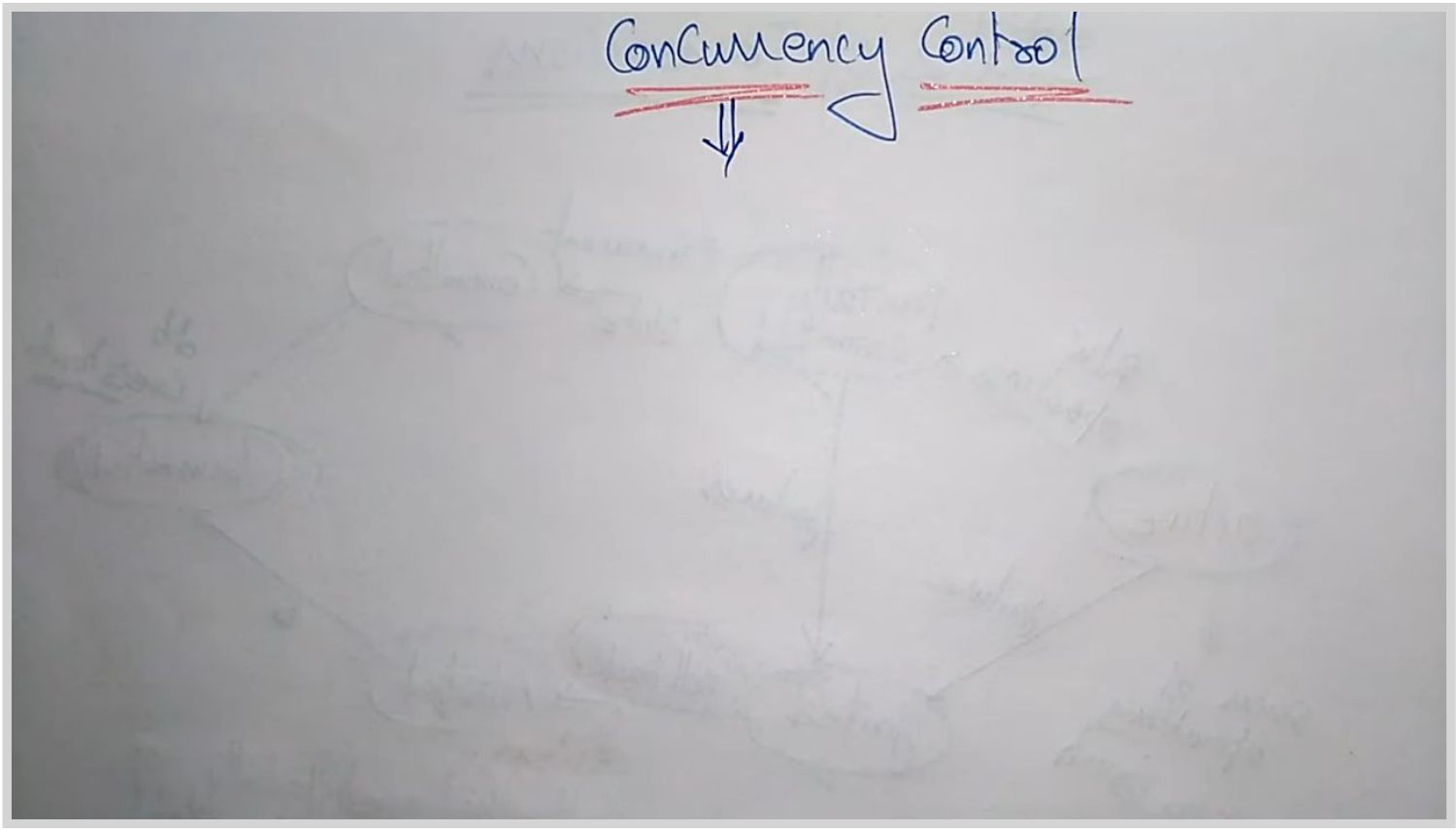
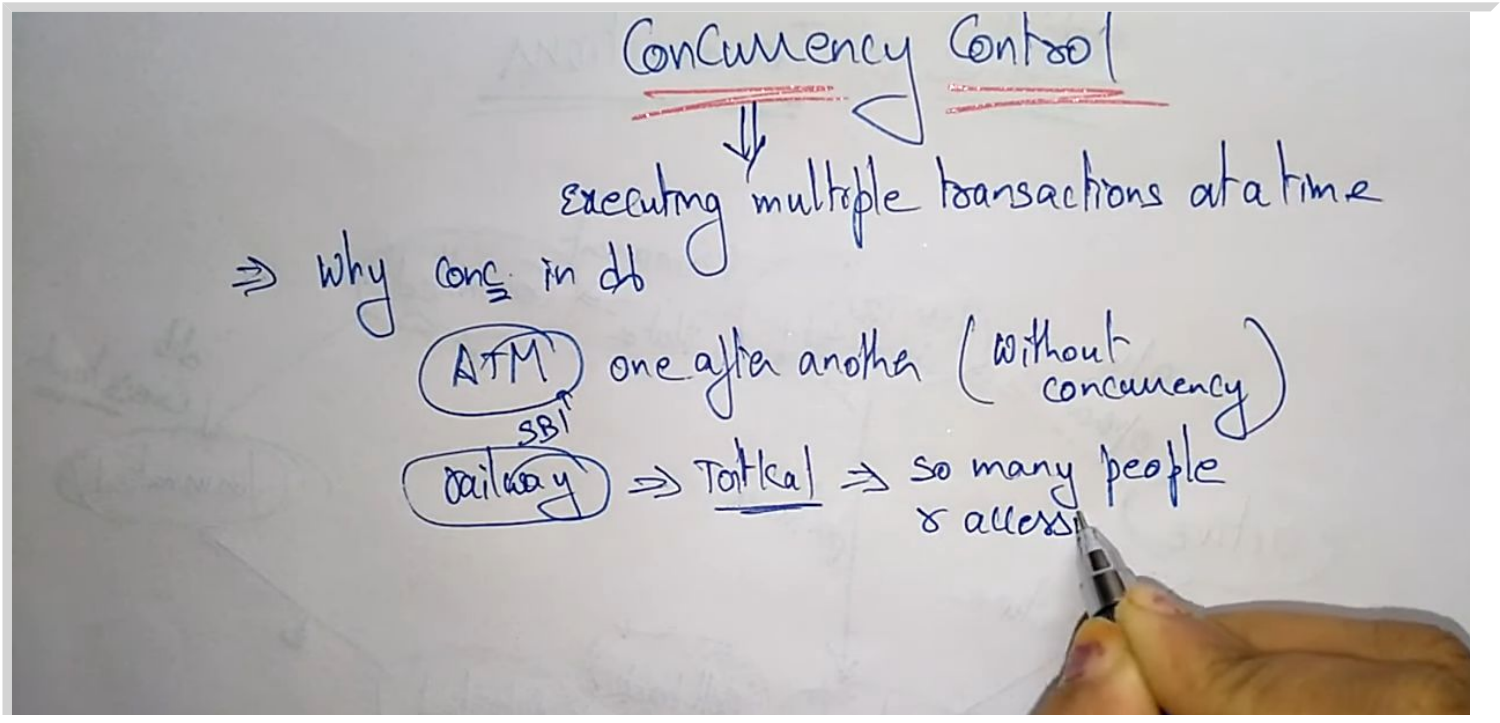
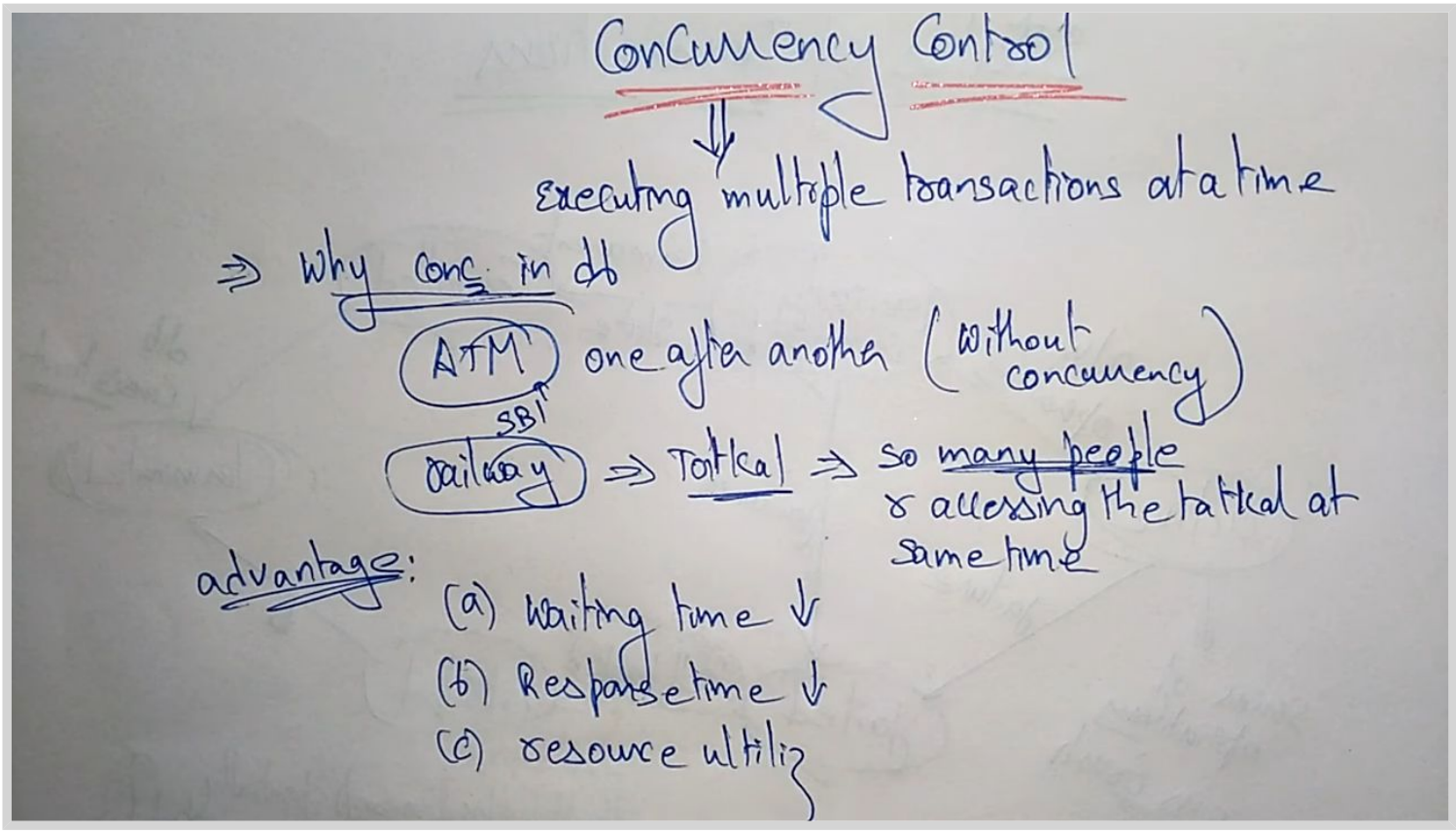
00:16



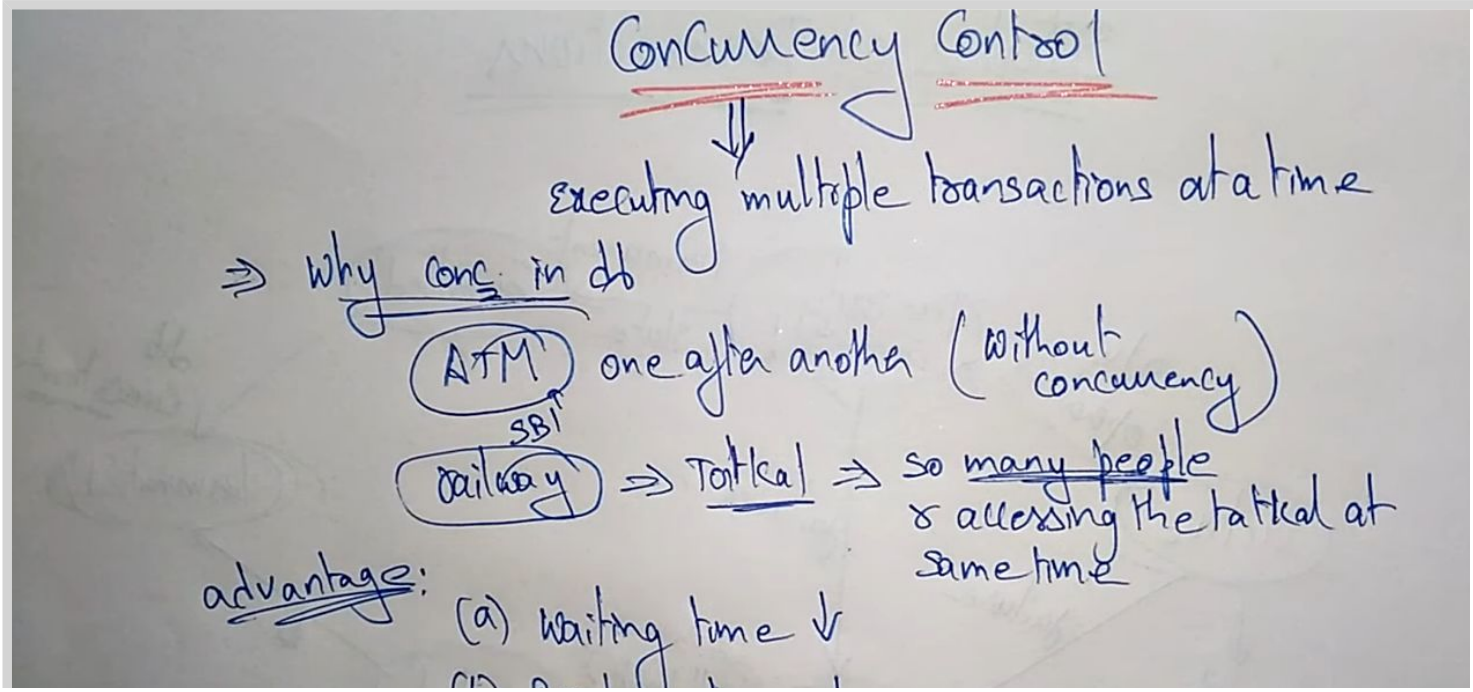
02:41



04:02


Concurrency Control
Executing multiple transactions at a time
⇒ Why conc in db
ATM one after another (without concurrency)
SBI
Railway ⇒ Tatkal ⇒ so many people r accessing the tatkal at same time
advantage:
(a) waiting time ↓
(b) Response time ↓
(c) resource utiliz

03:55


Concurrency Control
Executing multiple transactions at a time
⇒ Why conc in db
ATM one after another (without concurrency)
SBI
Railway ⇒ Tatkal ⇒ so many people r accessing the tatkal at same time
advantage:
(a) waiting time ↓

(b) Response time ↓

04:25

Executing multiple transactions at a time

⇒ Why conc in db

ATM → one after another (without concurrency)

SBI

Railway ⇒ Tatkal ⇒ so many people r accessing the tatkal at same time

advantage:

(a) waiting time ↓
(b) Response time ↓
(c) resource utilization ↑
(d) efficiency ↑

04:26

⇒ Why conc in db

ATM → one after another (without concurrency)

SBI

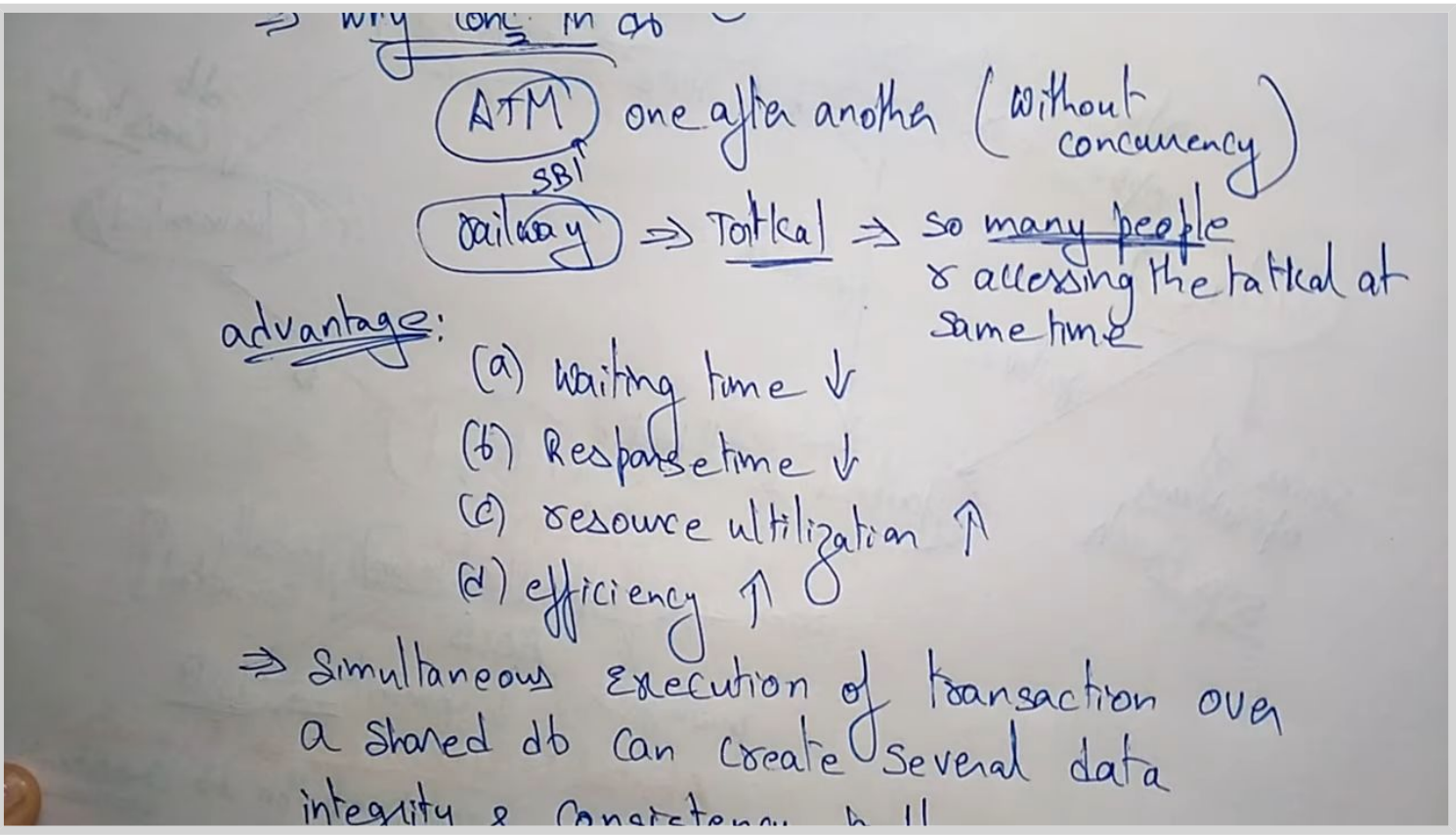
Railway ⇒ Tatkal ⇒ so many people r accessing the tatkal at same time

advantage:

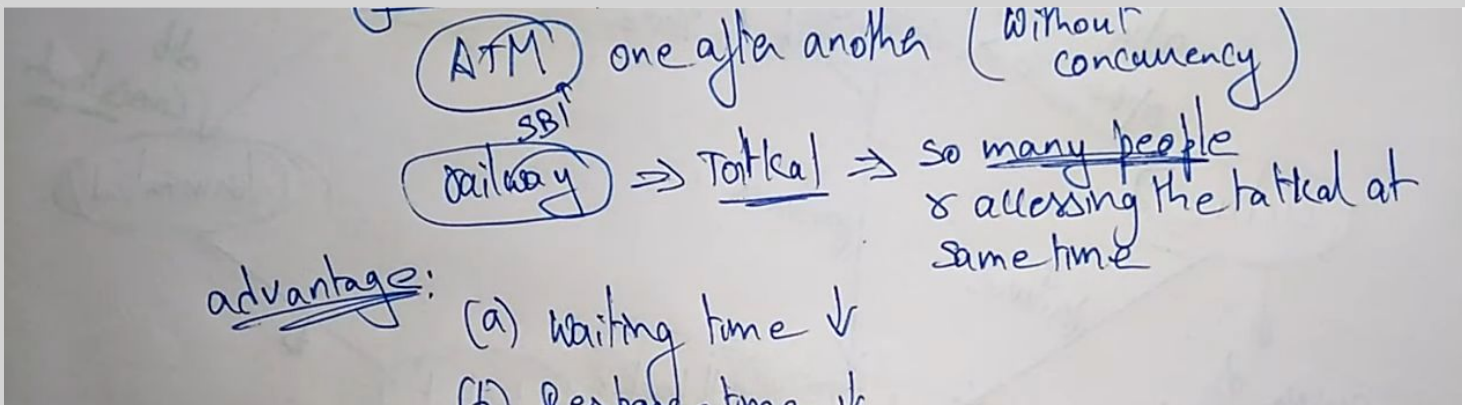
(a) waiting time ↓
(b) Response time ↓

(c) resource utilization ↑
(d) efficiency ↑

05:12



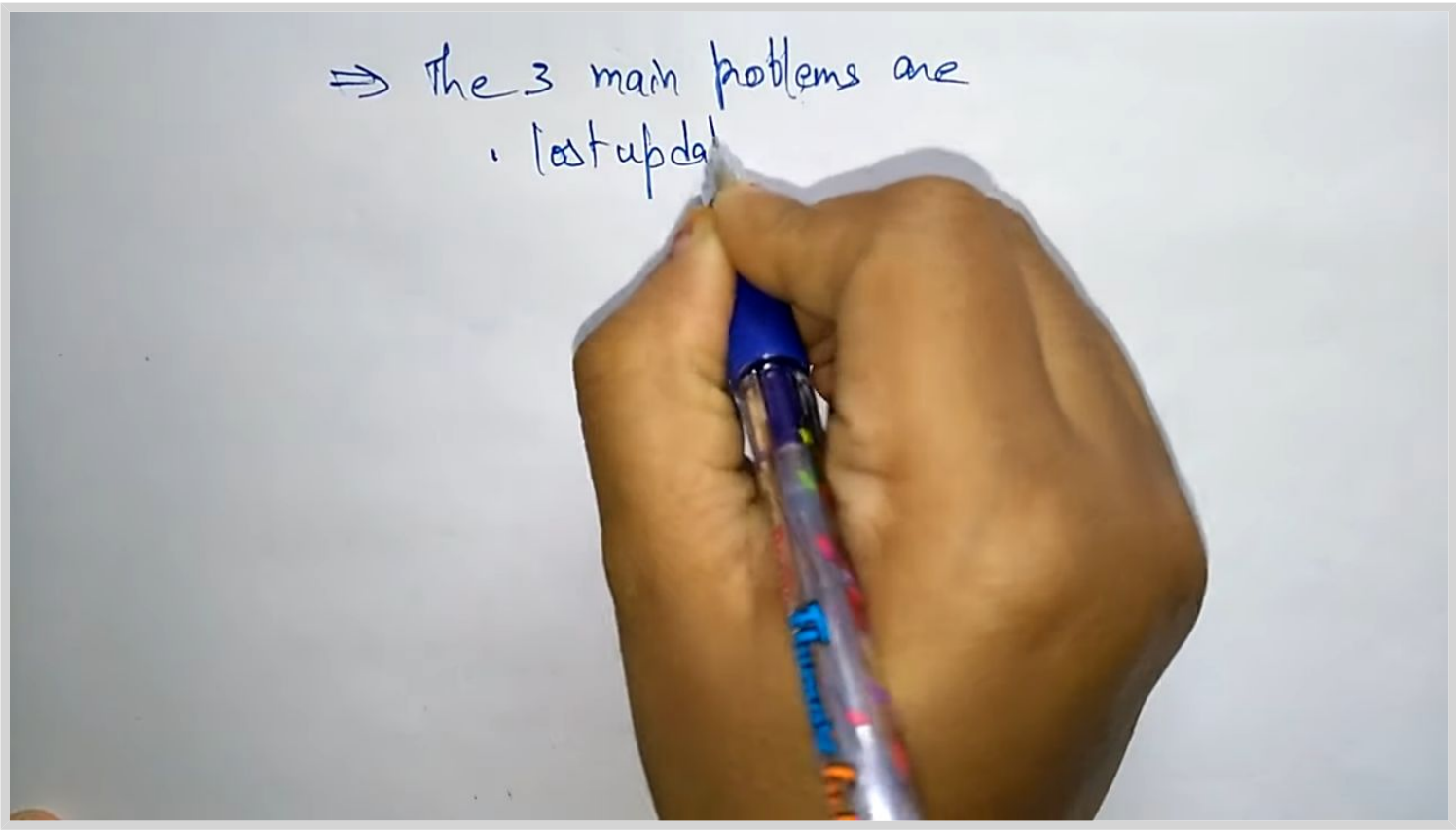
05:14



(c) resource utilization ↑
(d) efficiency ↑

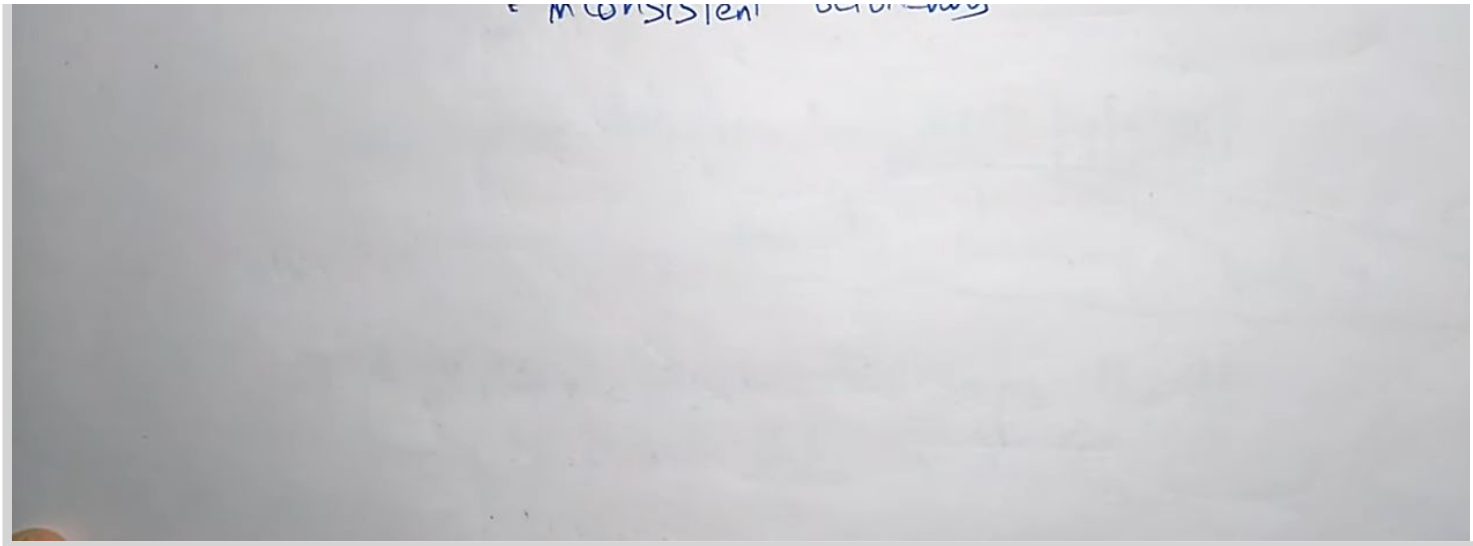
⇒ Simultaneous execution of transaction over a shared db can create several data integrity & consistency problems

