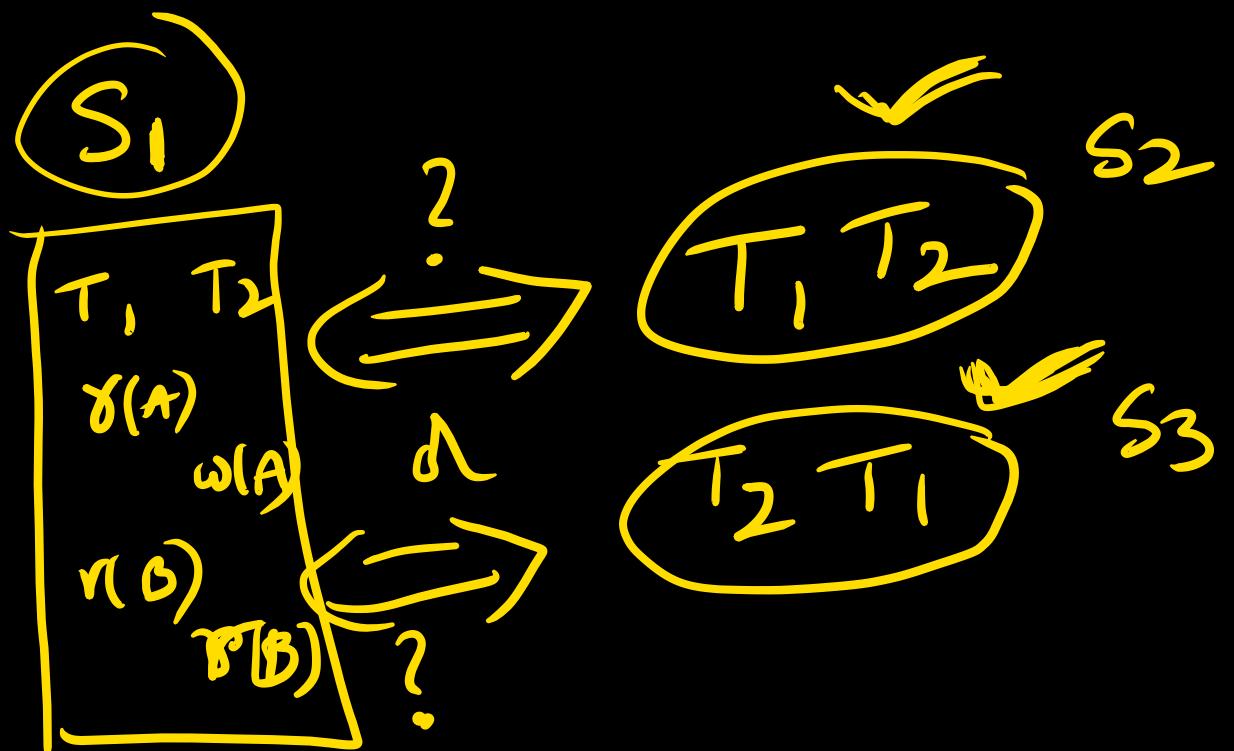


Classification of schedules based on serializability:

- Conflict ~~serializable~~ schedule
 - view ~~serializable~~ schedule
- Goal of this concept is to test if given schedule
is serializable or not.

$$S_1 \stackrel{?}{\equiv} S_2$$

$$S_1 \stackrel{?}{\equiv} S_3$$



To learn about Conflict serializability we must learn the below things:

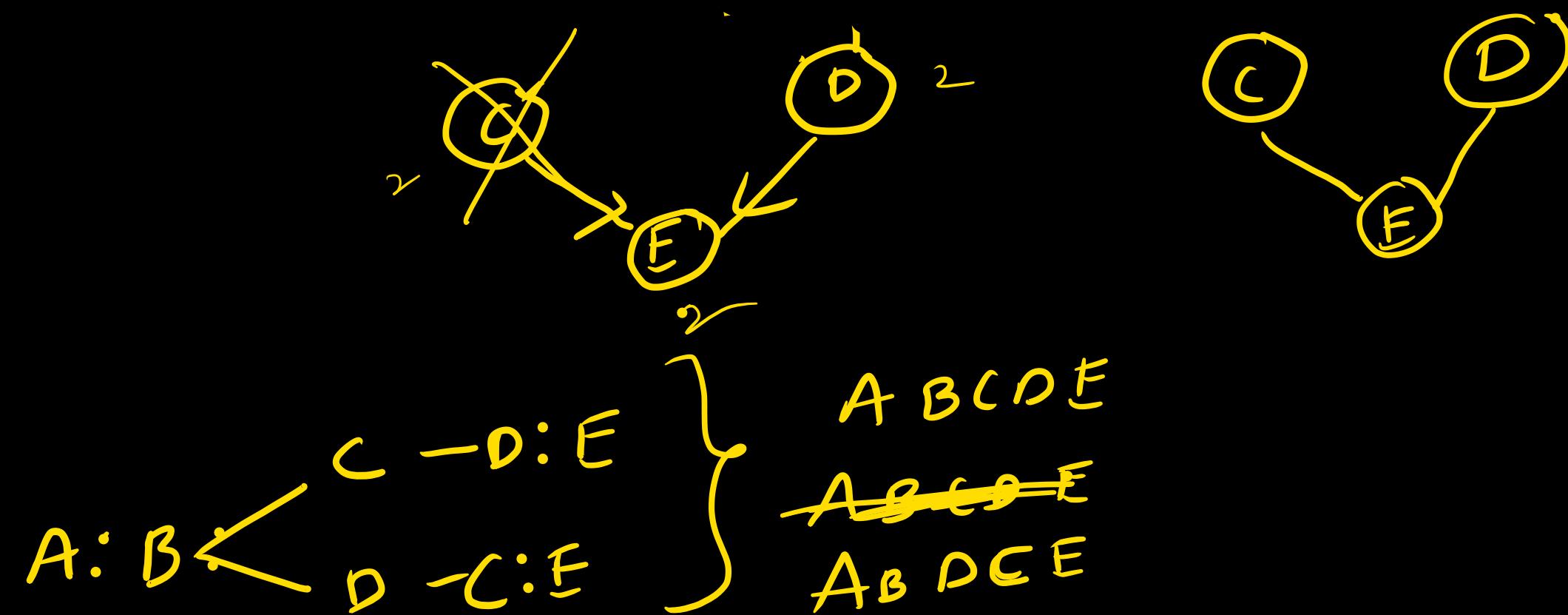
- Topological order
 - Conflict pairs
 - Precedence graph
 - Conflict equal schedules
- } are important in understanding what conflict serializability is.

