

CS 186 Discussion #9

Locking Granularity, MVCC

Logistics

- Homework 4 due this Wednesday
- Midterm 2 next Monday
 - See Piazza post
 - Review Session this Saturday, 1-3 PM
- No vitamin this week

Conflict Equivalence

Schedule 1

T1	R(A)		W(A)			R(B)			W(B)	
T2		R(A)						W(A)		R(C)
T3				R(A)	R(B)		R(C)			

Schedule 2

T1		R(A)		W(A)					R(B)	W(B)
T2			R(A)				W(A)	R(C)		
T3	R(B)				R(C)	R(A)				

Midterm Topics

- Relational Algebra
- Relational Modeling (ER Diagrams)
- Functional Dependencies
- Query Optimization
- Transactions
- Concurrency Control

Locking Granularity

- Databases have hierarchies
 - (Top) Databases > tables > pages > tuples
 - Assign locks top down, release bottom up
- Use intent locks!
 - IS - intent to get S
 - IX - intent to get X
 - SIX - S and IX

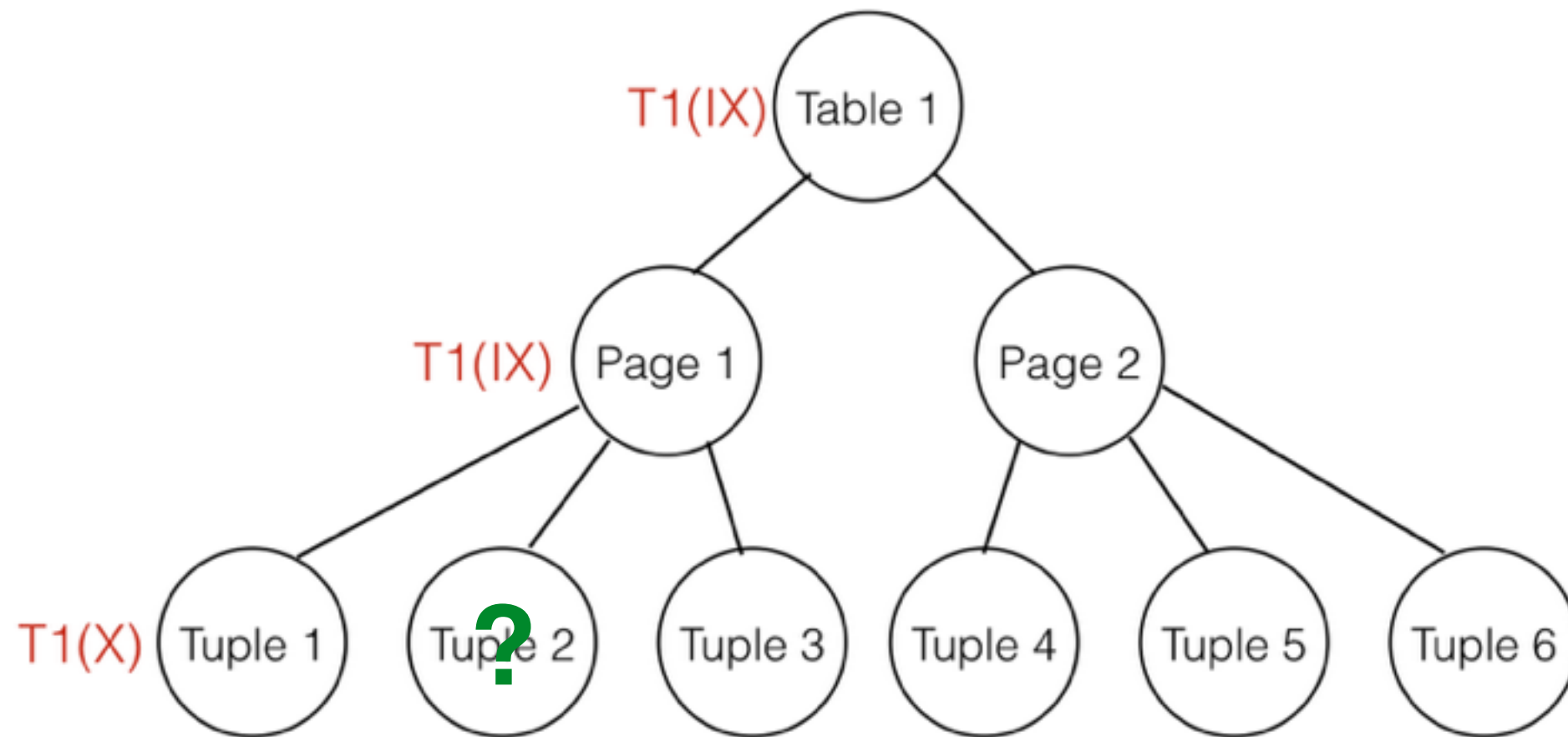
Locking Granularity

	IS	IX	SIX	S	X
IS	✓	✓	✓	✓	—
IX	✓	✓	—	—	—
SIX	✓	—	—	—	—
S	✓	—	—	✓	—
X	—	—	—	—	—

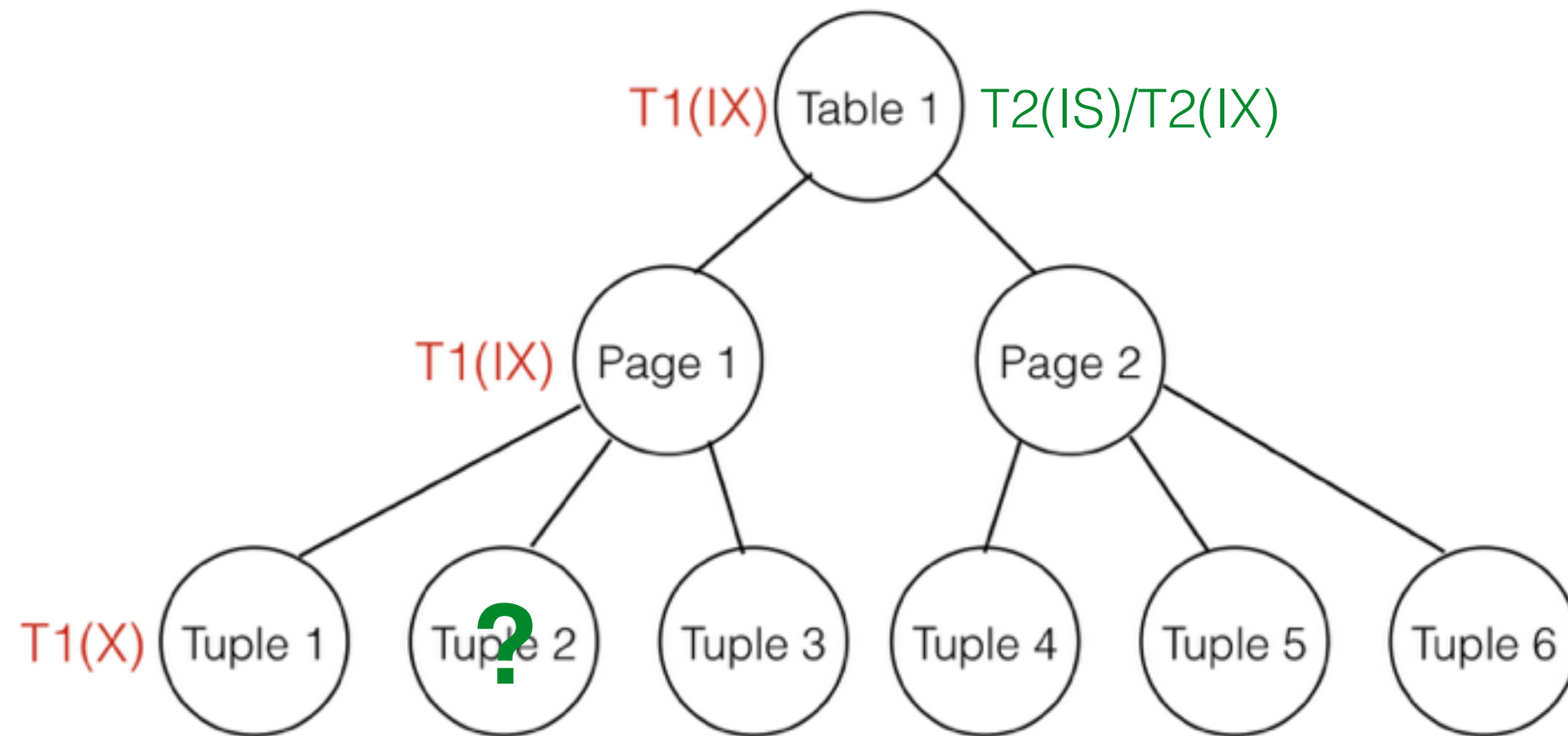
From Wikipedia:

To Get	Must Have on all Ancestors
IS or S	IS or IX
IX, SIX or X	IX or SIX

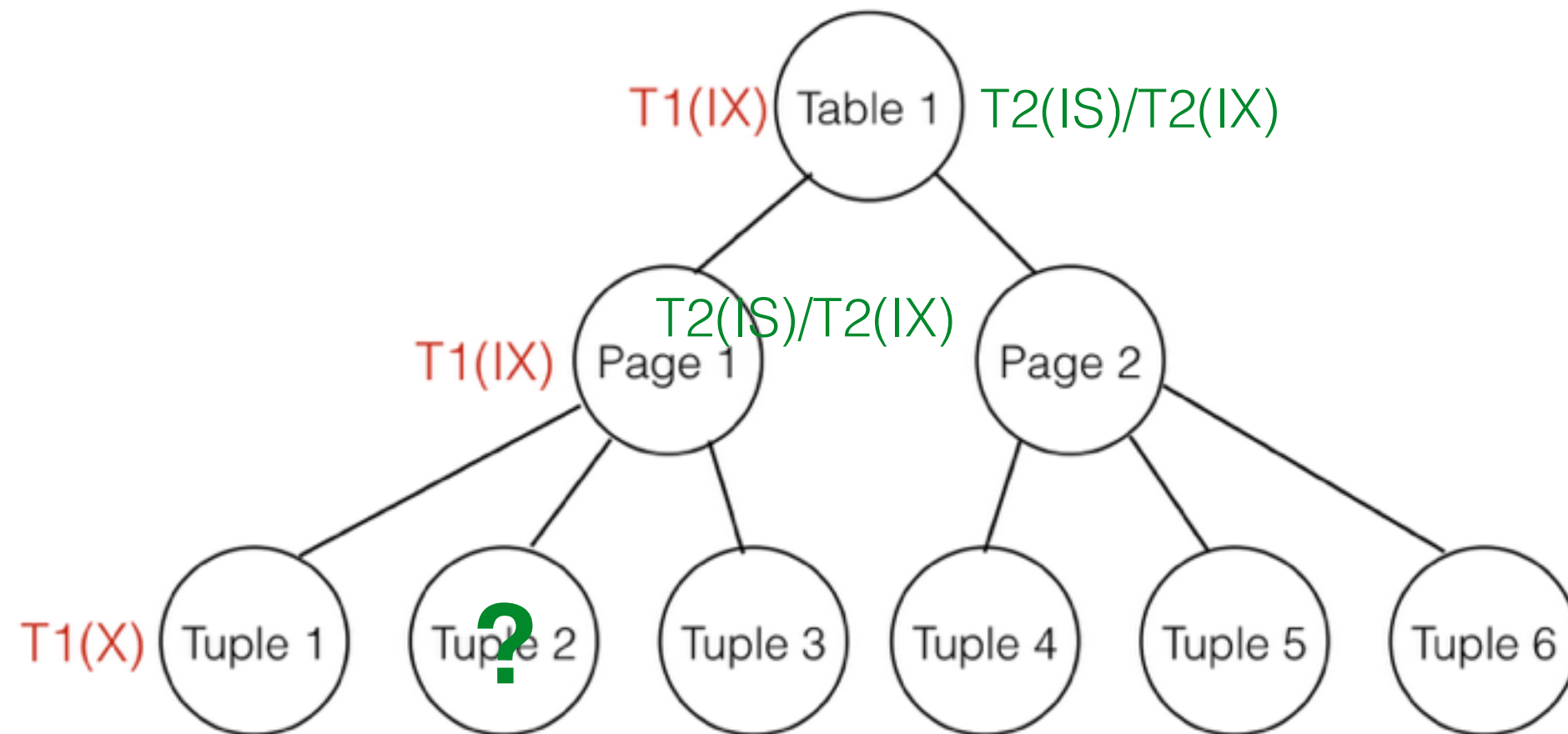