06:19

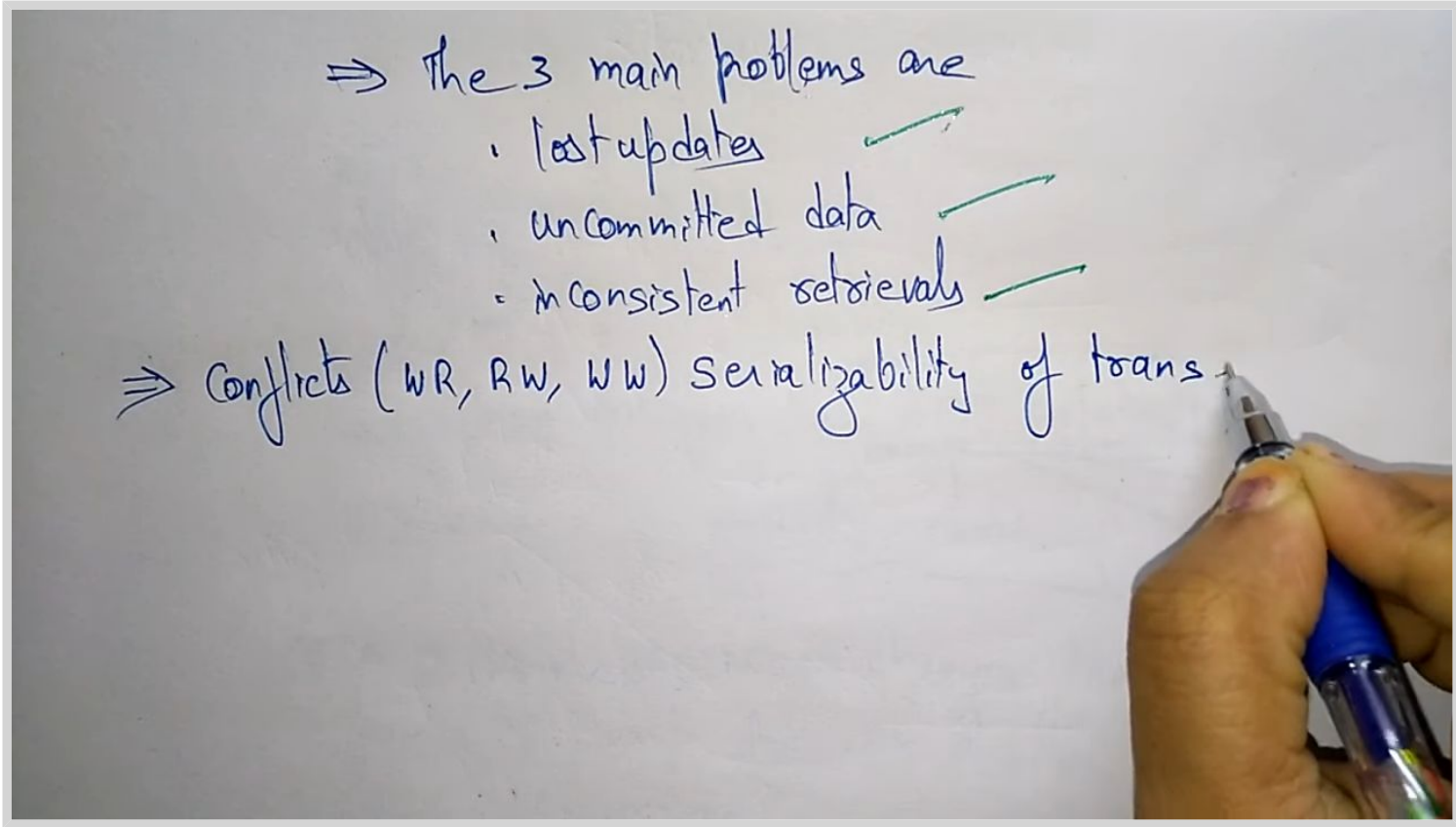


06:37

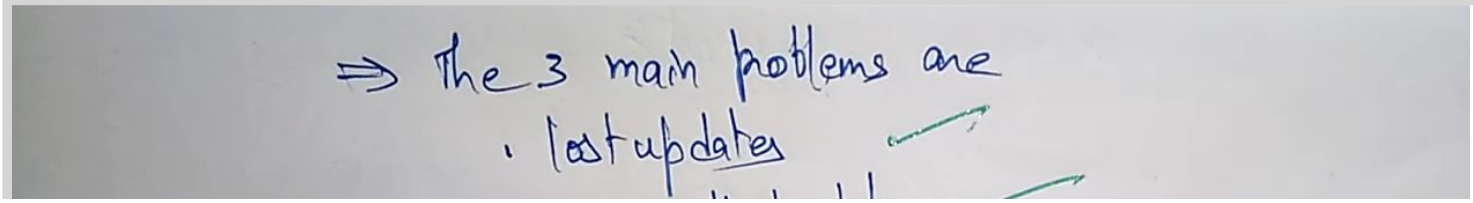
⇒ The 3 main problems are
. lost updates
. uncommitted data
. inconsistent retrievals

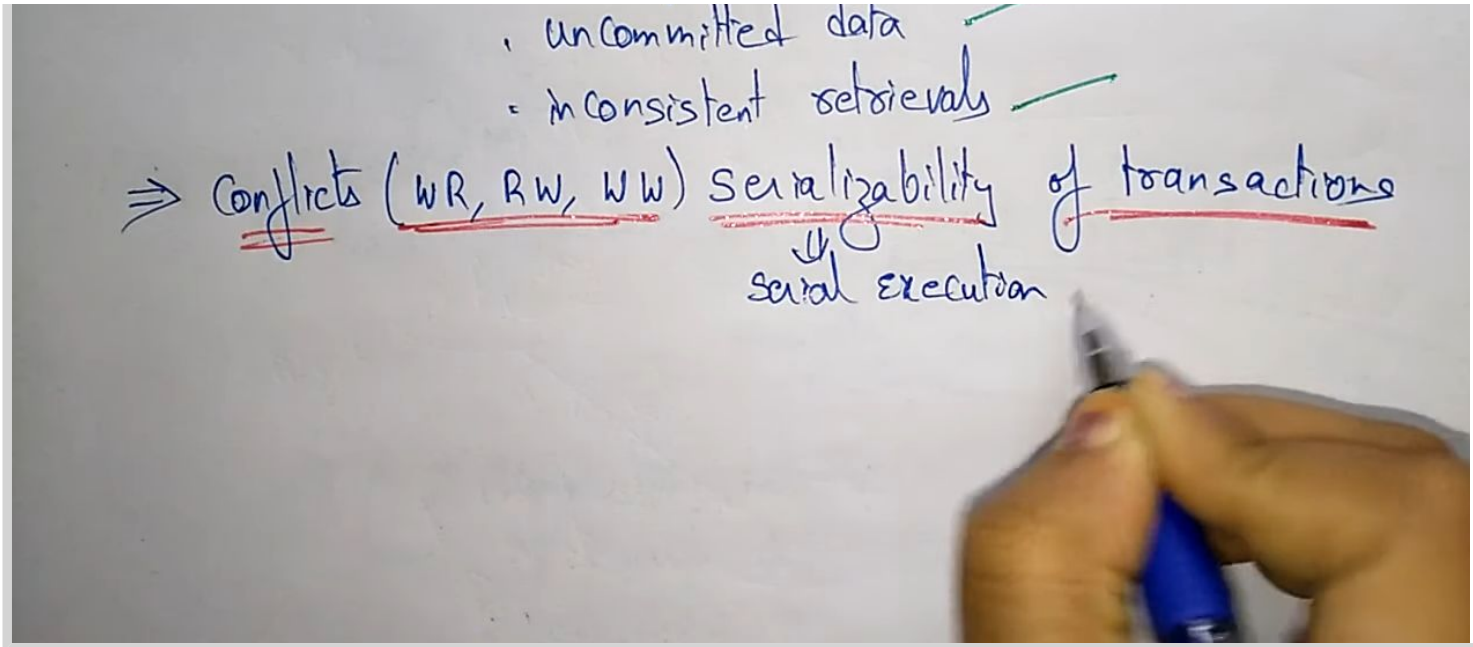

inconsistent

07:11


⇒ The 3 main problems are
. lost updates
. uncommitted data
. inconsistent retrievals
⇒ Conflicts (WR, RW, WW) serializability of trans.