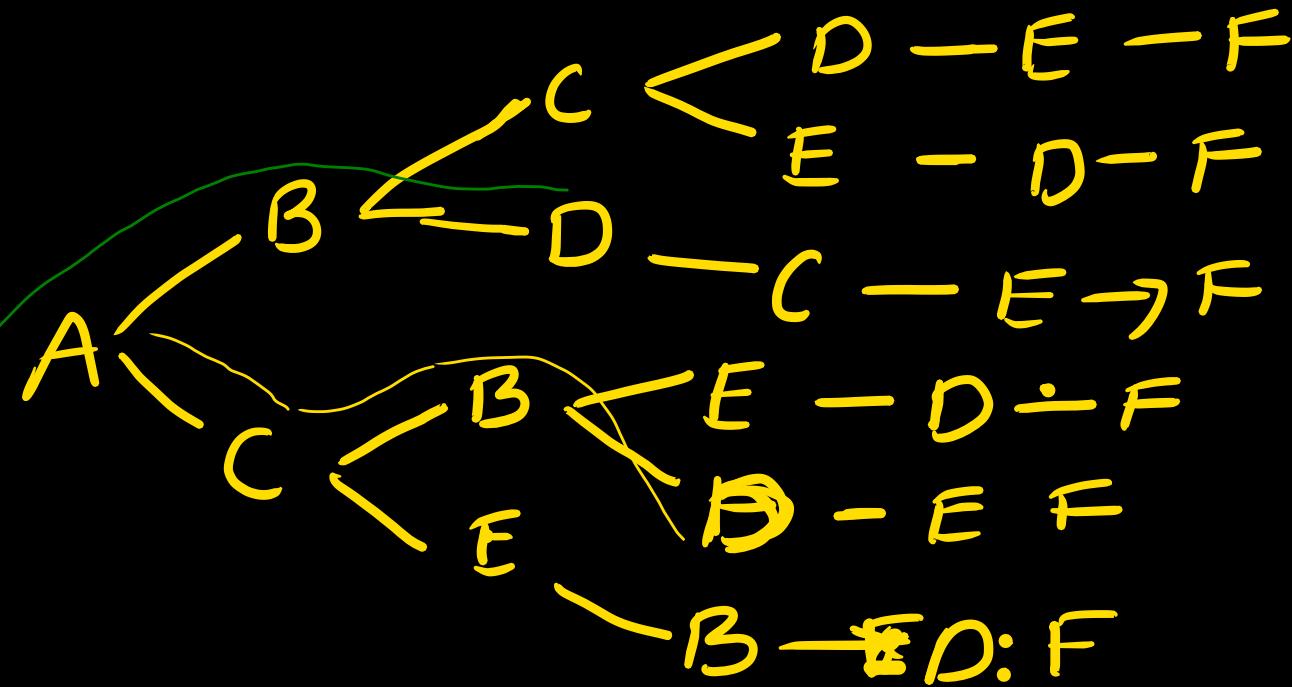
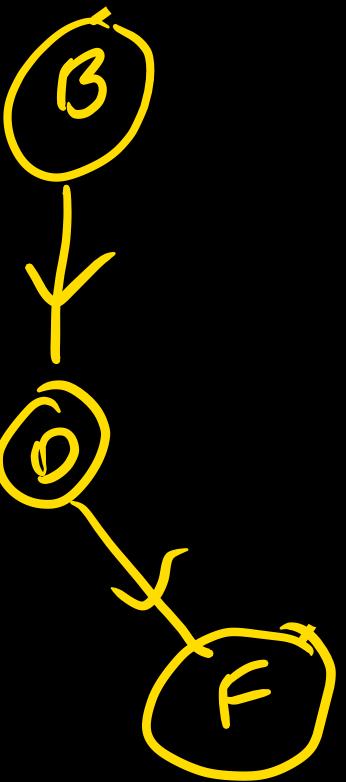
Topological Order

This is one of the graph traversal algorithm which is applied only on directed acyclic graph (DAG).

- Algo:
- 1) visit vertex (v) whose indegree is "0" and delete " v " from graph(G).
 - 2) Repeat ① for all vertices of graph.

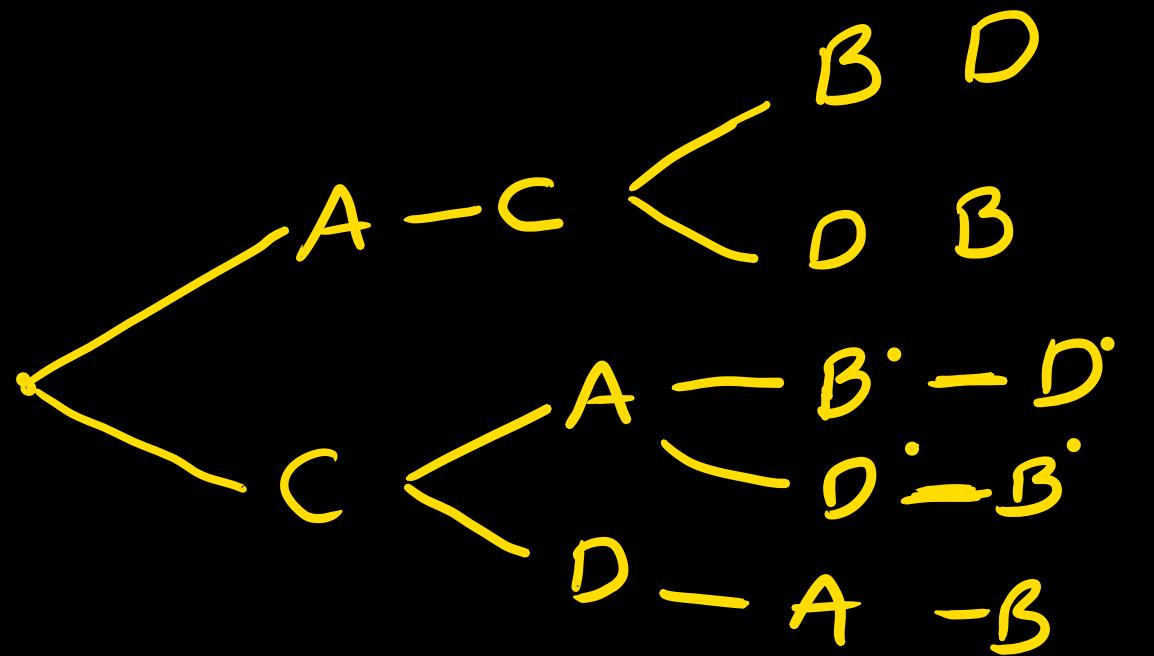
→ Find all topological orders & follow DAG





