Lock-Based Protocols

- *A lock is a mechanism to control concurrent access to a data item*
- Data items can be locked in two modes:
  - a) *exclusive (X) mode:*
    - Data item can be *both read as well as written*
    - **X-lock** is requested using **lock-X** instruction
  - b) *shared (S) mode:*
    - Data item can *only be read*
    - **S-lock** is requested using **lock-S** instruction
- A transaction can unlock a data item Q by the **unlock**(Q) Instruction
- Lock requests are made to the concurrency-control manager by the programmer
- *Transaction can proceed only after request is granted*

State of the lock	Lock request type	
	Shared	Exclusive
Shared	Yes	No
Exclusive	No	No

o

η

Schedule

T_1	T_2

locking protocol

conflict serializibar

lock-α (A)

W(A.)

unlock(A)

lock-α (A)

.W (A)

unlock(A)

lock-$\S$(A)

R(A)

unlock(A)
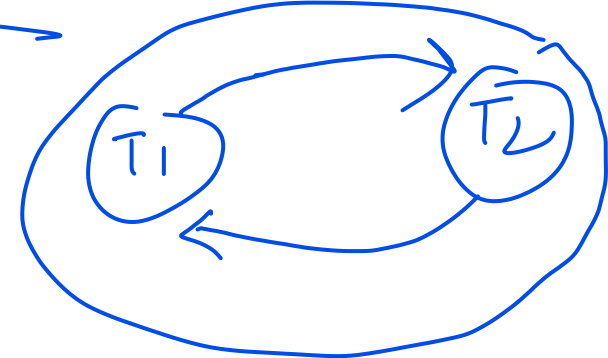
cycle

Two-Phase Locking Protocol

- Two-phase locking with lock conversions:
  - First Phase: *(Growing phase)*
    - ▷ can acquire a lock-S on item
    - ▷ can acquire a lock-X on item
    - ▷ can convert a lock-S to a lock-X (upgrade)
  - Second Phase: *(Shrinking Phase)*
    - ▷ can release a lock-S
    - ▷ can release a lock-X
    - ▷ can convert a lock-X to a lock-S (downgrade)

T_1

lock-R(A)

N(A)

S.P. unlock(A

lock-X(R)

R(R)

Two phase locking protocol:

- After shrinking phase goowing Phase is not possible

- Goowing phase is compulsory shrinking phase is optional

- Two phase locking protocol are alway conflict serializable but vice-versa is not true

- Does not ensure recoverability
- Cascading rollback
- Deadlock

Does not ensure recoverability

T1	T2
lock-X(A)	
R(A)	
W(A) -	
unlock-(A)	
	lock-X(A)
	R(A)
	W(A)
	unlock-(A)
	C2
C1	

Growing phes

S·P

4.

S·P

Cascading rollback

T_1 T_2

$lock-\alpha(A)$

$R(A)$

$\omega(A)$

$unlock(A)$

$lock-\alpha(A)$: L_1

$R(A)$ $10J$

$\omega(A)$ $10J+10$

$unlock(A$

C_1

C_2

Deadlock

$$S$$

$$T_1 \qquad T_2$$

$$
\text{i/p} \left[
\begin{array}{l}
\text{lock} \to \alpha (A)
\end{array}
\right.
\qquad
\text{lock} \to \alpha (B) \Big] \quad \text{i/p}
$$

$$R(A) \qquad\qquad R(B)$$

$$W(A) \qquad\qquad W(B$$

want.