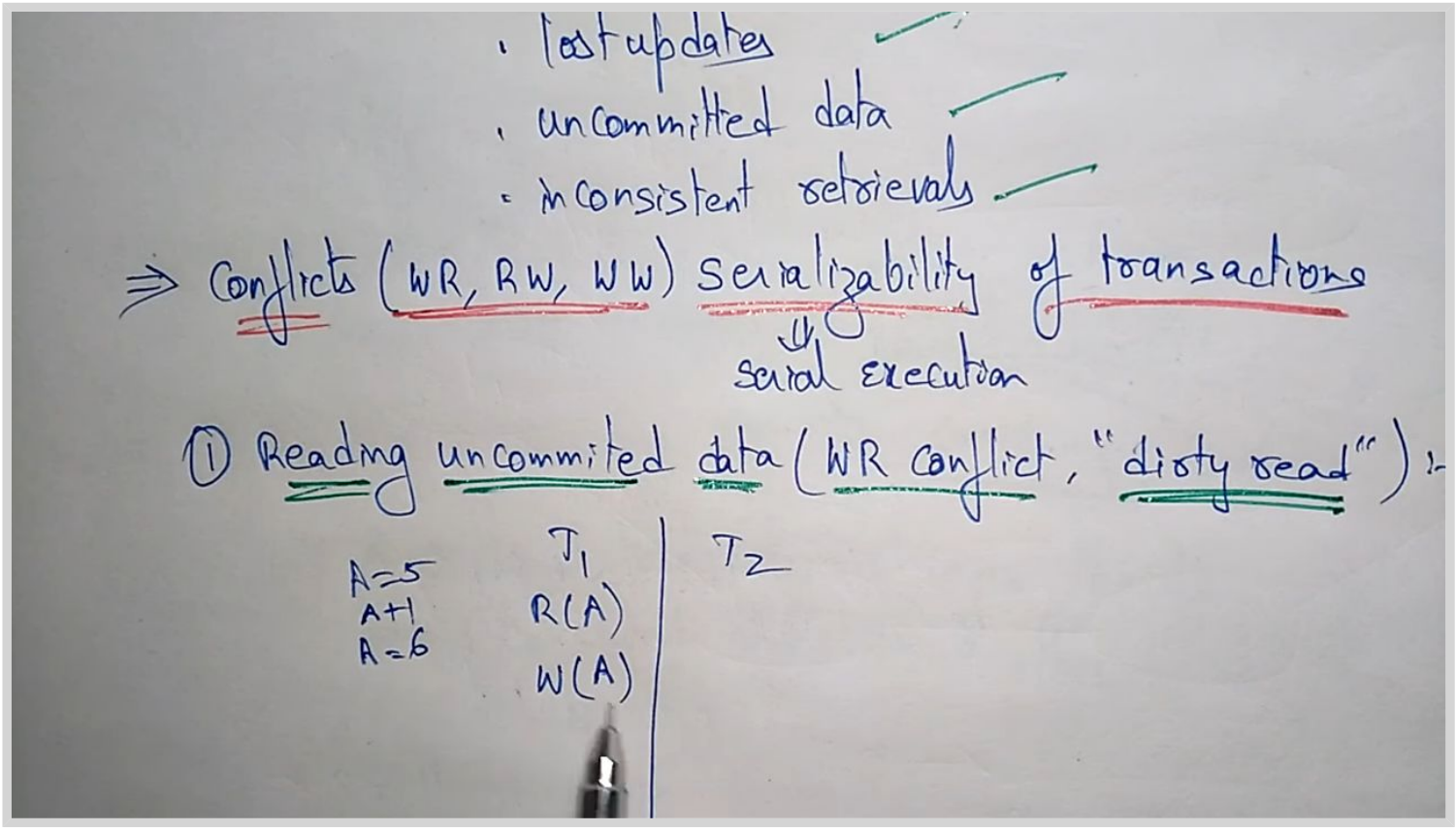
07:41


⇒ The 3 main problems are
. lost updates



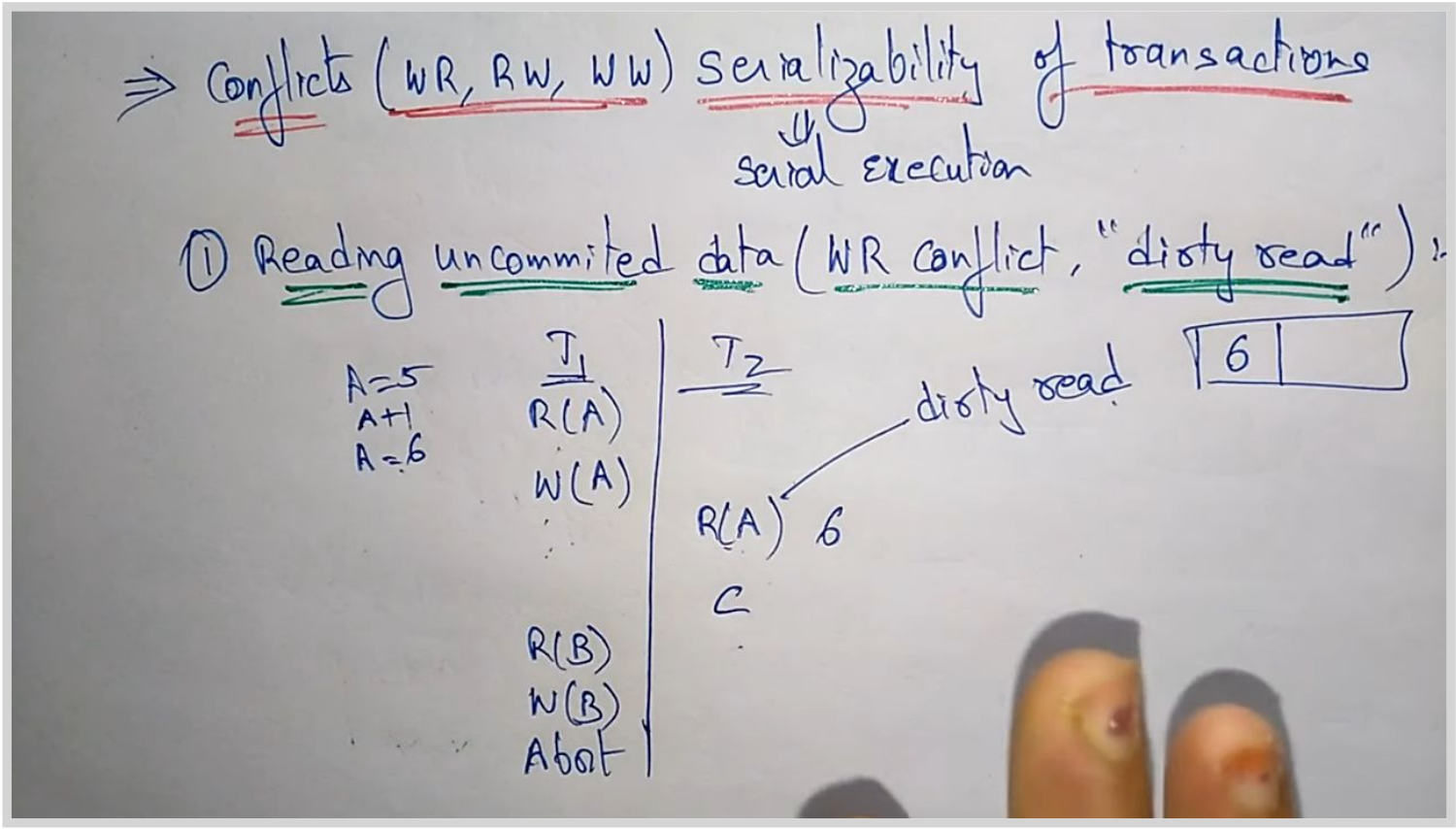
08:20

09:01

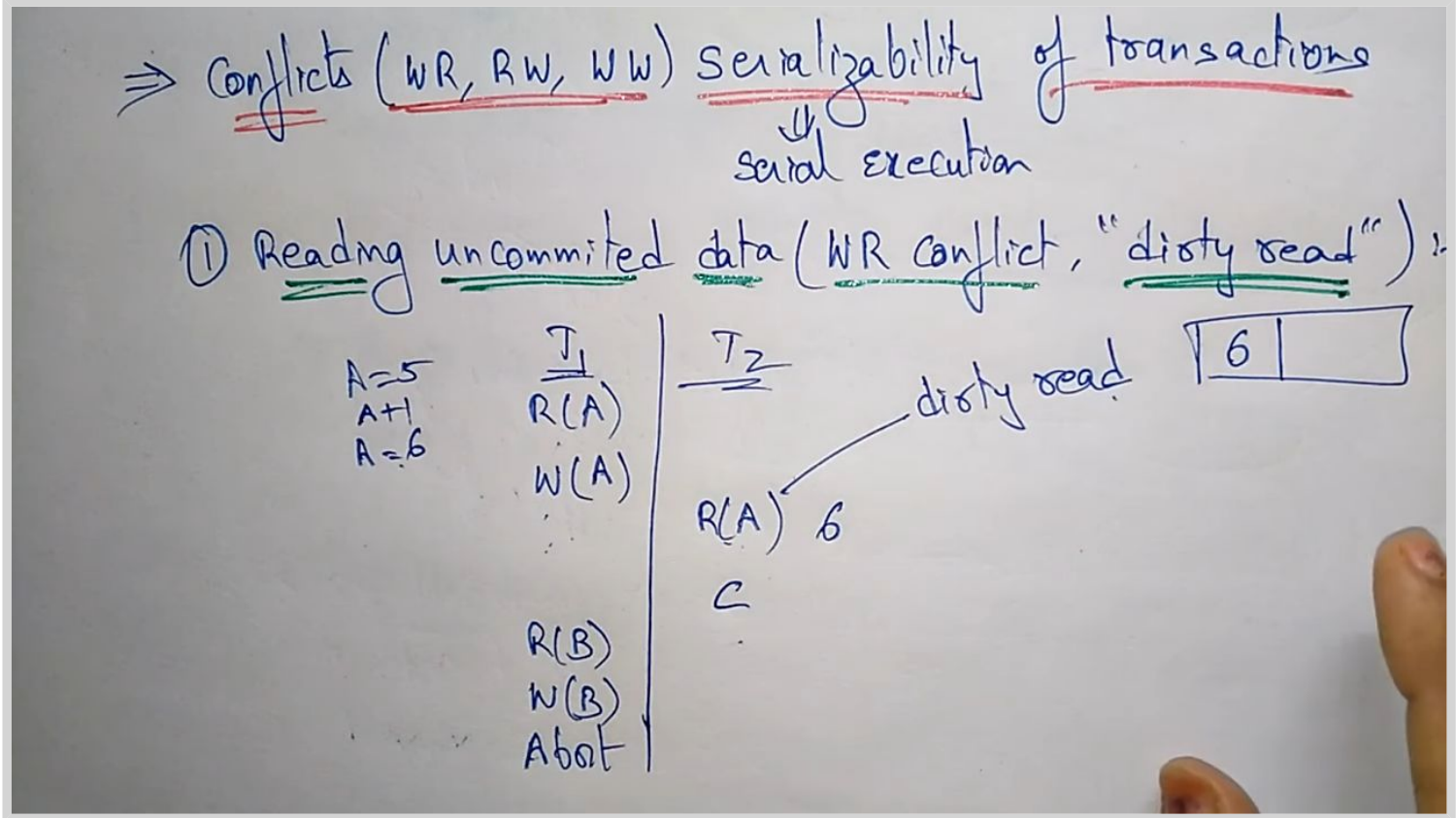


10:20

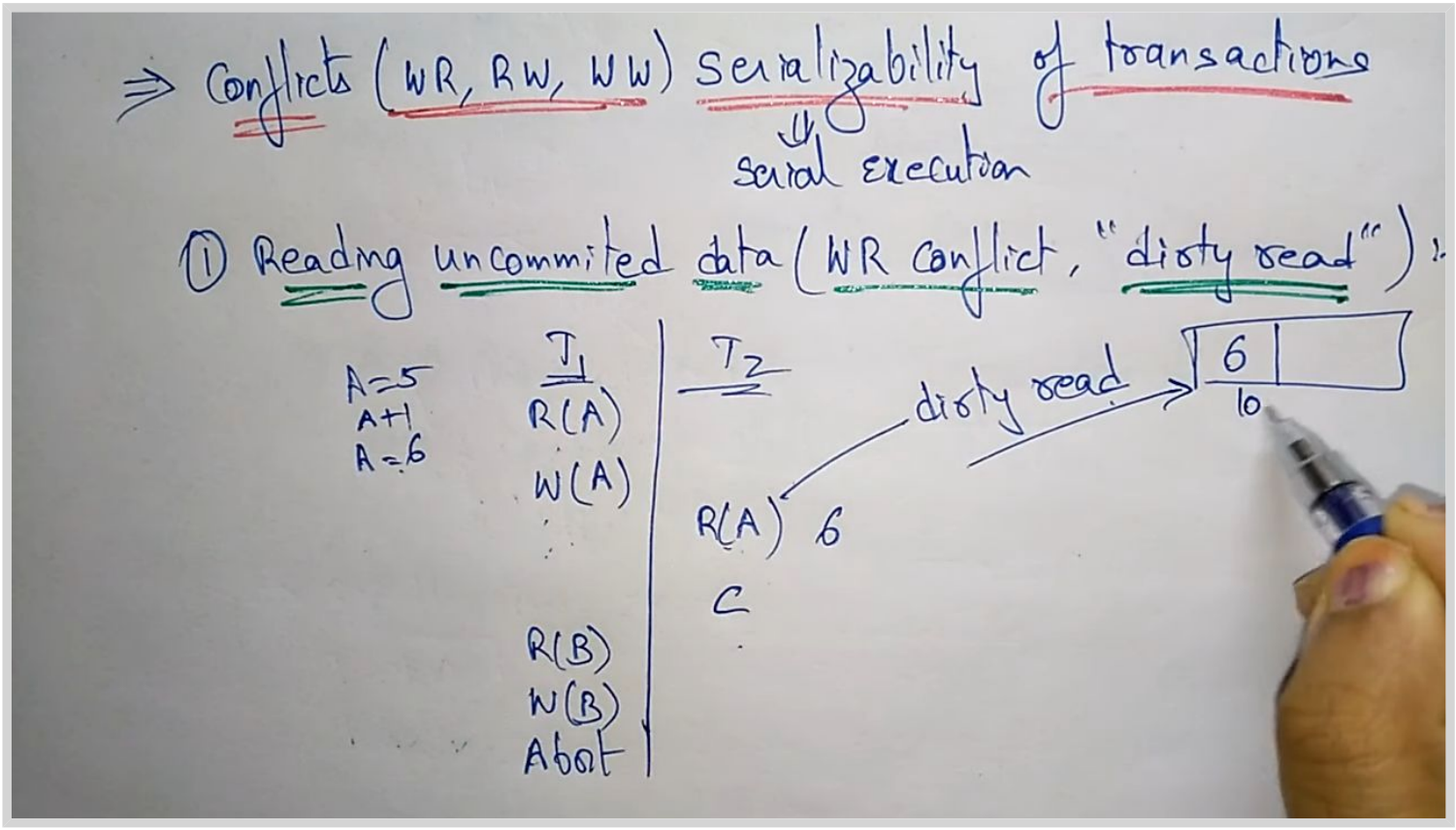
10:30


⇒ Conflicts (WR, RW, WW) serializability of transactions
serial execution
① Reading uncommited data (WR conflict, "dirty read"):-
A=5
A+1
A=6
T1
R(A)
W(A)
R(B)
W(B)
Abort
T2
R(A) 6
C
dirty read
6

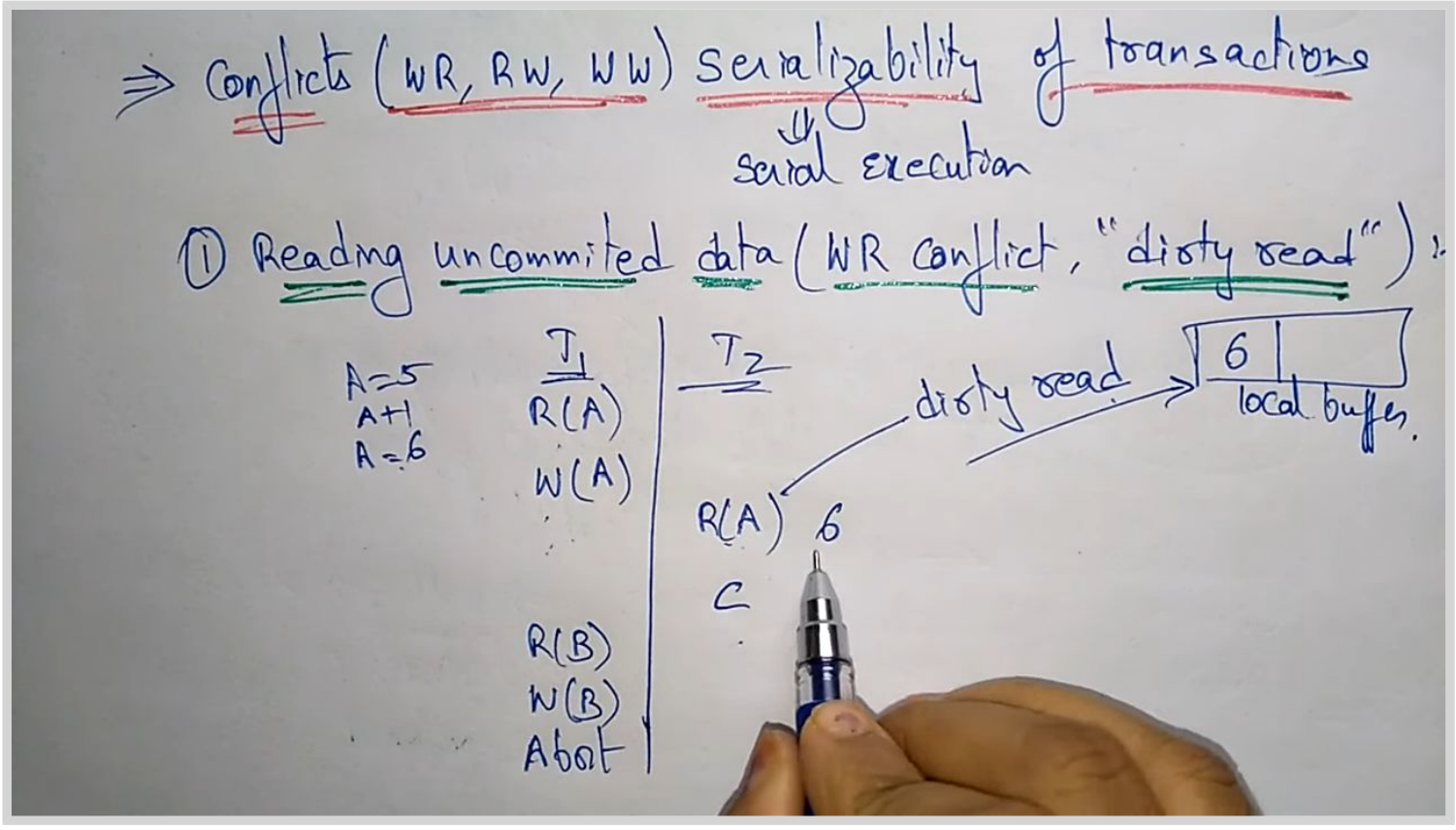
10:31


⇒ Conflicts (WR, RW, WW) serializability of transactions
serial execution
① Reading uncommited data (WR conflict, "dirty read"):-
A=5
A+1
A=6
T1
R(A)
W(A)
R(B)
W(B)
Abort
T2
R(A) 6
C
dirty read
6

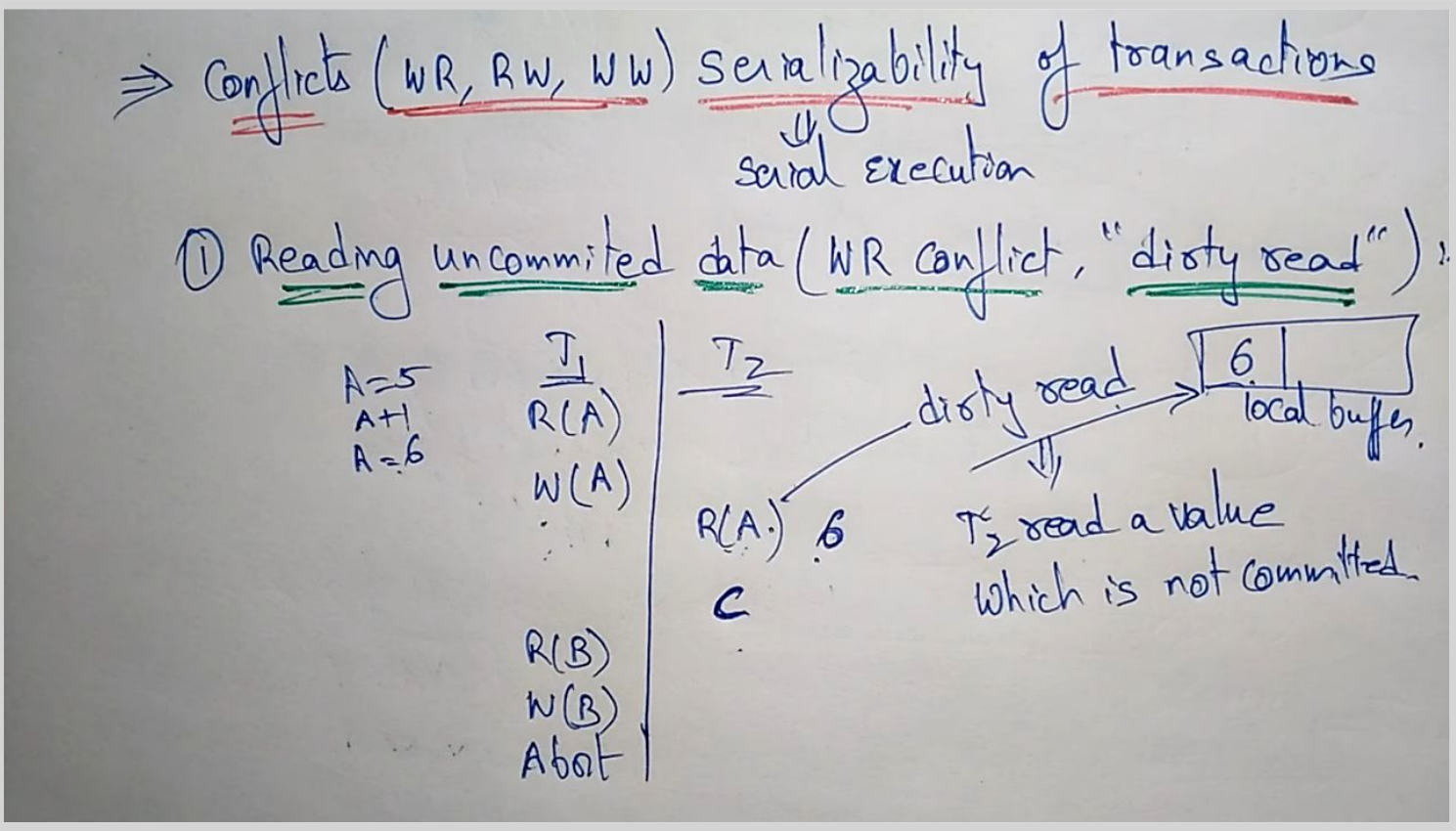
10:38


⇒ Conflicts (WR, RW, WW) serializability of transactions
serial execution
① Reading uncommited data (WR conflict, "dirty read"):
A=5
A+1
A=6
T1
R(A)
W(A)
R(B)
W(B)
Abort
T2
R(A) 6
C
dirty read
6
10