- 1 ABCDEF
- ~~2 ABCDECDF~~
- 2 ABCEDEF
- 3 ABDCCEF
- 4 ACBEBEDEF
- 5 ACBDEEF
- 6 ACEBEDF

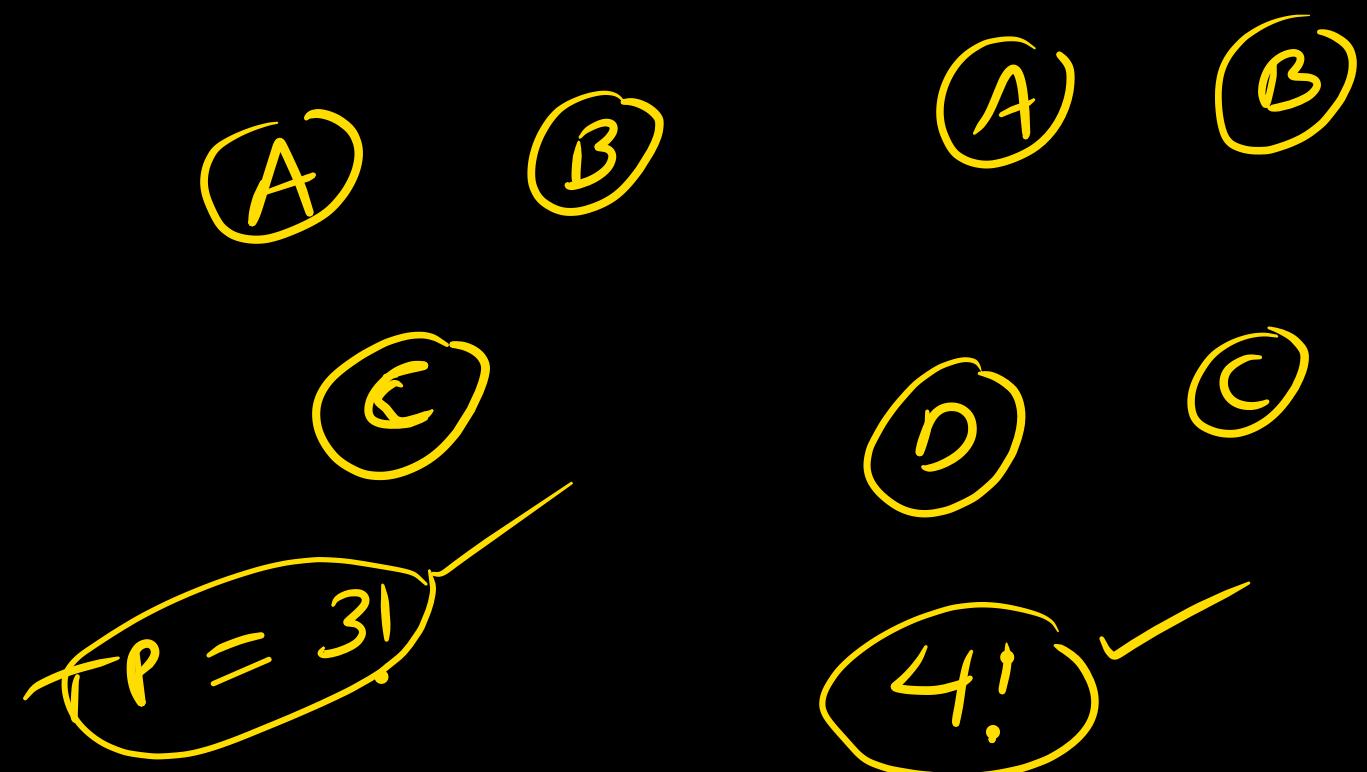
$A \rightarrow B$



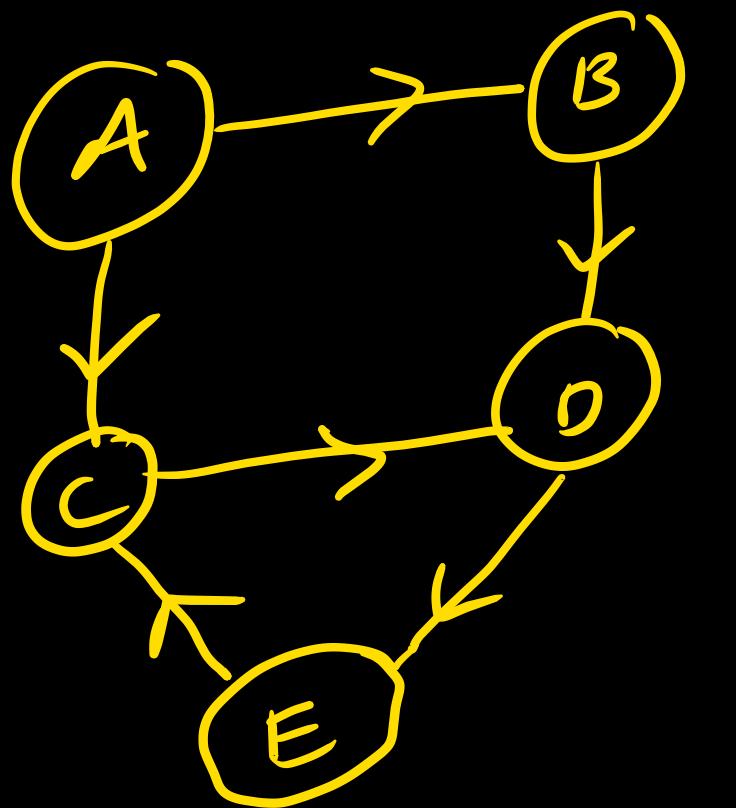
$ACBD$
 $ACDB$
 $CABD$
 $CADB$
 $CDAB$

Null graph: Graph with n vertices and 0 edges.

of topological orders for null graph with ' n ' vertices:



No of topological orders for the below graph.



\Rightarrow it is cyclic.

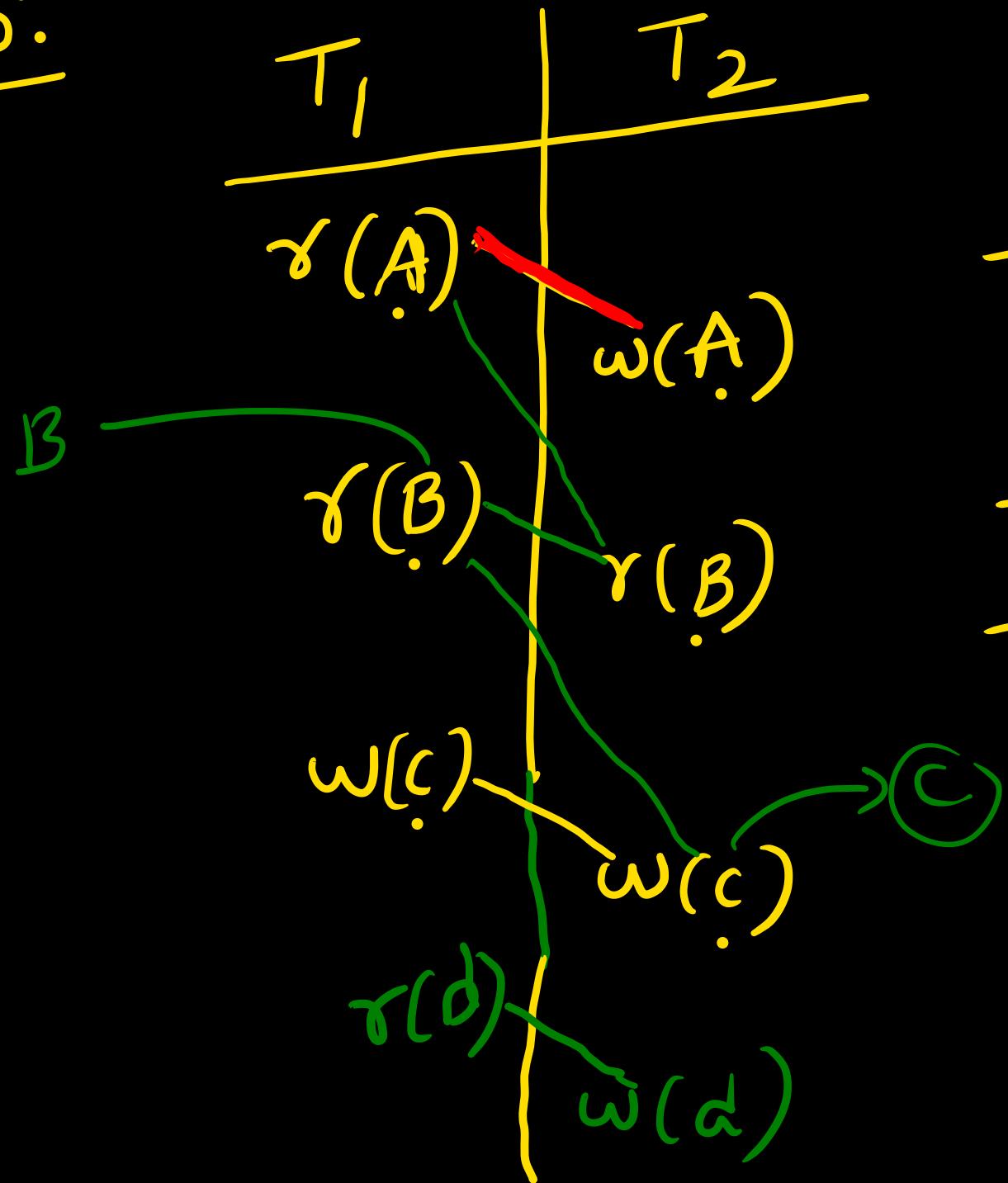
Conflict pair

Pair of operations from schedule 'S' are conflict pairs if
at least one of them is write up

