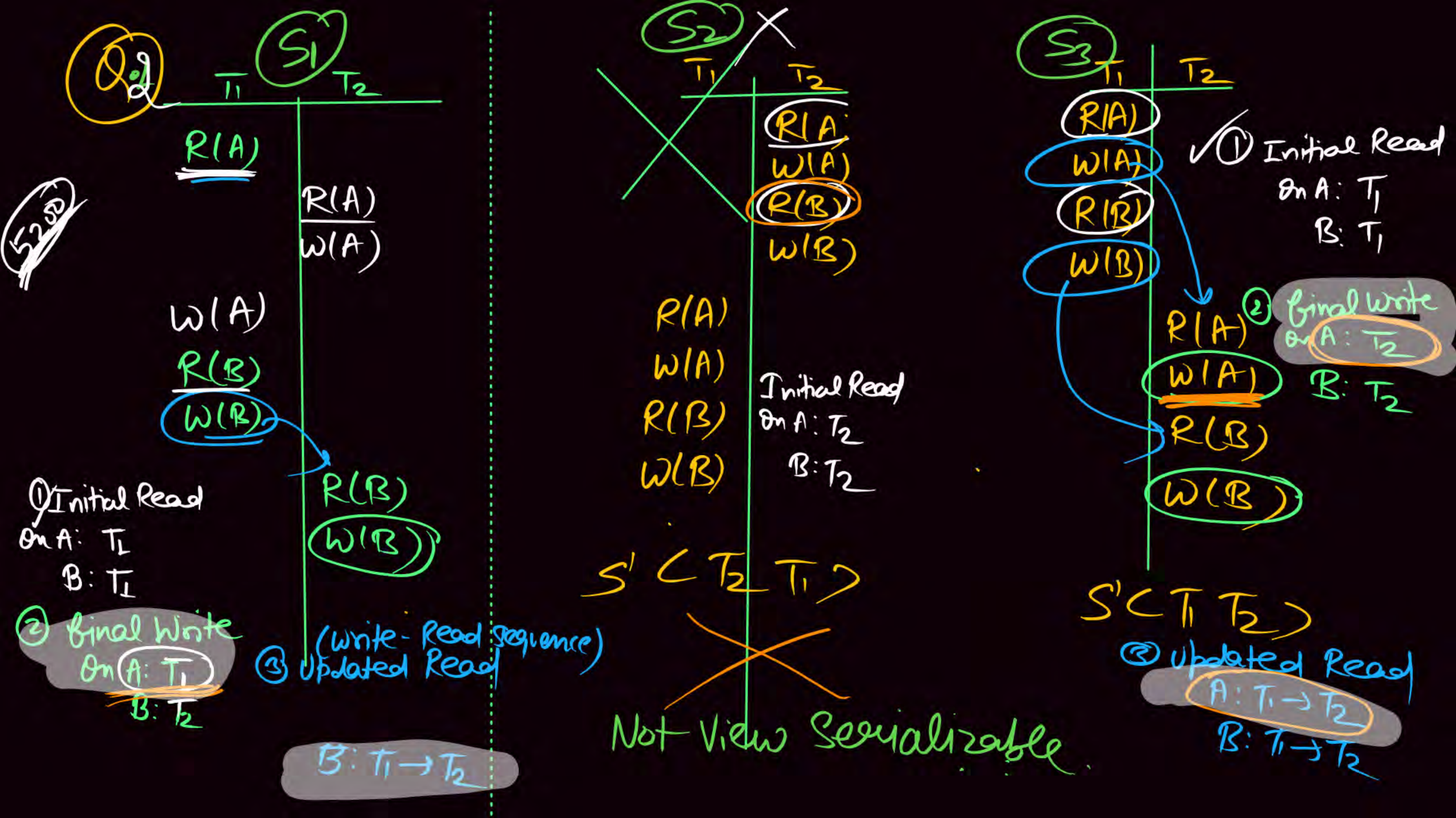
② final Write
On A: T_2
B: T_2

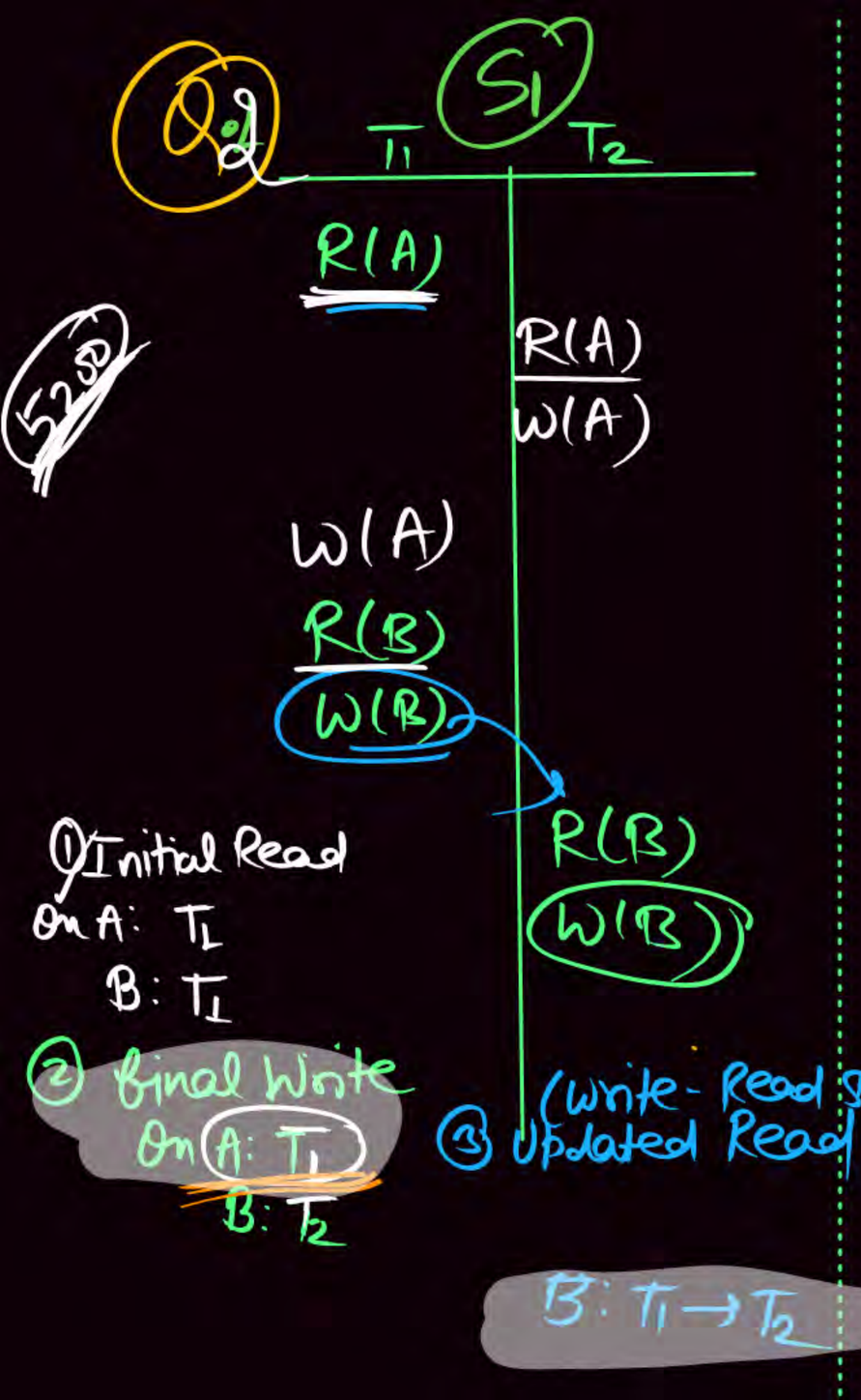
$S' \langle T_1\ T_2 \rangle$

③ Updated Read
A: $T_1 \to T_2$
B: $T_1 \to T_2$

S_1 is View equivalent to S_3

$\langle T_1\ T_2 \rangle$


S1
T1
T2
R(A)
R(A)
W(A)
W(A)
R(B)
W(B)
R(B)
W(B)
① Initial Read
On A: T1
B: T1
② Final Write
On A: T1
B: T2
③ Updated Read
(Write - Read Sequence)
B: T1 → T2
S2
T1
T2
R(A)
W(A)
R(B)
W(B)
R(A)
W(A)
R(B)
W(B)
Initial Read
On A: T2
B: T2
S' < T2 T1 >
Not View Serializable
S3
T1
T2
R(A)
W(A)
R(B)
W(B)
① Initial Read
On A: T1
B: T1
R(A)
W(A)
R(B)
W(B)
② Final Write
On A: T2
B: T2
S' < T1 T2 >
② Updated Read
A: T1 → T2
B: T1 → T2



Cycle Not Conflict

& Not View Serializable

Not Serializable ⇒ Inconsistent Result

(eg 5200 Wallah)

Q.3 Given Question

T_1	T_2	T_3
R(Q)		
	W(Q)	
W(Q)		
		W(Q)

Cycle Not Conflict