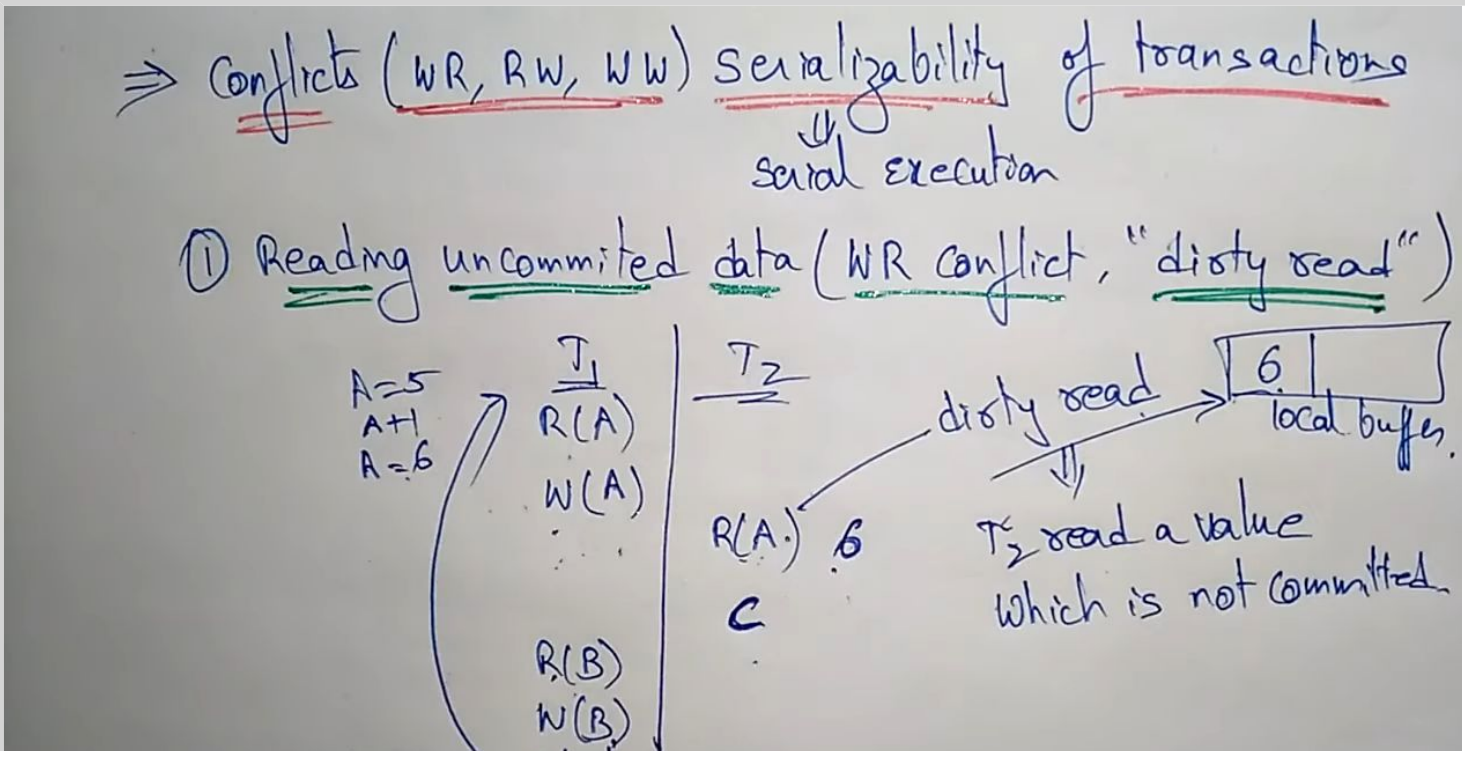
10:42


⇒ Conflicts (WR, RW, WW) serializability of transactions
serial execution
① Reading uncommited data (WR conflict, "dirty read"):
A=5
A+1
A=6
T1
R(A)
W(A)
R(B)
W(B)
Abort
T2
R(A) 6
C
dirty read
6
local buffer

11:24

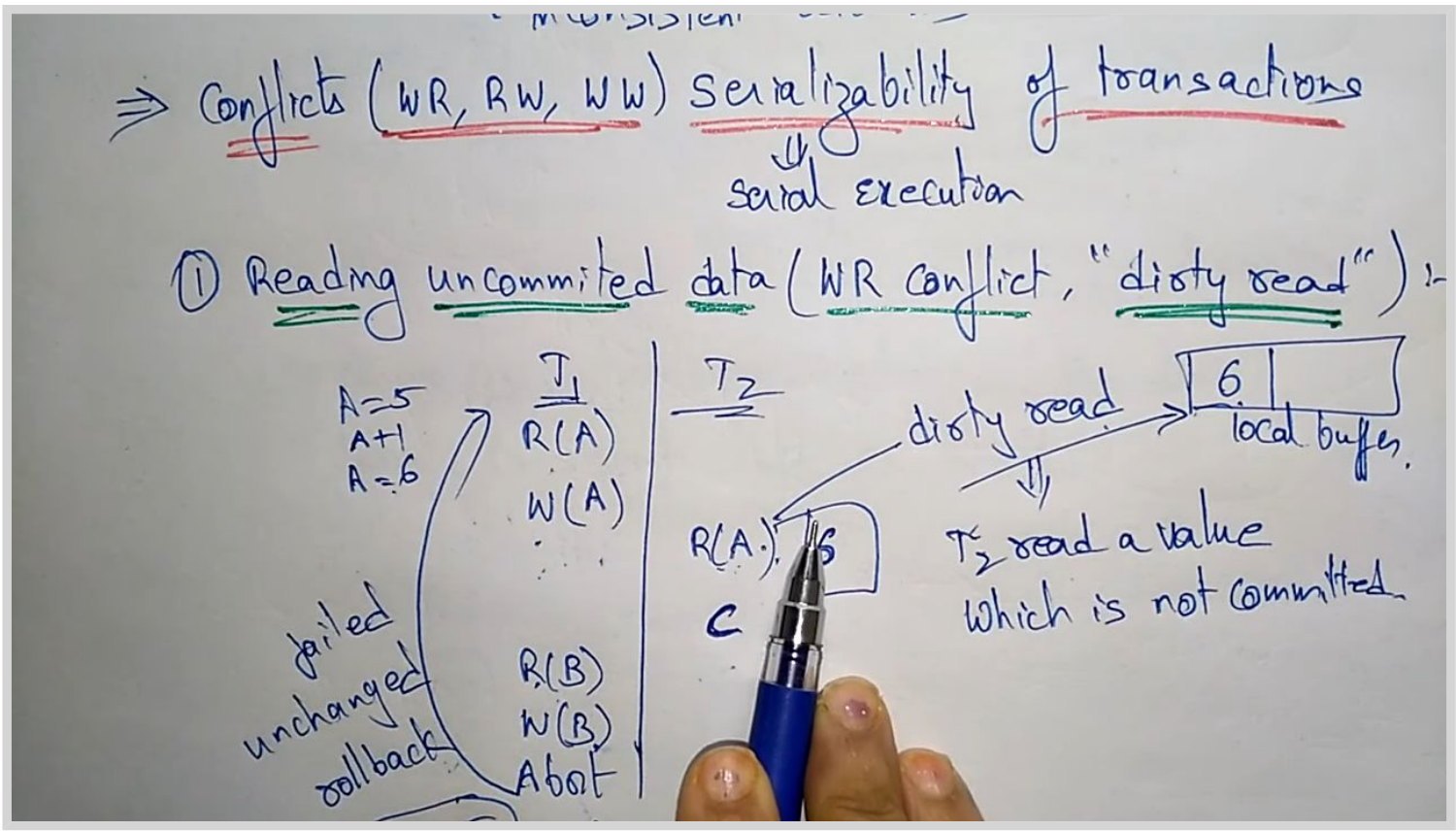

⇒ Conflicts (WR, RW, WW) serializability of transactions
serial execution
① Reading uncommited data (WR conflict, "dirty read"):
A=5
A+1
A=6
T1
R(A)
W(A)
R(B)
W(B)
Abort
T2
R(A) 6
C
dirty read
6
local buffer
T2 read a value
which is not committed

11:41

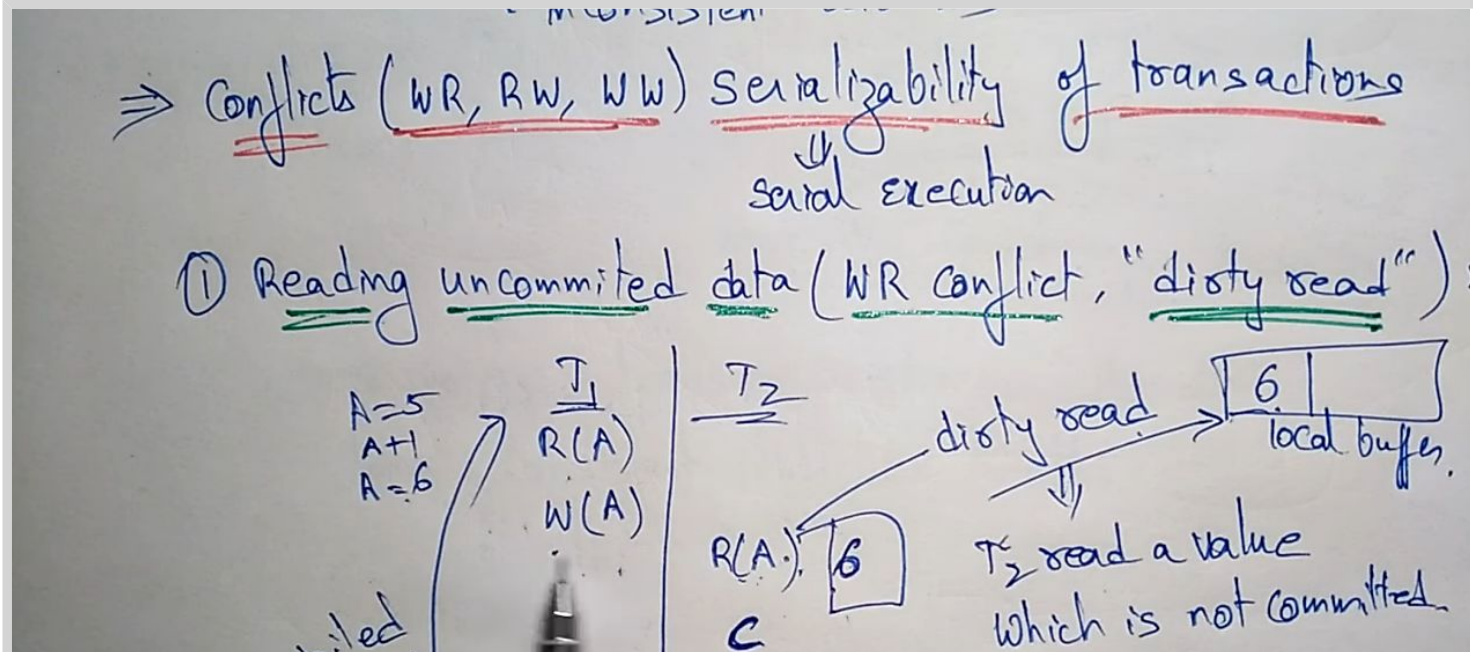

⇒ Conflicts (WR, RW, WW) serializability of transactions
serial execution
① Reading uncommited data (WR conflict, "dirty read")
A=5
A+1
A=6
T1
R(A)
W(A)
R(B)
W(B)
T2
R(A) 6
C
dirty read
6
local buffer
T2 read a value
which is not committed