and

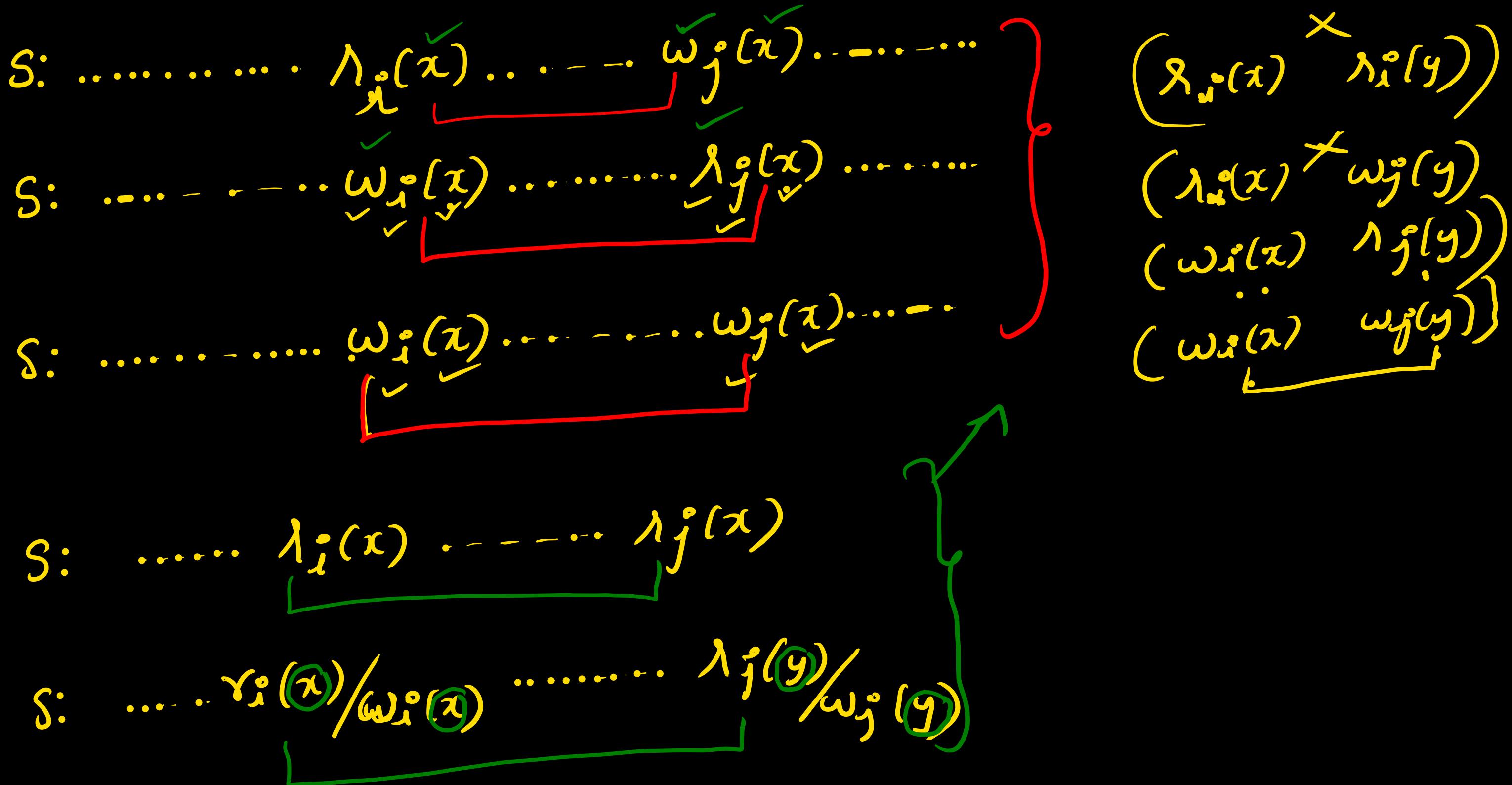
- 1) over same data item
- 2) from different transaction
- 3) at least one of them is write up

S.:



- Conflict

- pair of operations must be from different transactions.
- Data item should be same.
- At least one of them should be write

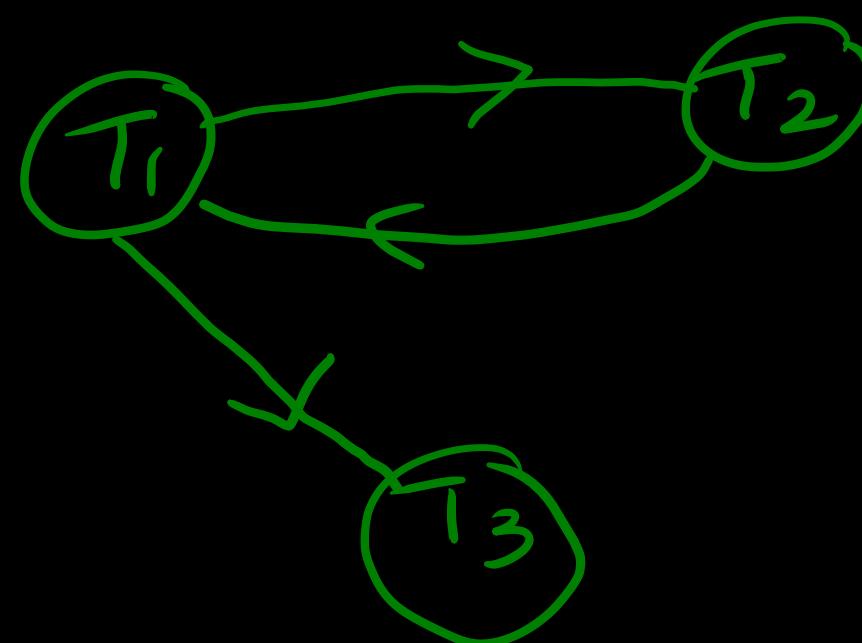
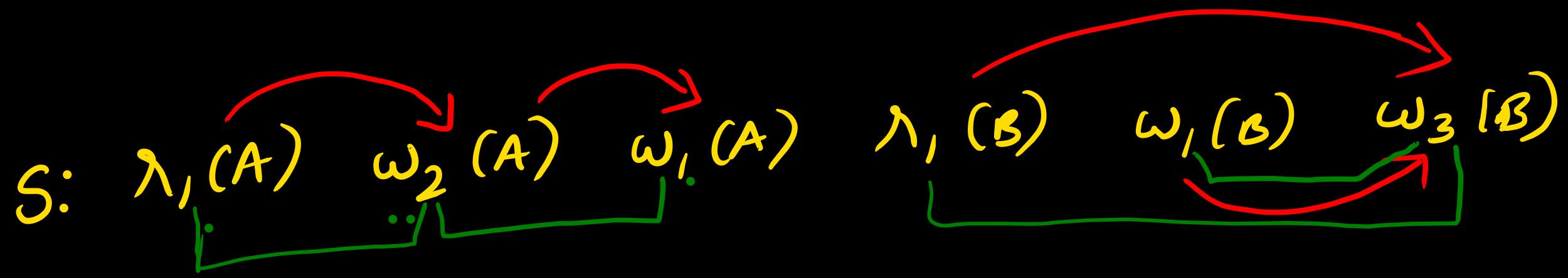