$T_1 \wedge$ update
the value of B

(A) T_2 want to
update
the value of A
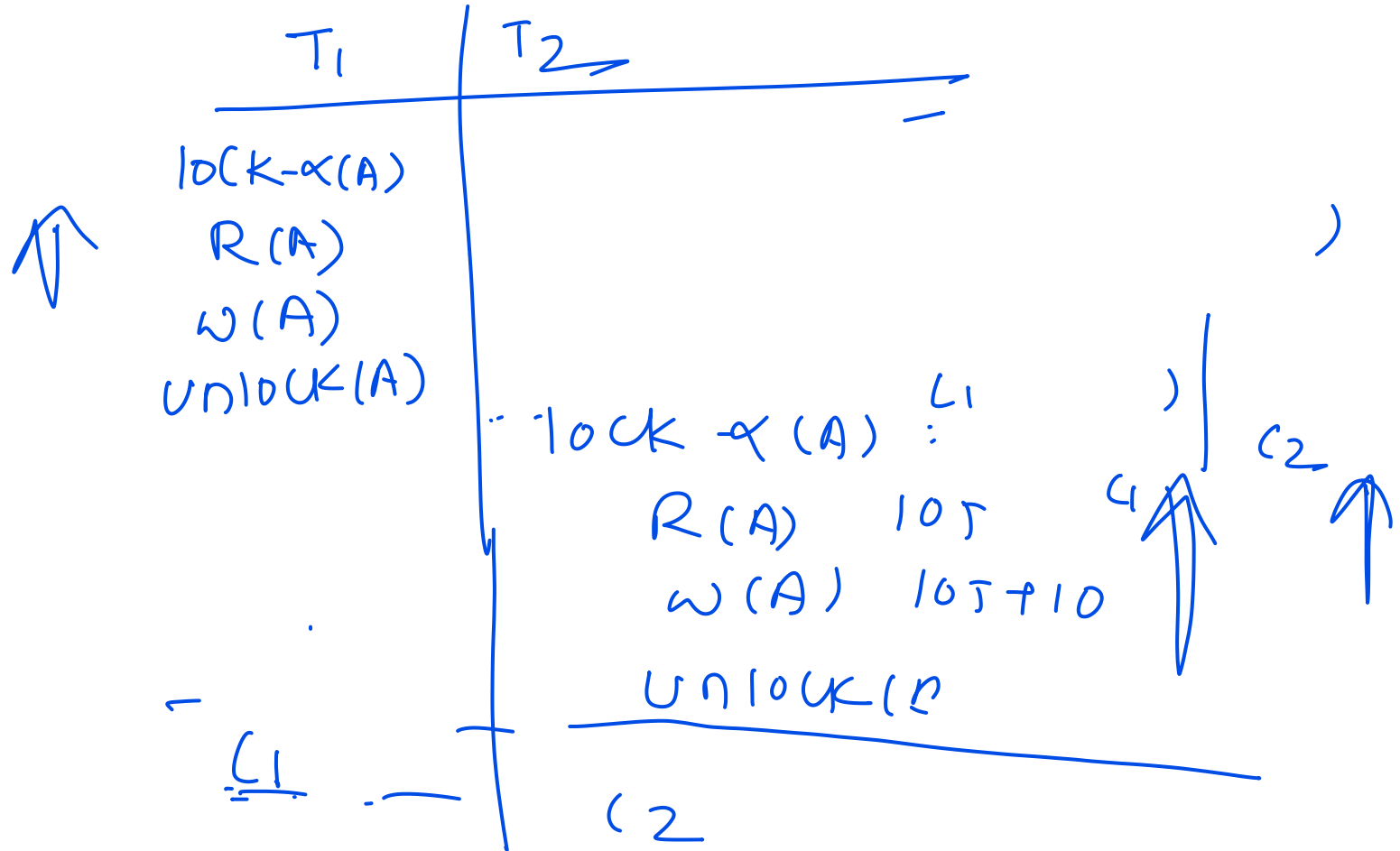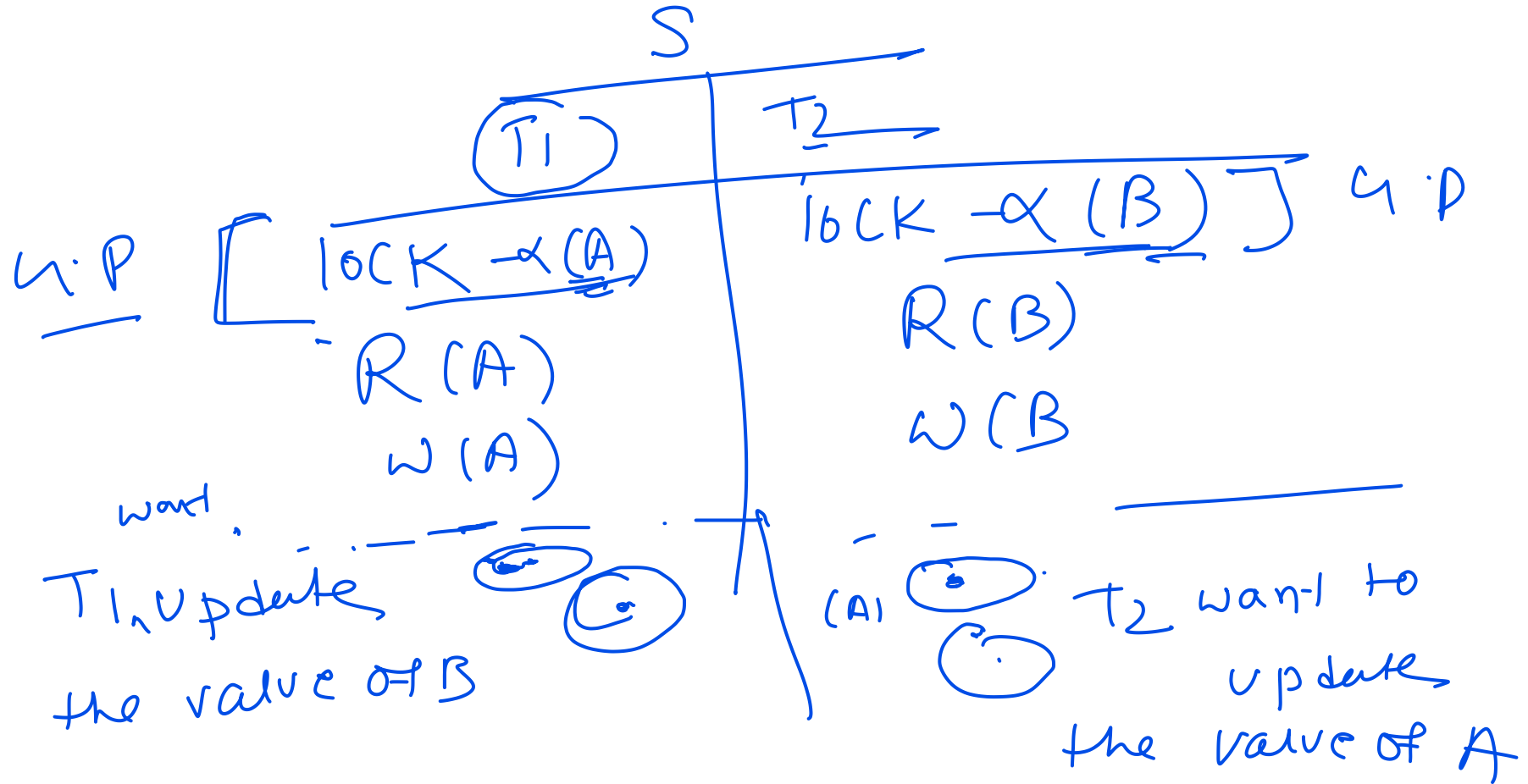
7) Schedule **S** is as given: w2(P), w1(P), w3(P), w2(Q), w1(Q)
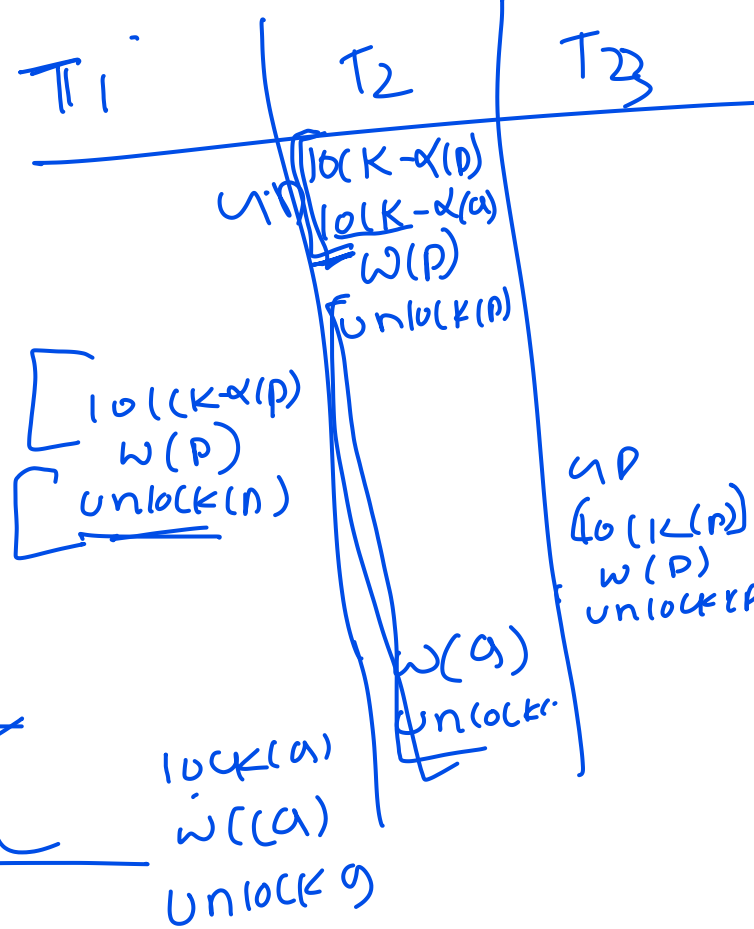
Consider the statements:

- All Conflict serializable schedules are 2-P lockable α
- The given schedule **S** is Conflict serializable
- The given schedule is 2-P lockable

The number of correct statements is

$T_1 \propto 2PL$

$T_3 \rightarrow 2PL$

$T_2 \rightarrow 2PL$

T_1 | T_2 | T_3

lock-α(P)
lock-α(Q)
W(P)
unlock(P)

4P [lock-α(P)
     W(P)
S·P [unlock(n)

4P
lock(P)
W(P)
unlock(P)

W(Q)
unlock

S·D [lock(Q)
      W(Q)
      unlock Q

Calend.

previ

Timestamp