Abort

12:36

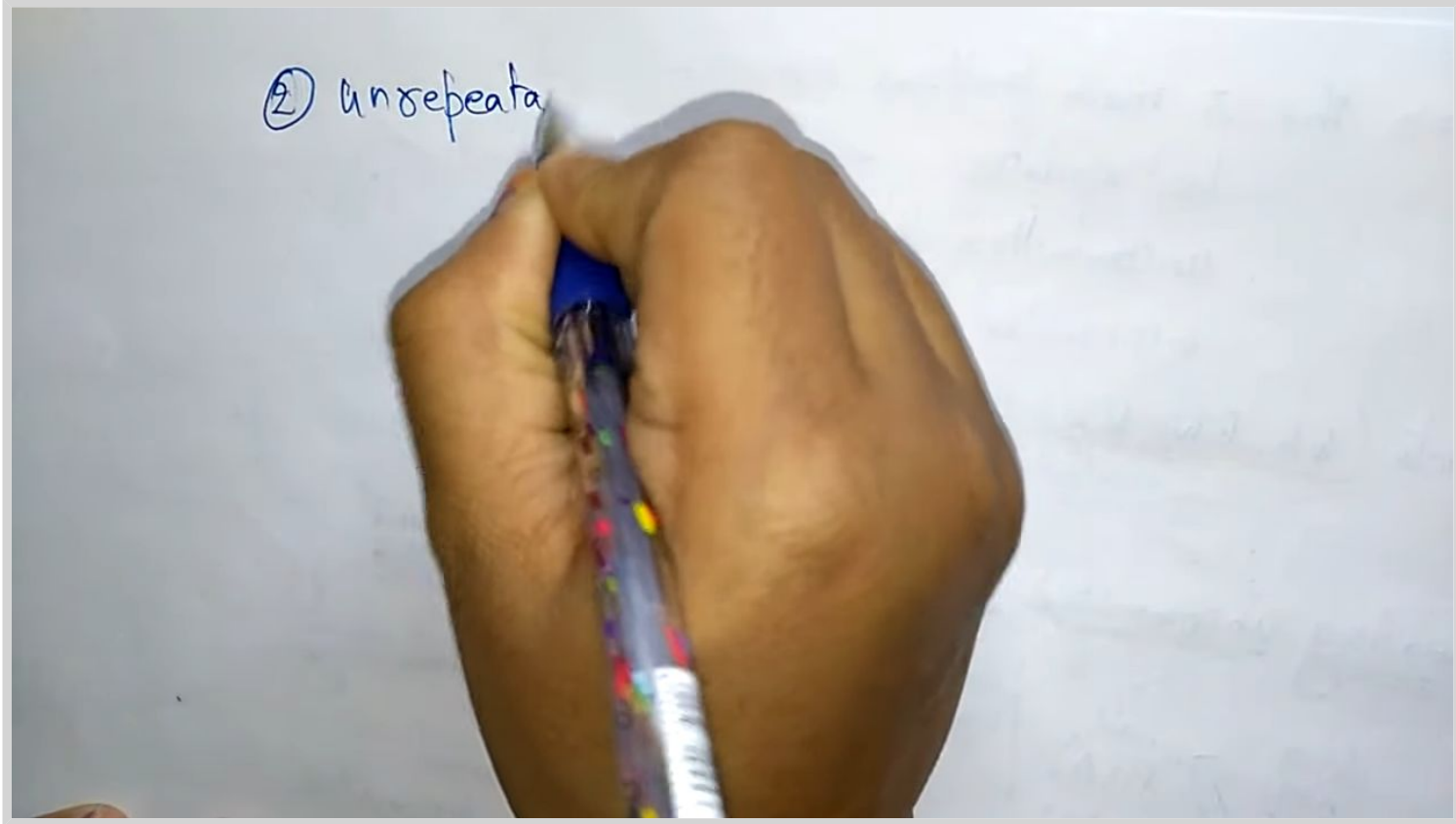


12:39

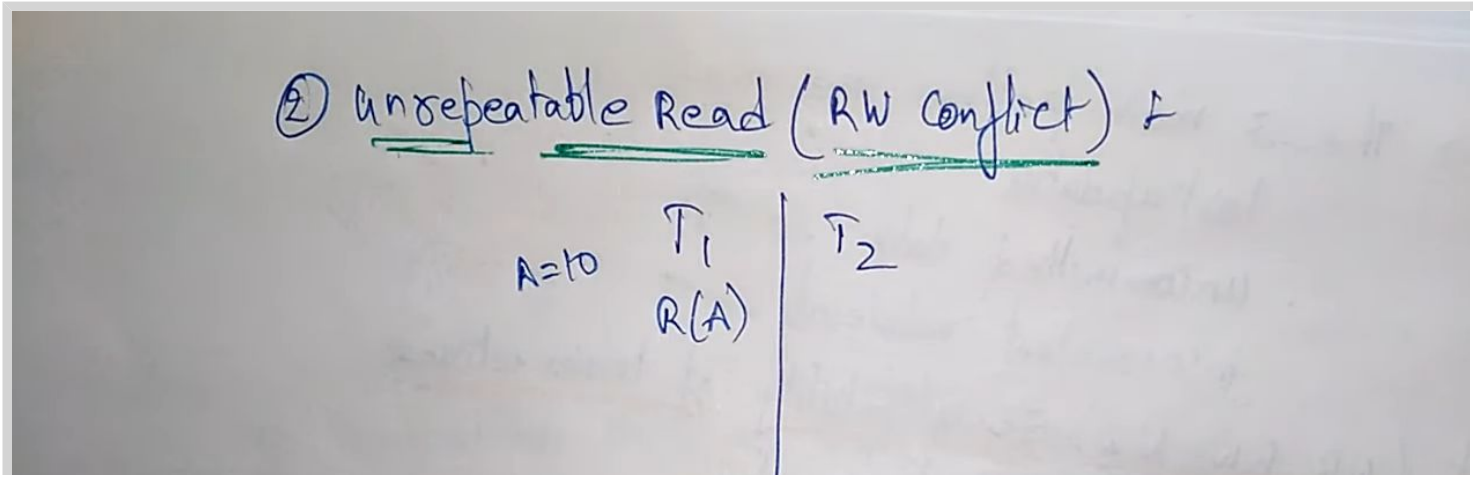


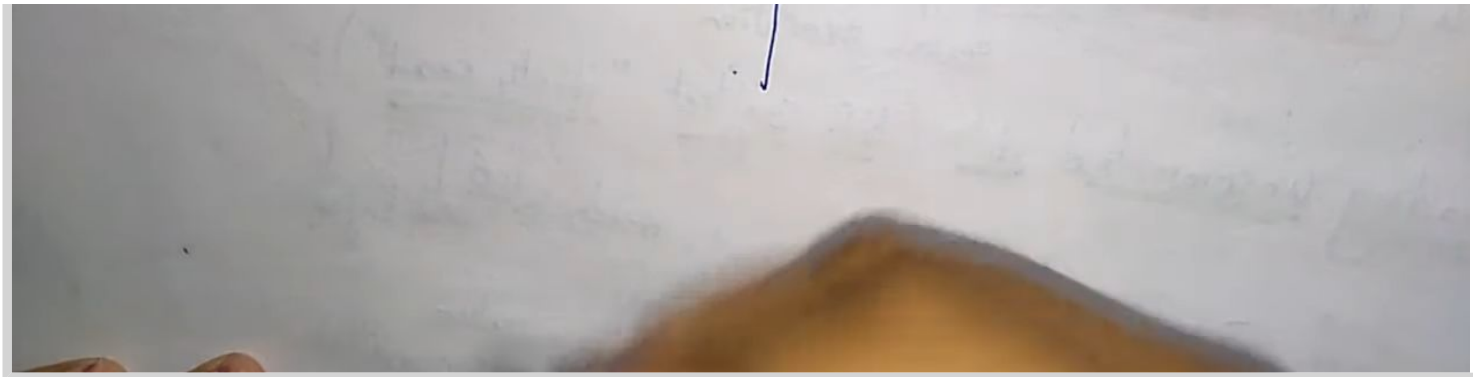

unchanged
rollback
B)

13:32

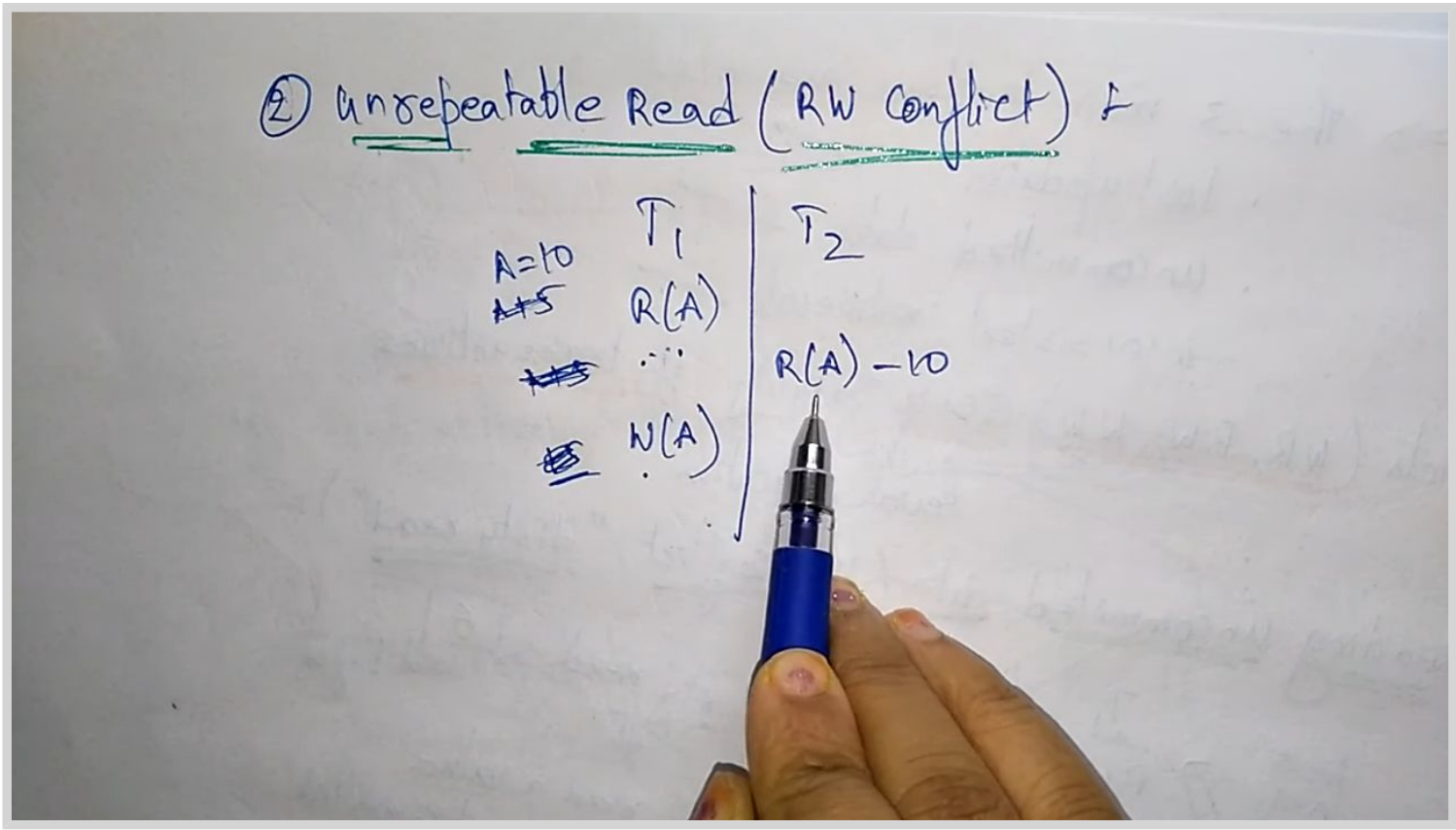

(2) unrepeata

13:58

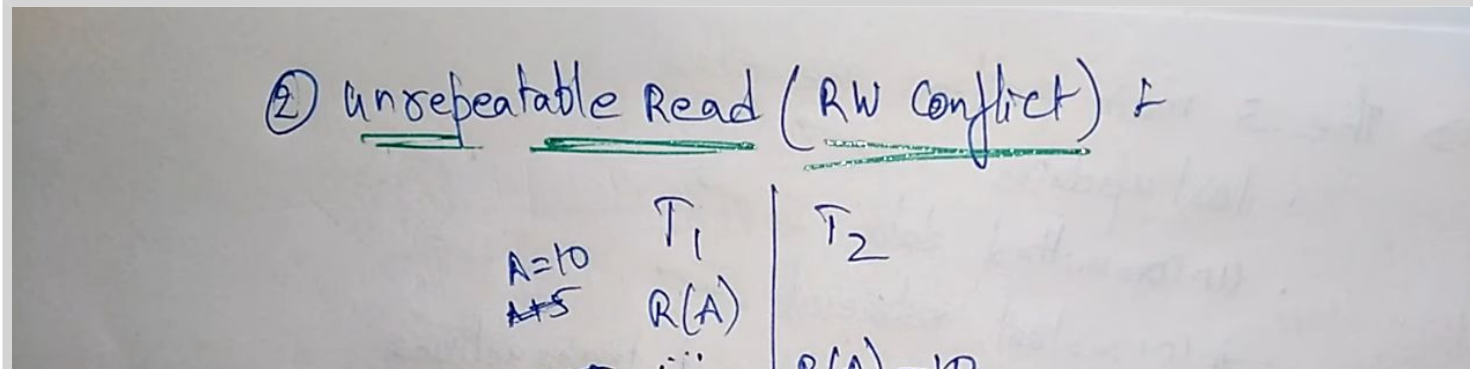

(2) unrepeatable Read (RW Conflict)
A=10
T1
R(A)
T2

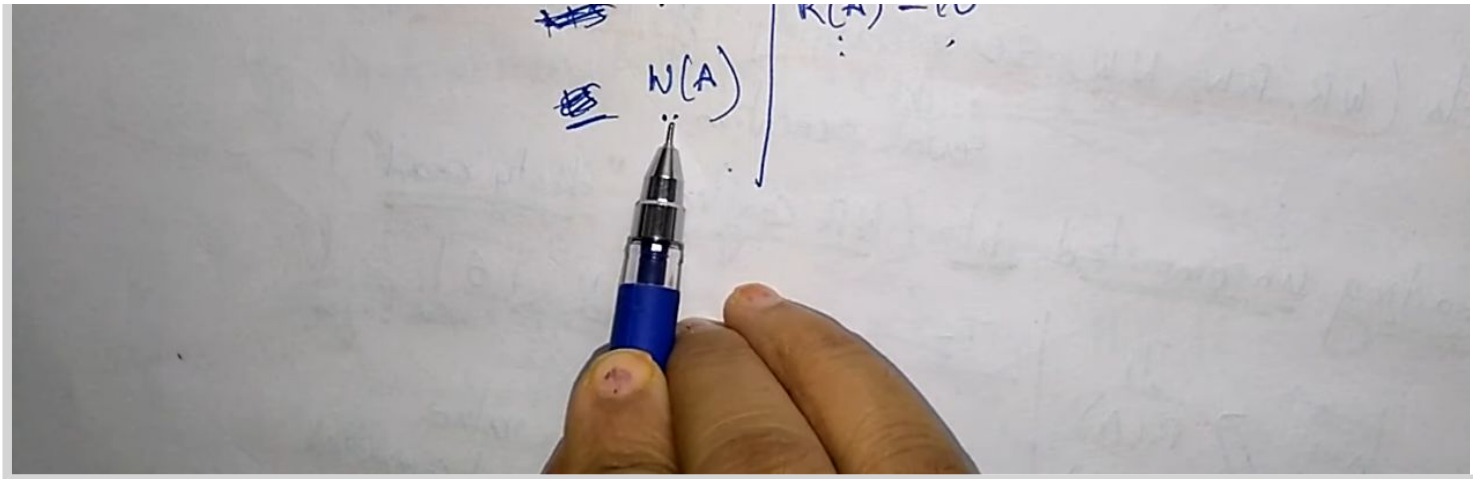


14:45

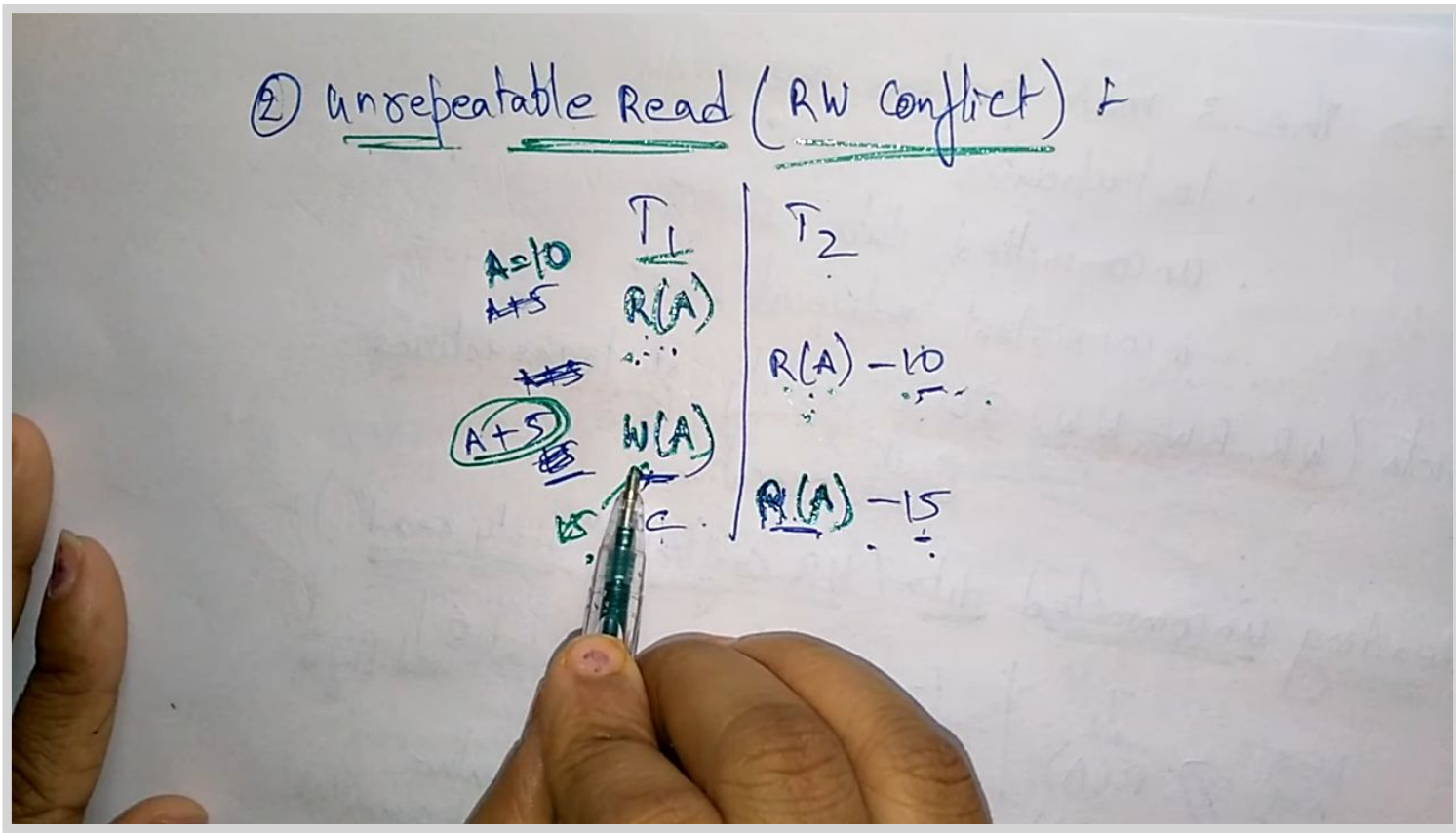

② Unrepeatable Read (RW Conflict)
T1
T2
A=10
R(A)
R(A) - 10
W(A)