Precedence graph:

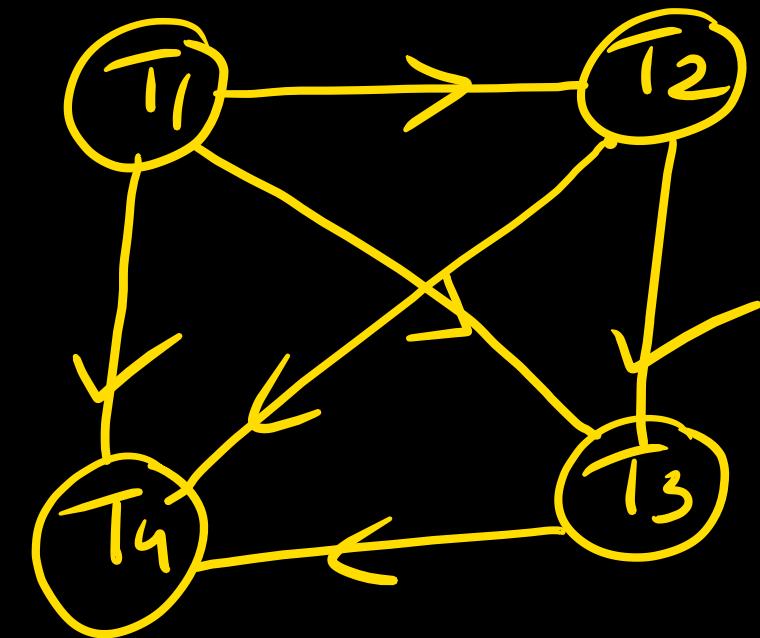
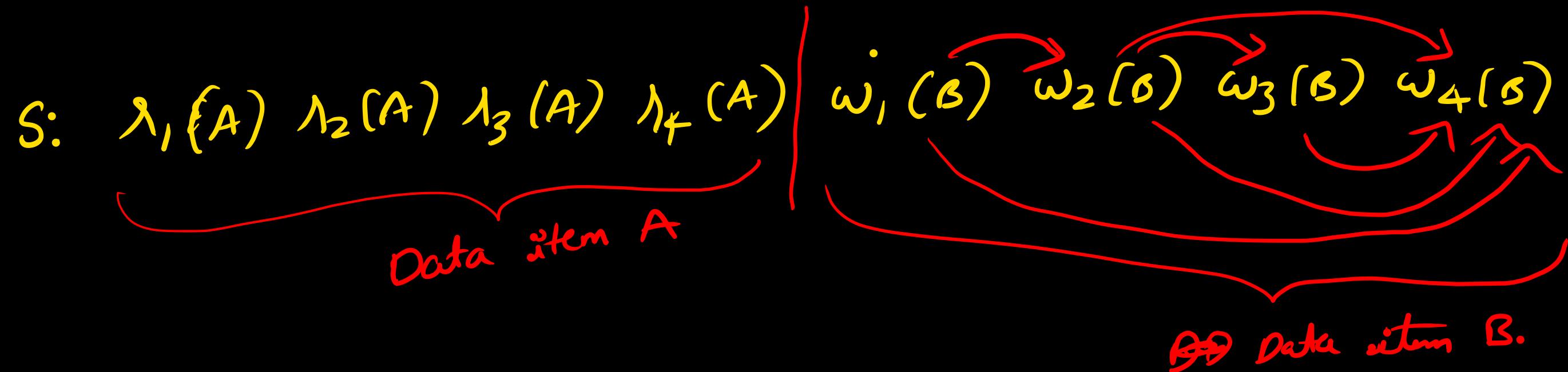
Precedence graph of schedule (s)

Vertices (V): Transactions of schedule

Edge (E): Conflict pairs precedences (def.)



based
precedence graph.

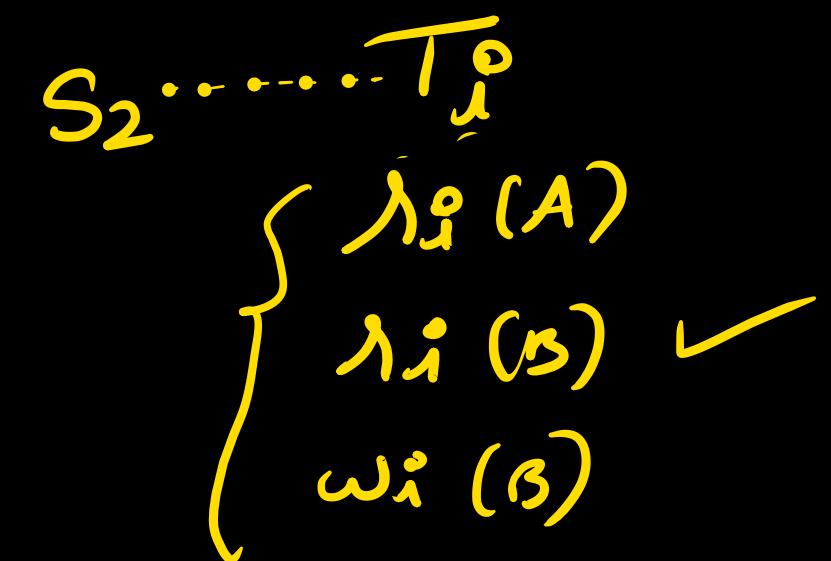
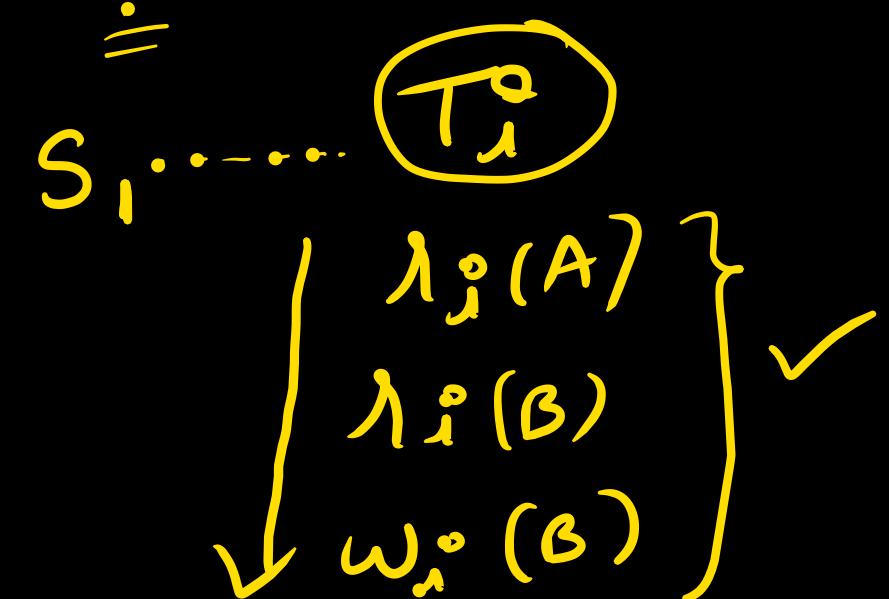


Conflict equal schedules :-

s_1 s_2 are conflict equal if

- ① Each transaction of s_1 must be exactly same as transaction of s_2 .

of s_2 :



② Every conflict pair precedence of S_1 must be exactly same precedence in S_2

Ex:

$S_1: \gamma_1(A) \gamma_2(A) \gamma_3(A) \omega_1(B) \omega_2(S) \omega_3(B)$

$S_2: \gamma_1(A) \omega_1(B) \gamma_3(A) \gamma_2(A) \omega_2(B) \omega_3(B)$

①

$S_1: T_1$
 $\gamma_1(A)$
 $\omega_1(B)$

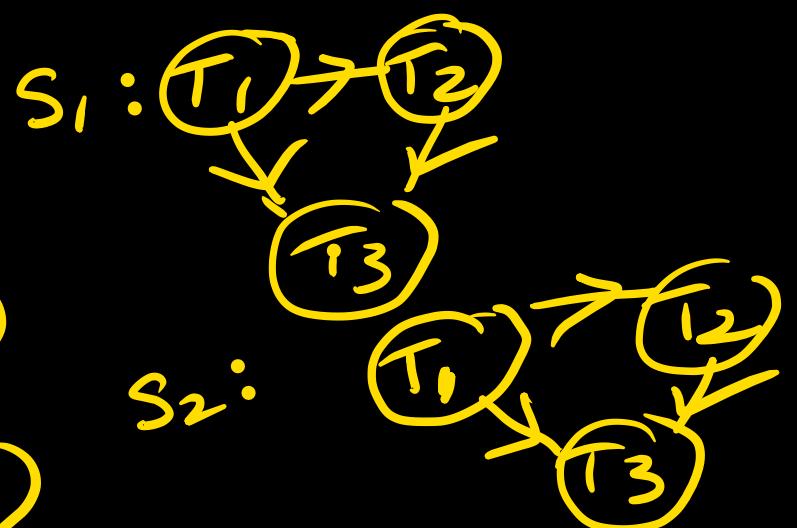
$S_2: T_1$
 $\gamma_1(A)$
 $\omega_1(B)$

$S_1: T_2$
 $\gamma_2(A)$
 $\omega_2(B)$

$S_2: T_2$
 $\gamma_2(A)$
 $\omega_2(B)$

$S_1: T_3$
 $\gamma_3(A)$
 $\omega_3(B)$

$S_2: T_3$
 $\gamma_3(A)$
 $\omega_3(B)$



$S_1: \tau_1(A) \omega_1(A) \tau_2(A) \tau_2(B) \tau_1(B) \omega_1(B)$

$S_2: \tau_1(A) \omega_1(A) \lambda_1(B) \omega_1(B) \tau_2(A) \lambda_2(B)$

① $\tau_2 \rightarrow \tau_1$

Not conflict equd.

② $\tau_1 \rightarrow \tau_2$

Ex:

$S_1: \gamma_1(A) \gamma_2(A) \gamma_3(A) \gamma_4(A) \omega_1(B) \omega_2(B) \omega_3(B) \omega_4(B)$

$S_2: \gamma_1(A) \omega_1(B) \gamma_4(A) \gamma_2(A) \gamma_3(A) \omega_2(B) \omega_3(B) \omega_4(B)$

all precedences will be same.

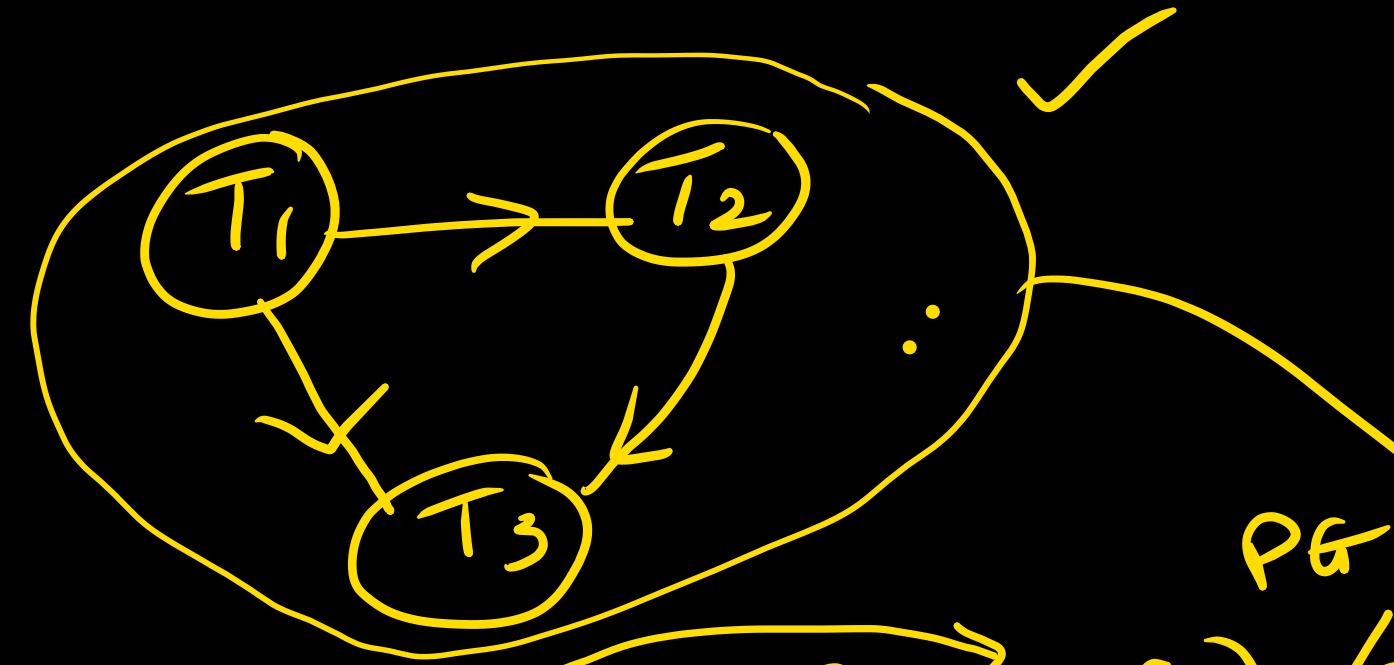
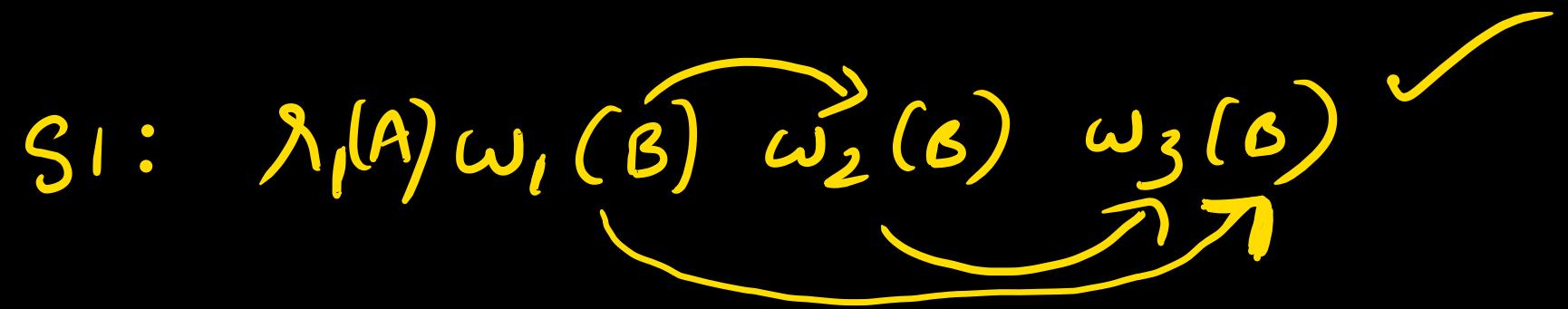
S_1 and S_2 conflict equal schedules.

5 min Met.