Check for View → 6 possibility.

$3! = 6$ Serial Schedule.

<T1 T2 T3>

<T1 T3 T2> X

X <T2 T1 T3>

X <T2 T3 T1>

X <T3 T1 T2>

X <T3 T2 T1>

Initial Read T_1.

Approach

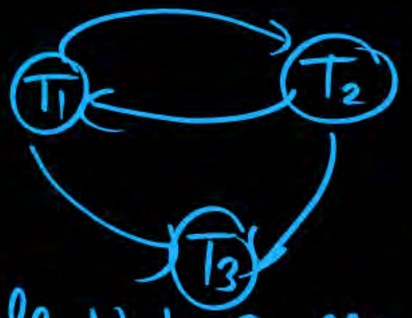
Dummy

OR

eliminate other option.

Q.3

T_1	T_2	T_3
R(Q)		
	W(Q)	
W(Q)		
		W(Q)



Cycle Not Conflict

① Initial Read on Q: T_1

② final write on Q: T_3

③ Updated Read Not Present

T_1	T_2	T_3
R(Q)		
W(Q)		
	W(Q)	
		W(Q)

View serializable

$\langle T_1\ T_2\ T_3 \rangle$

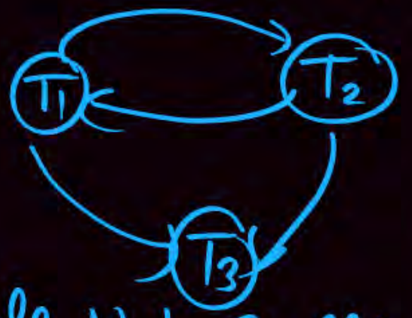
① Initial Read on Q: T_1

② final write on Q: T_3

③ Updated Read Not Present

Q.3

T_1	T_2	T_3
R(Q)		
	W(Q)	
W(Q)		
		W(Q)



Cycle Not Conflict

① Initial Read on Q: T_1

② final write on Q: T_3

③ Updated Read Not present

Not Conflict But View serializable

Serializable Schedule (Blind write (T_2 & T_3))

View serializable $<T_1\ T_2\ T_3>$.

① Initial Read on Q: T_1

② final write on Q: T_3

③ Updated Read Not present

View Serializability (Cont.)

- A schedule S is **view serializable** if it is view equivalent to a serial schedule.
- Every conflict serializable schedule is also view serializable.
- Below is a schedule which is view-serializable but not conflict serializable.

T_{27}	T_{28}	T_{29}
read(Q)		
	write(Q)	
write(Q)		
		write(Q)

Blind Write : Write Without Reading

[T_{28}, T_{29} Without Read(Q), Perform Write(Q)]

Note:

Every view serializable schedule that is not conflict serializable has **blind writes.**

Q.1

HW

T_1	T_2	T_3
	R(A)	
	R(B)	
W(B)		
		R(B)
W(A)		
	W(A)	
		W(A)

$\langle T_1\ T_2\ T_3 \rangle$

$\langle T_1\ T_3\ T_2 \rangle$

$\langle T_2\ T_1\ T_3 \rangle$

$\langle T_2\ T_3\ T_1 \rangle$

$\langle T_3\ T_1\ T_2 \rangle$

$\langle T_3\ T_2\ T_1 \rangle$.