14:54

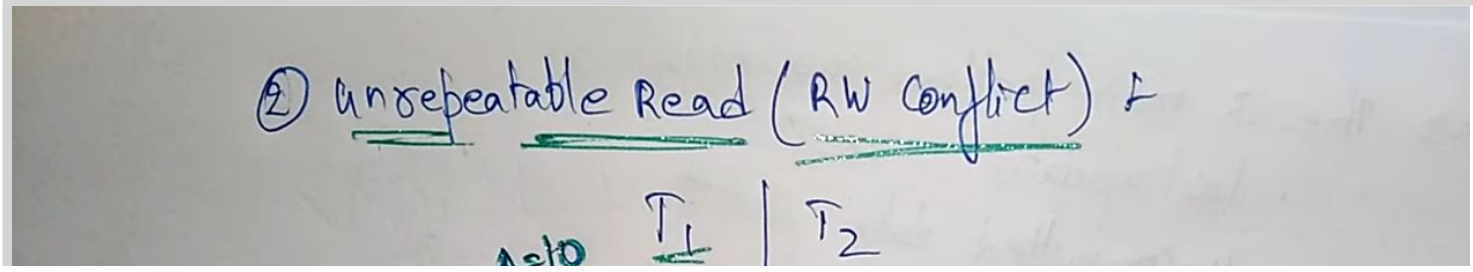

② Unrepeatable Read (RW Conflict)
T1
T2
A=10
R(A)

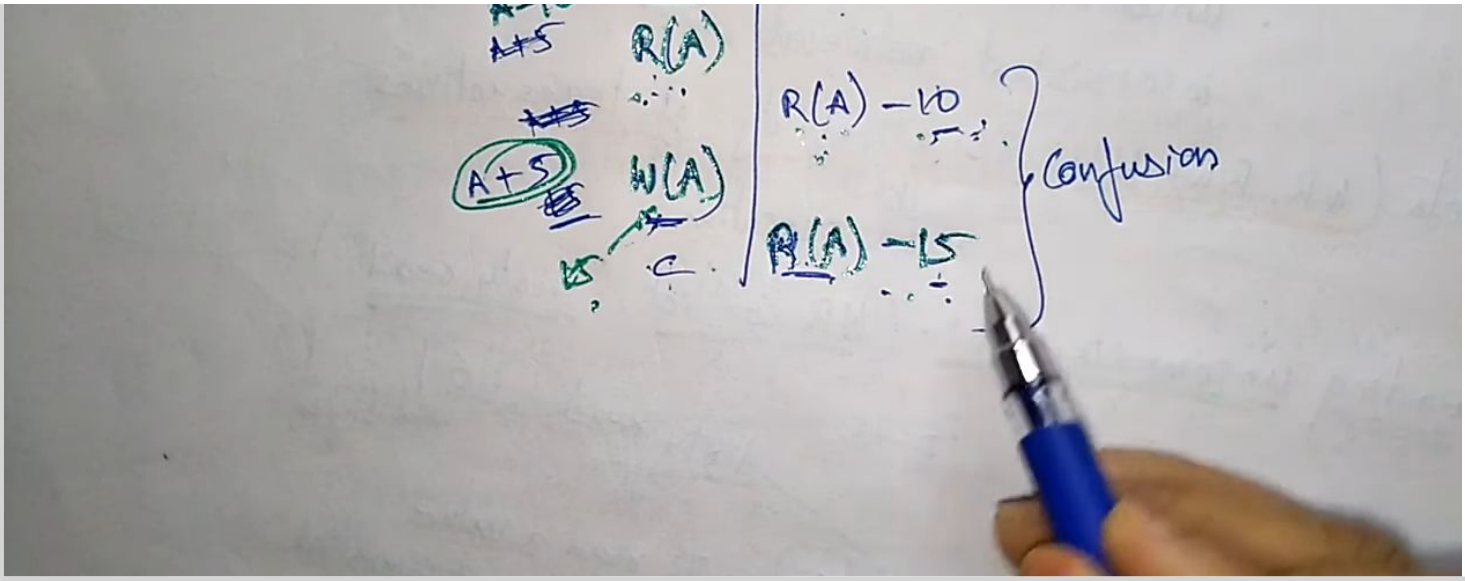

W(A)

16:11

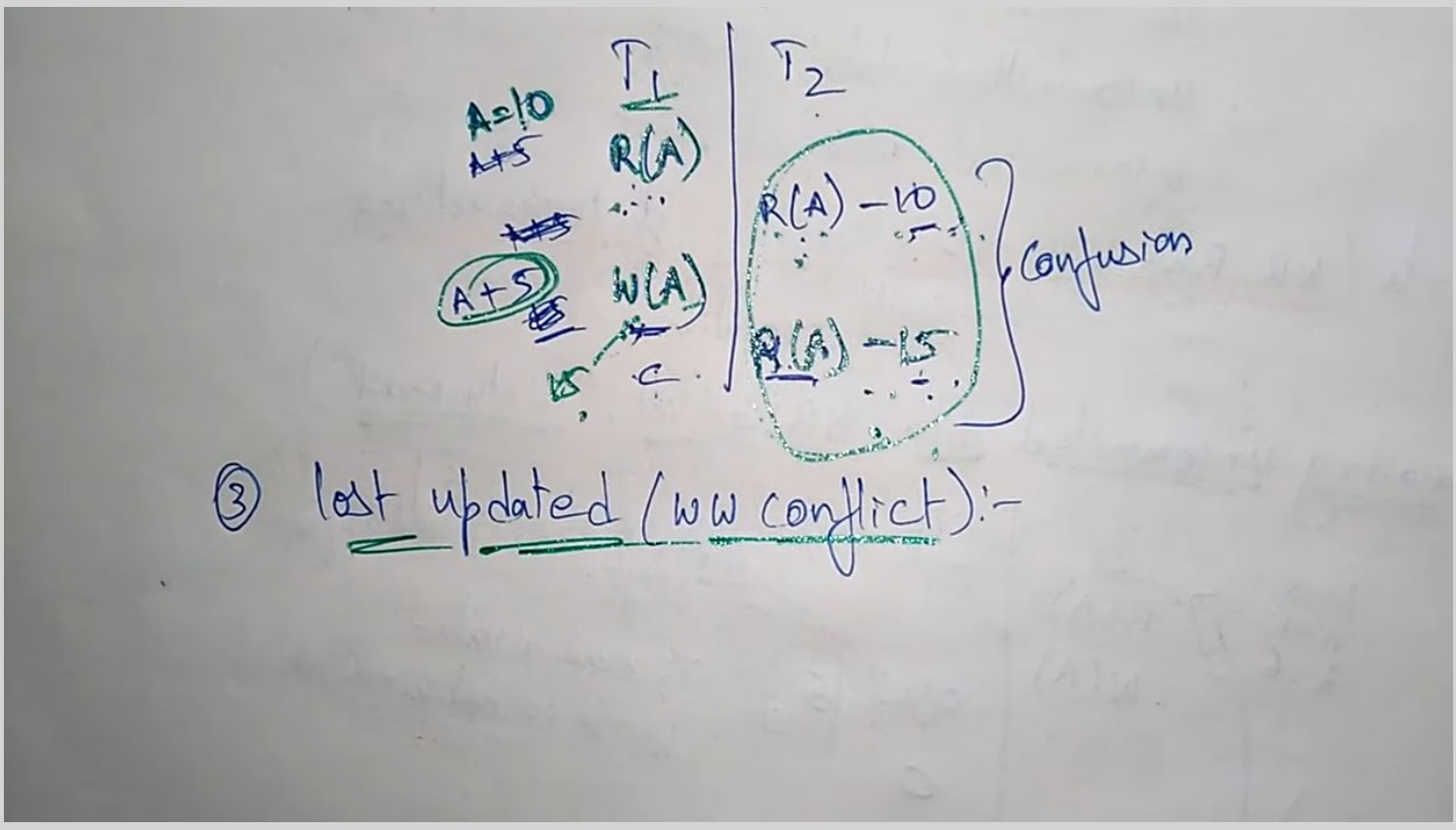

② Unrepeatable Read (RW Conflict)
T1
T2
A=10
A+5
R(A)
R(A) - 10
A+5
W(A)
R(A) - 15

16:32

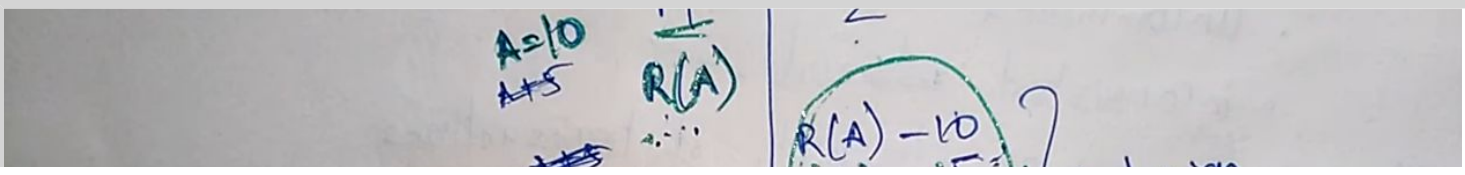

② Unrepeatable Read (RW Conflict)
T1
T2

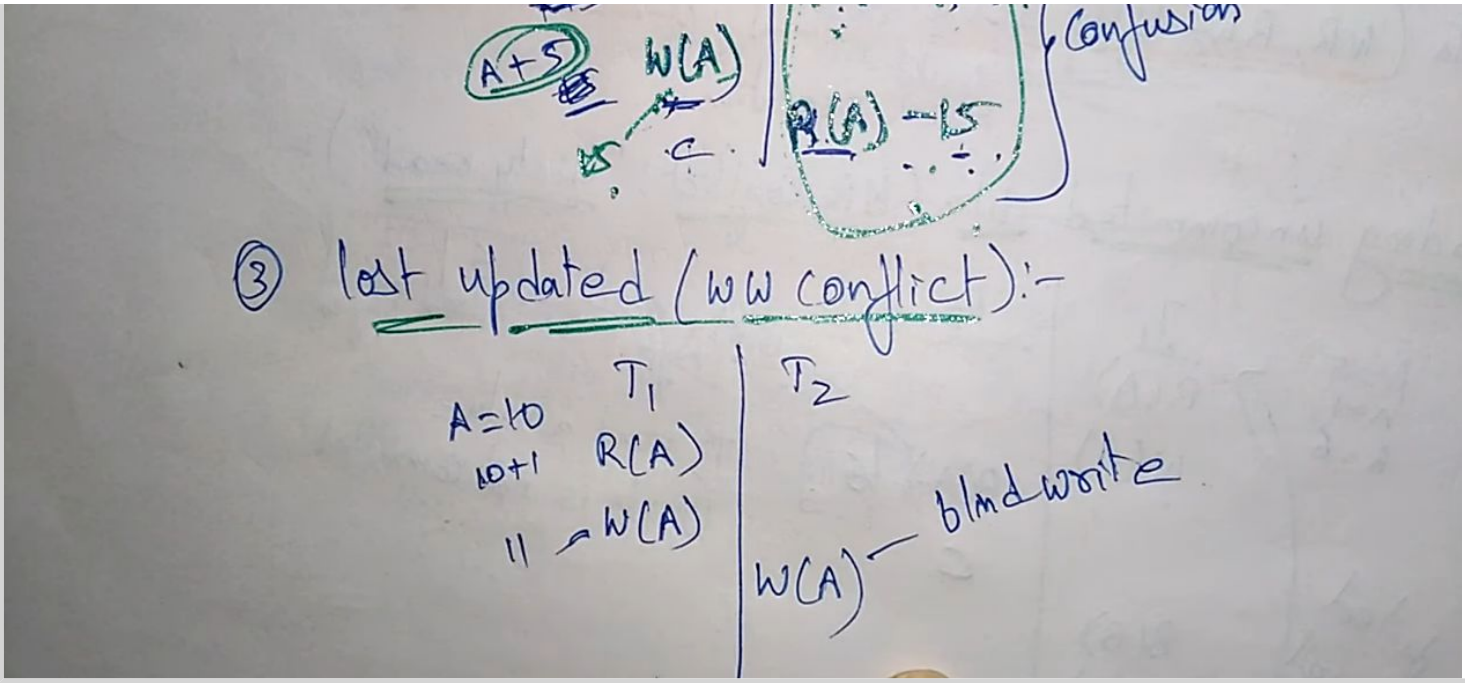

R(A)
A+5
W(A)
R(A) - 10
R(A) - 15
Confusion

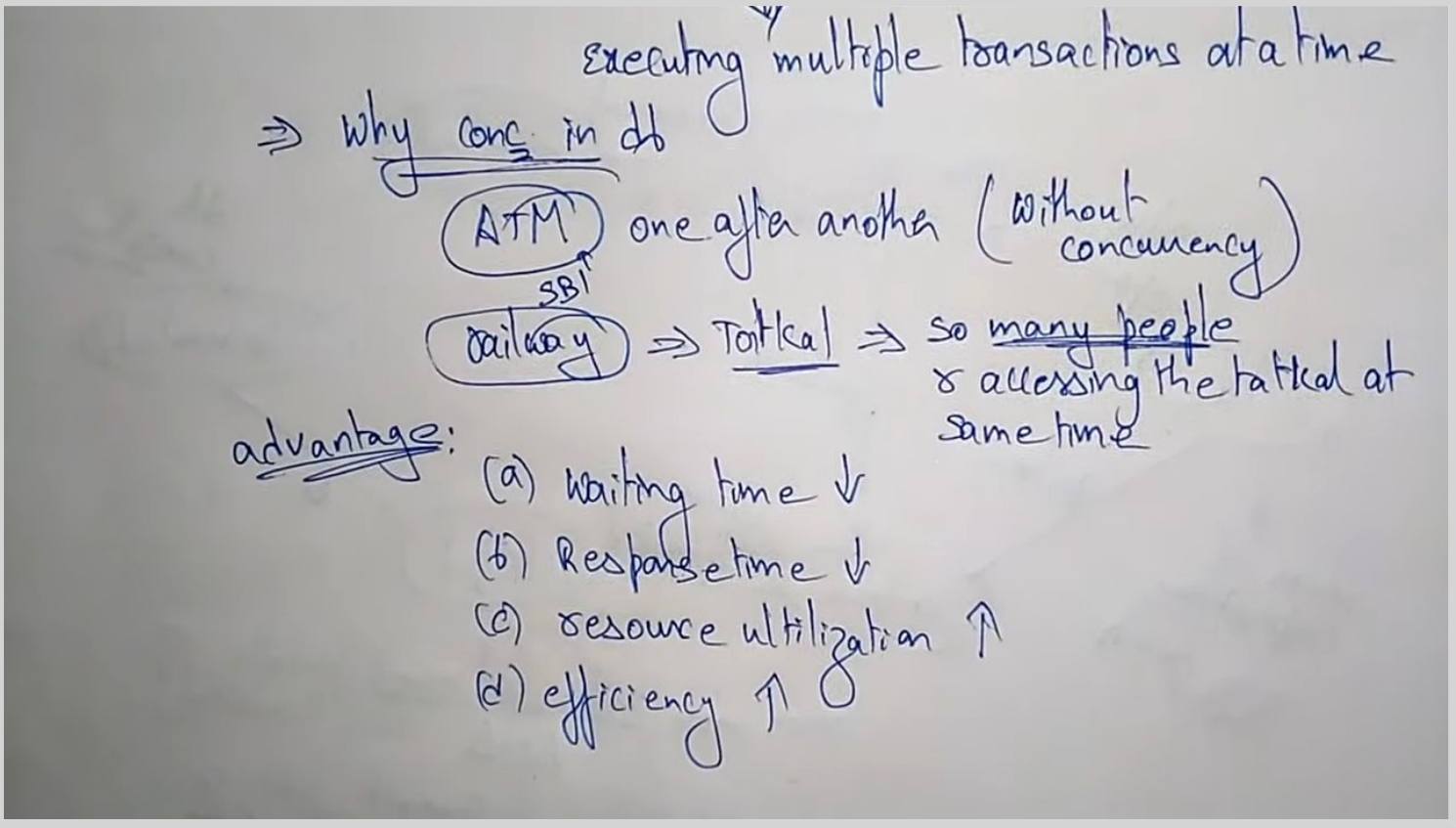
17:28


A=10
T1
T2
R(A)
A+5
W(A)
R(A) - 10
R(A) - 15
Confusion
③ lost updated (ww conflict):-