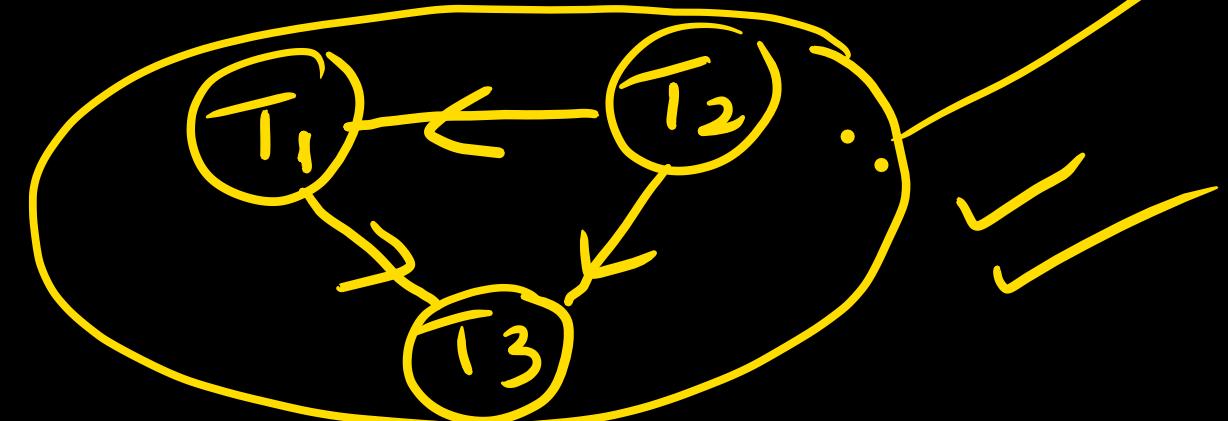
- If $s_1 s_2$ are conflict equal then $s_1 s_2$ precedence graph will be same
- If $s_1 s_2$ schedules precedence graph not equal then s_1 and s_2 are not conflict equal.

~~Exem~~ Examples

- If $s_1 s_2$ precedence graphs are equal then $s_1 s_2$ may not be conflict equal.



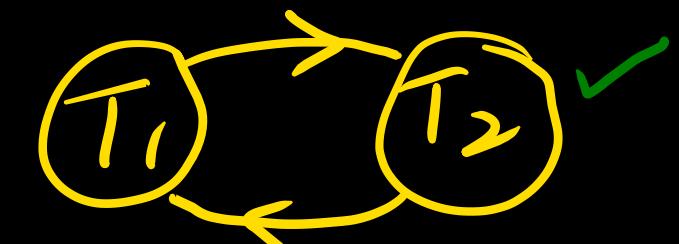
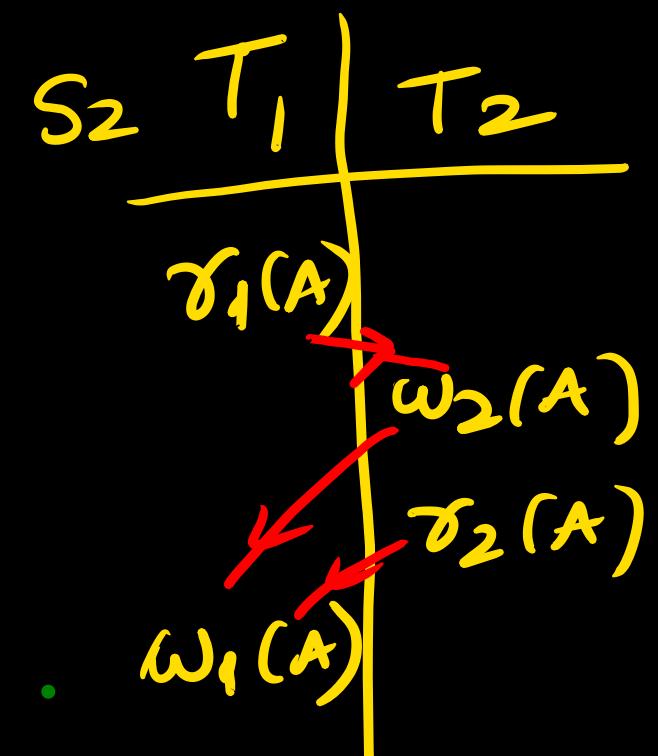
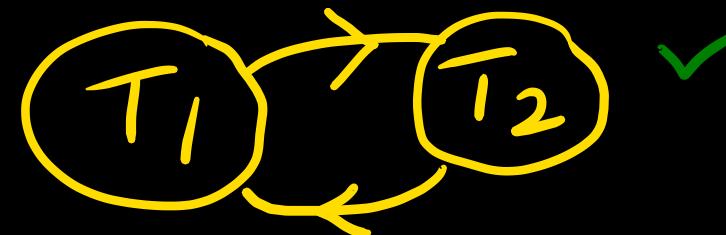
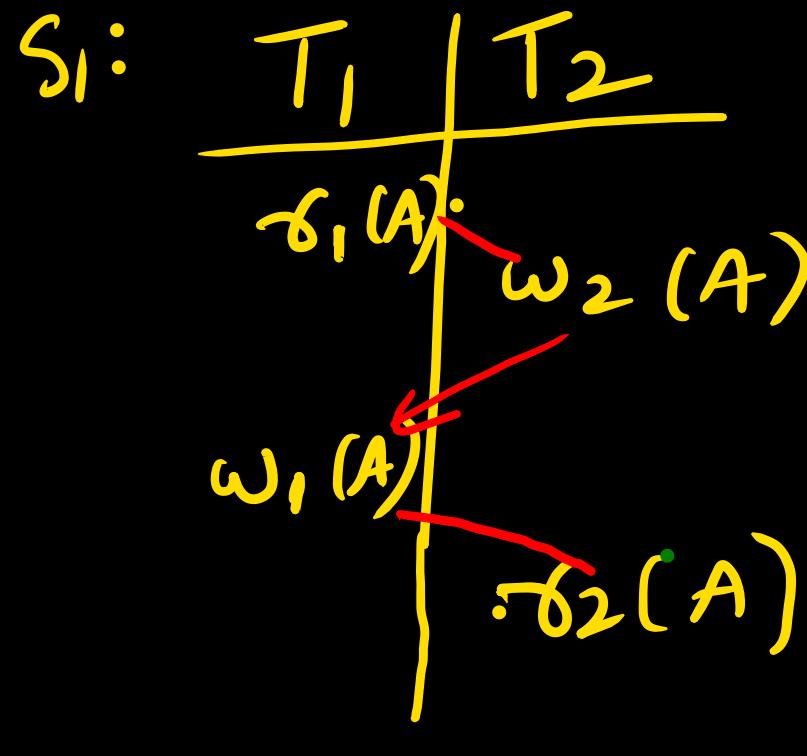
PG not equal



$S_1, S_2 \equiv$ conflict equal

Example of schedules which have same precedence graph

but schedules are not conflict equal:



S_1 : conflicts:

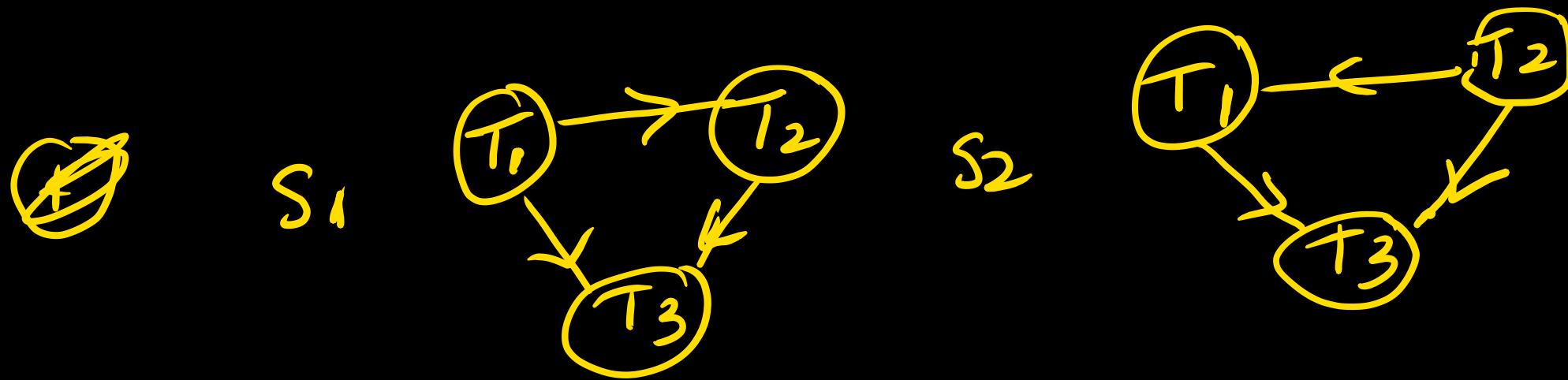
- $\gamma_1(A) \quad w_2(A); \quad w_2(A) \quad w_1(A)$
- $w_1(A) \quad r_2(A)$
- $\gamma_1(A) \quad w_2(A); \quad w_2(A) \quad w_1(A)$
- $\gamma_2 \wedge w_1(A)$

The above case happens only when there are cycles in the precedence graph.

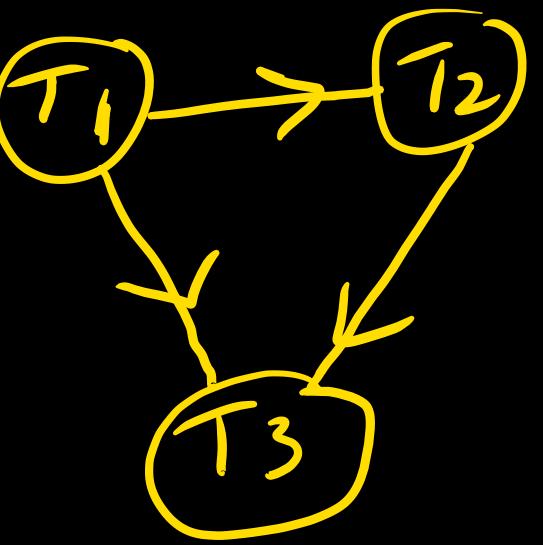
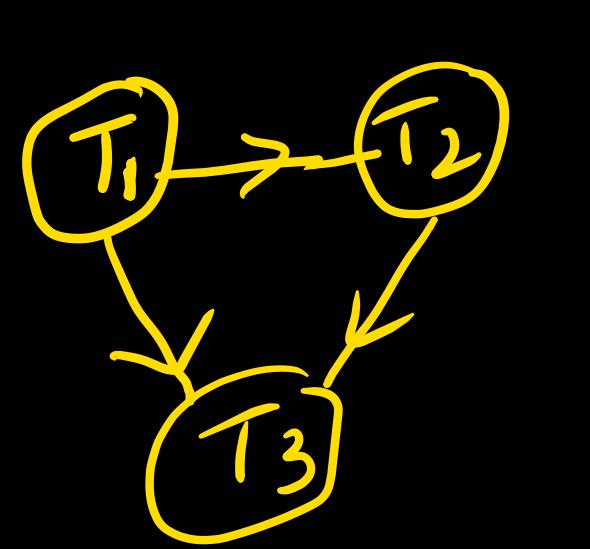
If $\underline{s_1 \ s_2}$ schedules have

- a) Same precedence graph
BB
 - b) ~~both~~ precedence graph are acyclic
- then s_1, s_2 schedules are conflict equal

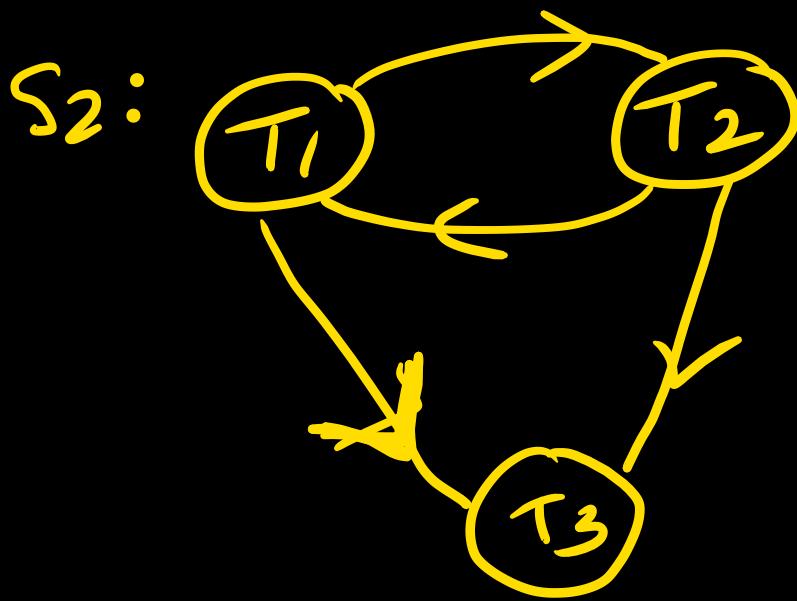
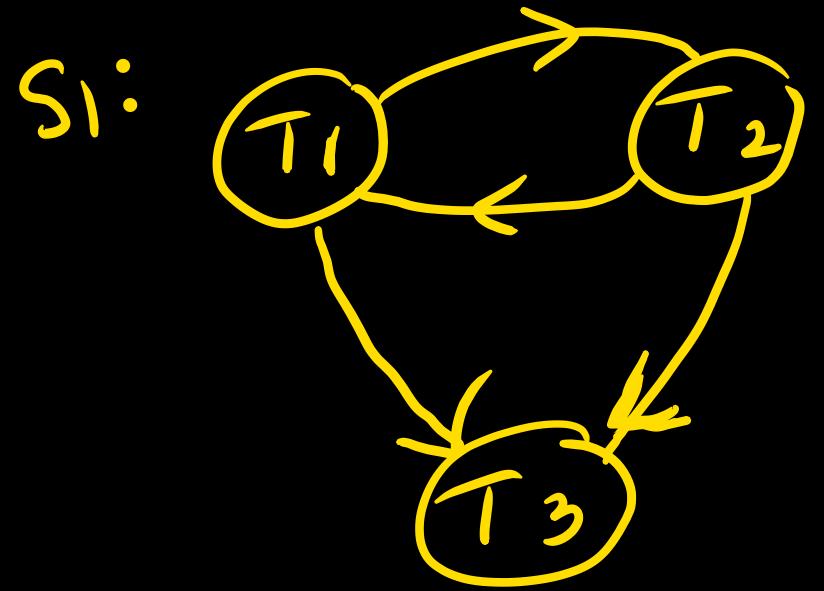
Given a precedence graph, can you say conflict equality:



precedence graph is different, therefore
Schedulers are not conflict equal.



when precedence graphs are same and are acyclic,
then they guy schedules are conflict equal.



In case if both precedence graphs are equal and are cyclic,
then S_1, S_2 may or may not be conflict equal.
In this case we have to manually check all
conflict pair precedence.