Q.1

T_1	T_2	T_3
	R(A)	
	R(B)	
W(B)		
		R(B)
W(A)		
	W(A)	
		W(A)

Q. 2

Consider the following schedule S of transactions T_1 and T_2:

Which of the following is TRUE about the schedule S? [2004: 2 Marks]

T_1	T_2
Read(A)	
A = A–10	
	Read(A)
	Temp = 0.2 * A
	Write(A)
	Read(B)
Write(A)	
Read(B)	
B = B +10	
Write(B)	
	B = B + Temp
	Write(B)

(A) S is serializable only as T_1, T_2

(B) S is serializable only as T_2, T_1

(C) S is serializable both as T_1, T_2 and T_2 T_1

(D) S is not serializable either as T_1 or as T_2

MCQ

Q.3 HW

Consider the following schedule for transactions T1, T2 and T3:

Which one of the schedules below is the correct serialization of the above?

[GATE-2010-CS: 2M]

T1	T3	T3
Read(X)		
	Read (Y)	
		Read (Y)
	Write (Y)	
Write (X)		
		Write (X)
	Read (X)	
	Write (X)	

A) T 1 → T 3 → T 2

B) T 2 → T 1 → T 3

C) T2 → T3 → T1

D) T3 → T 1 → T 2

MCQ

Consider two transactions T_1 and T_2, and four schedules S_1, S_2, S_3, S_4 of T_1 and T_2 as given below:

T_1: $R_1[x]$ $W_1[x]$ $W_1[y]$;

T_2: $R_2[x]$ $R_2[y]$ $W_2[y]$;

S_1: $R_1[x]$ $R_2[x]$ $R_2[y]$ $W_1[x]$ $W_1[y]$ $W_2[y]$;

S_2: $R_1[x]$ $R_2[x]$ $R_2[y]$ $W_1[x]$ $W_2[y]$ $W_1[y]$;

S_3: $R_1[x]$ $W_1[x]$ $R_2[x]$ $W_1[y]$ $R_2[y]$ $W_2[y]$;

S_4: $R_2[x]$ $R_2[y]$ $R_1[x]$ $W_1[x]$ $W_1[y]$ $W_2[y]$;

Which of the above schedules are conflict serializable?

[GATE-2009-CS: 2M]

A S_1 and S_2

B S_3 and S_3

C S_3 only

D S_4 only

MCQ

Q.S HW

Consider the following three schedules of transactions T1, T2 and T3. [Notation: In the following NYO represents the action Y (R for read, W for write) performed by transaction N on object O.]

S1: 2RA 2WA 3RC 2WB 3WA 3WC 1RA 1RB 1WA 1WB

S2: 3RC 2RA 2WA 2WB 3WA 1RA 1RB 1WA 1WB 3WC

S3: 2RA 3RC 3WA 2WA 2WB 3WC 1RA 1RB 1WA 1WB

Which of the following statements is TRUE? [GATE-2008-CS: 2M]

A. S1, S2 and S3 are all conflict equivalent to each other

B. No two of S1, S2 and S3 are conflict equivalent to each other

C. S2 is conflict equivalent to S 3, but not to S1

D. S1 is conflict equivalent to S2, but not to S3

MCQ

Q6 HW

Consider the transactions T1, T2 and T3 and the schedules S1 and S2 given below.

T1: r1(X); r1(Z); w1(X); w1(Z)
T2: r2(Y) ; r2(Z) ; w2(Z)
T3: r3(Y); r3(X); w3(Y)
S1: r1(X); r3(Y); r3(X); r2(Y); r2(Z); w3(Y); w2(Z); r1(Z); w1(X); w1(Z)
S2: r1(X); r3(Y); r2(Y); r3(X); r1(Z); r2(Z); w3(Y); w1(X); w2(Z); w1(Z)

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

[GATE-2014-CS: 2M]

A. Only S1 is conflict-serializable.

B. Only S2 is conflict-serializable.

C. Both S1 and S2 are conflict-serializable.

D. Neither S1 nor S2 is conflict-serializable.

Any Doubt ?

THANK YOU!