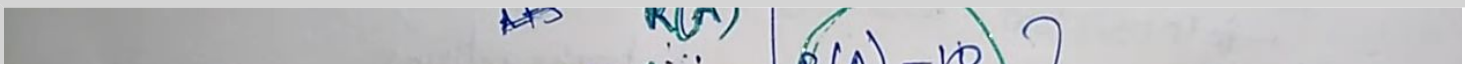
18:06

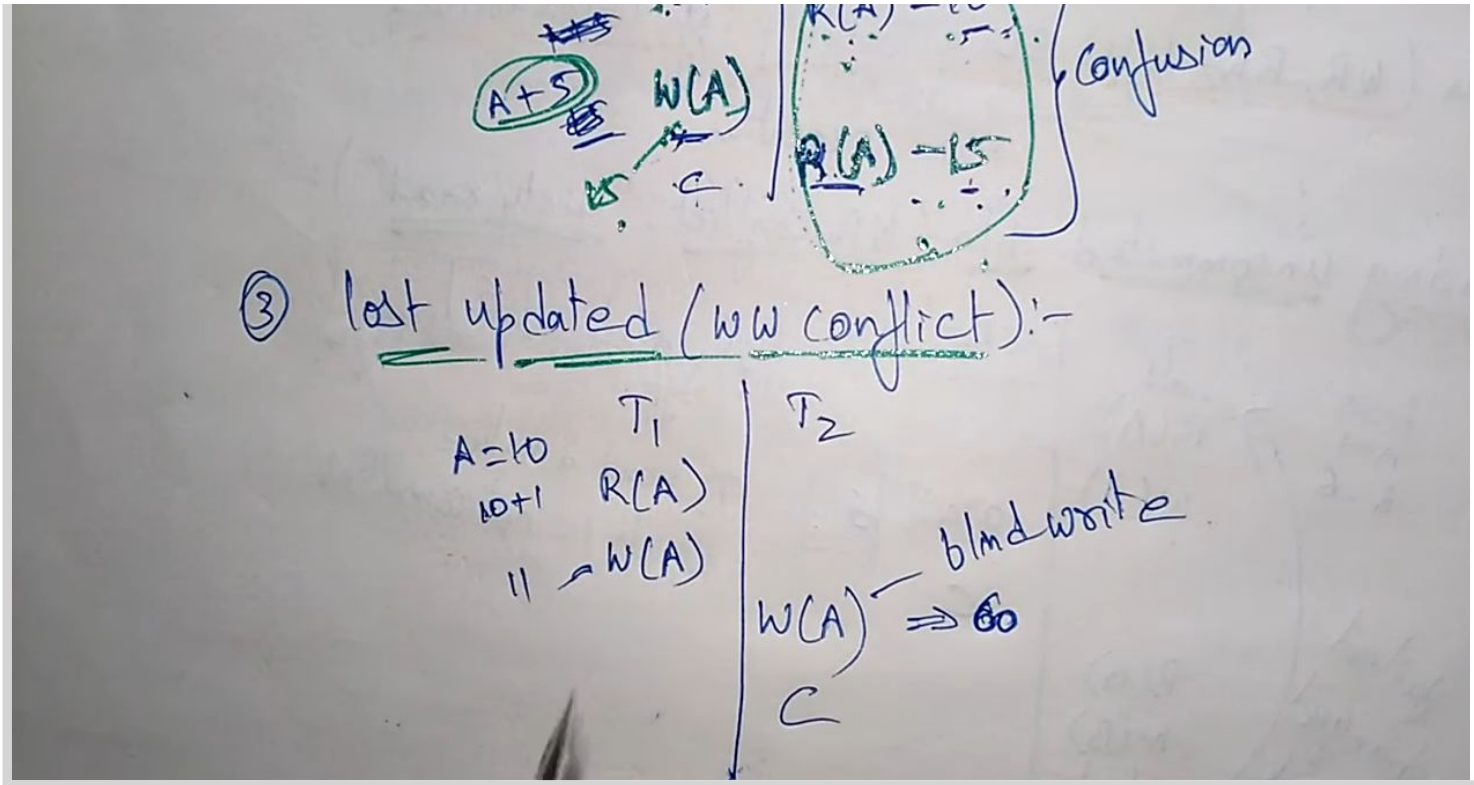

A=10
R(A)
A+5
R(A) - 10


A+5
W(A)
R(A) -15
Confusion
③ lost updated (ww conflict):-
T1
T2
A=10
10+1
R(A)
W(A)
W(A) — blind write

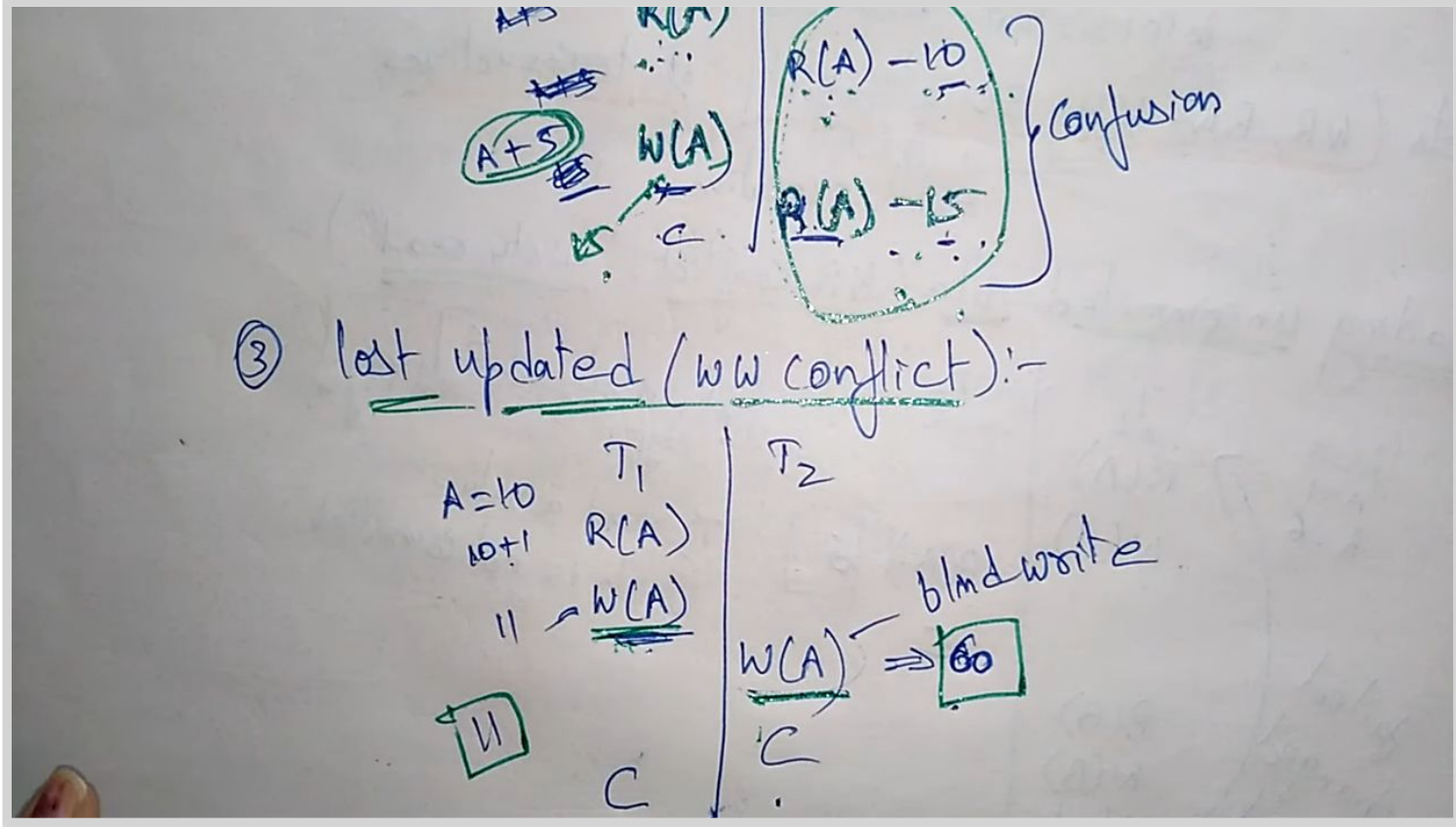

executing multiple transactions at a time
⇒ why conc in db
ATM
SBI
one after another (without concurrency)
Railway
⇒ Tatkal ⇒ so many people
& accessing the tatkal at same time
advantage:
(a) waiting time ↓
(b) Response time ↓
(c) resource utilization ↑
(d) efficiency ↑

18:19

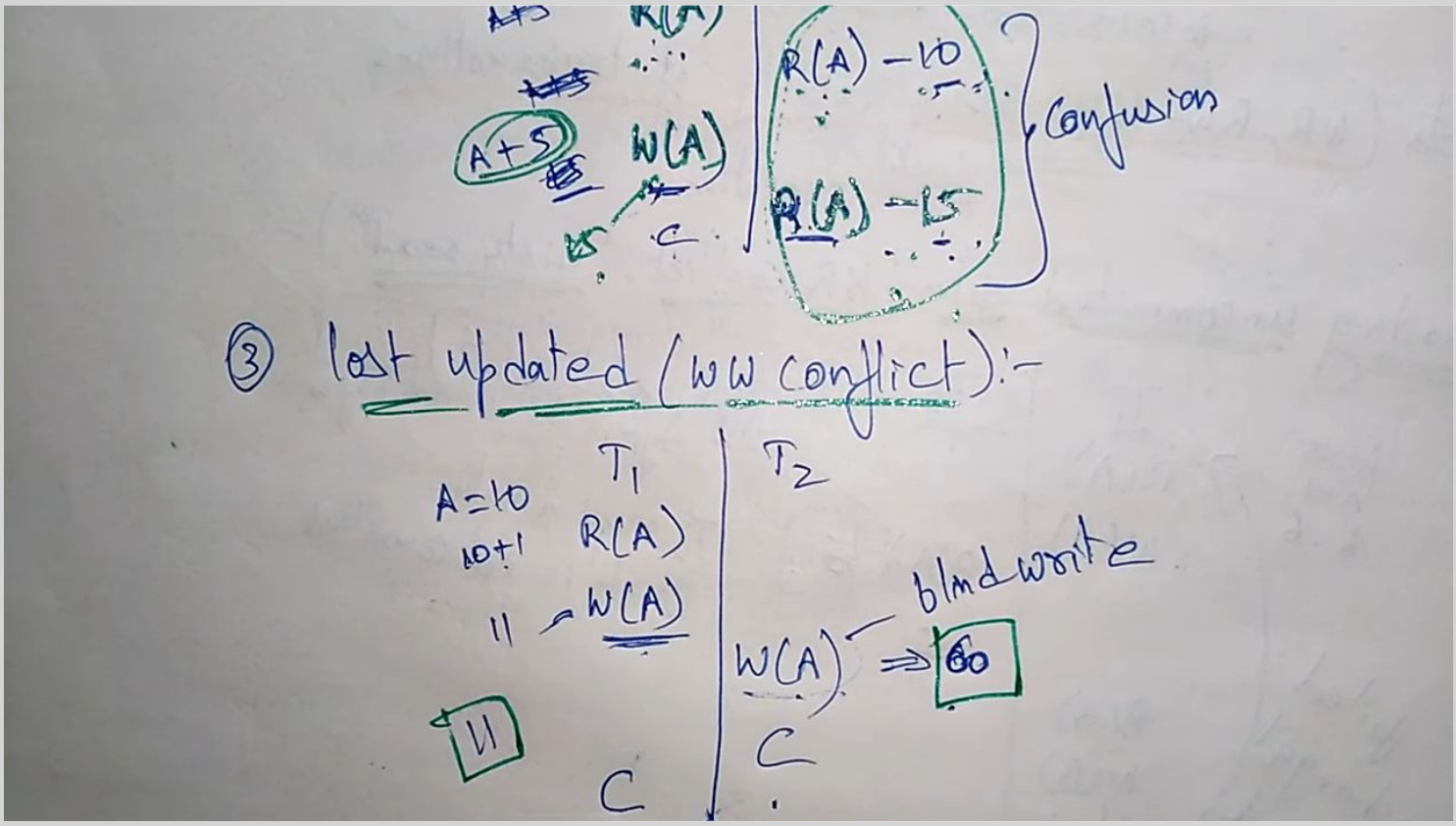



A+5
W(A)
C
R(A) - 10
R(A) - 15
Confusion
③ last updated (ww conflict):-
T1
T2
A=10
10+1
R(A)
11
W(A)
blind write
W(A) ⇒ 60
C

19:05


A+5
W(A)
C
R(A) - 10
R(A) - 15
Confusion
③ last updated (ww conflict):-
T1
T2
A=10
10+1
R(A)
11
W(A)
blind write
W(A) ⇒ 60
11
C
C

18:53



③ lost updated (ww conflict):-

	T_1	T_2
A=10		
10+1	R(A)	
11	W(A)	
		W(A) — blind write ⇒ 60
11		
	C	C