Worksheet - #3



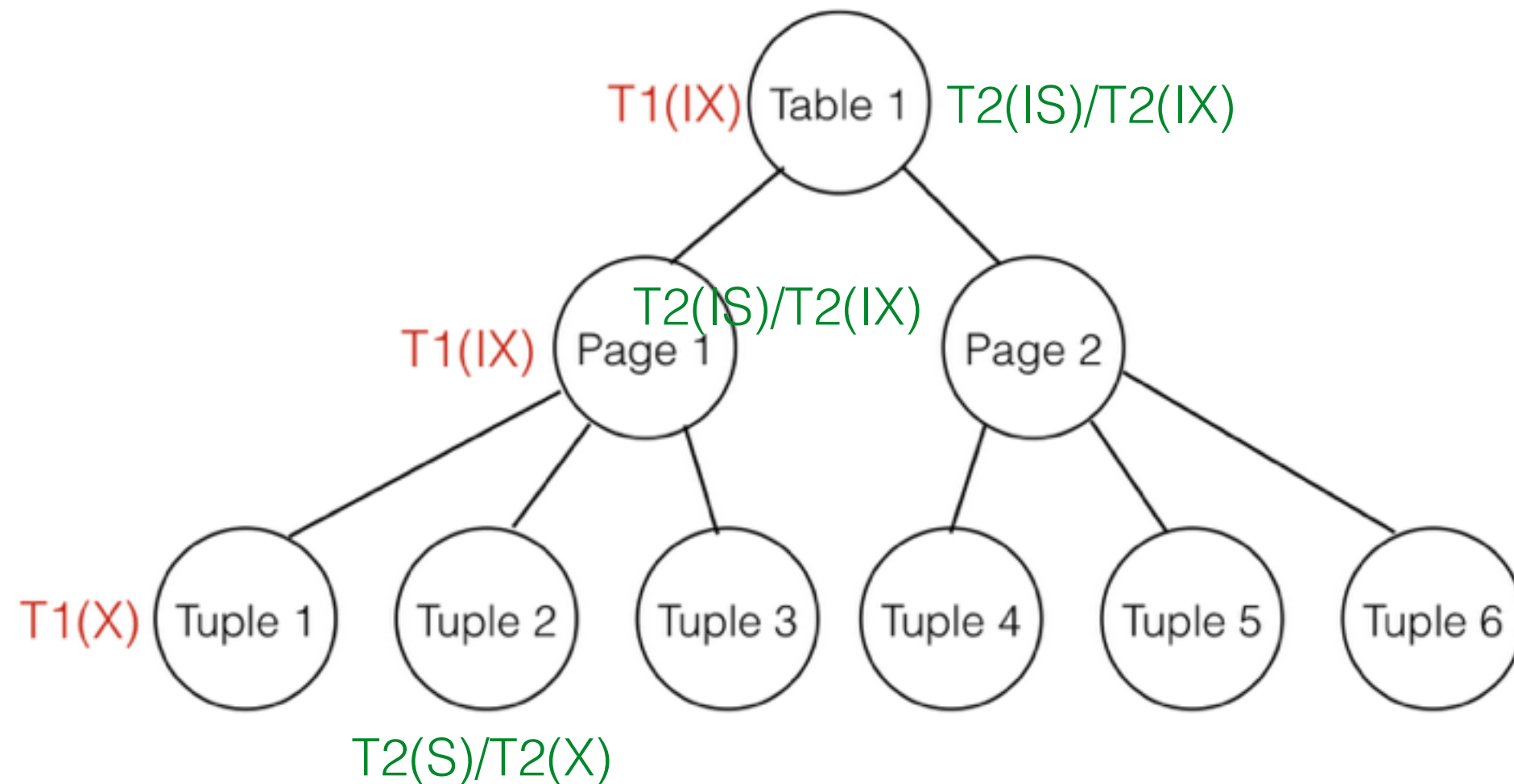
Worksheet - #3



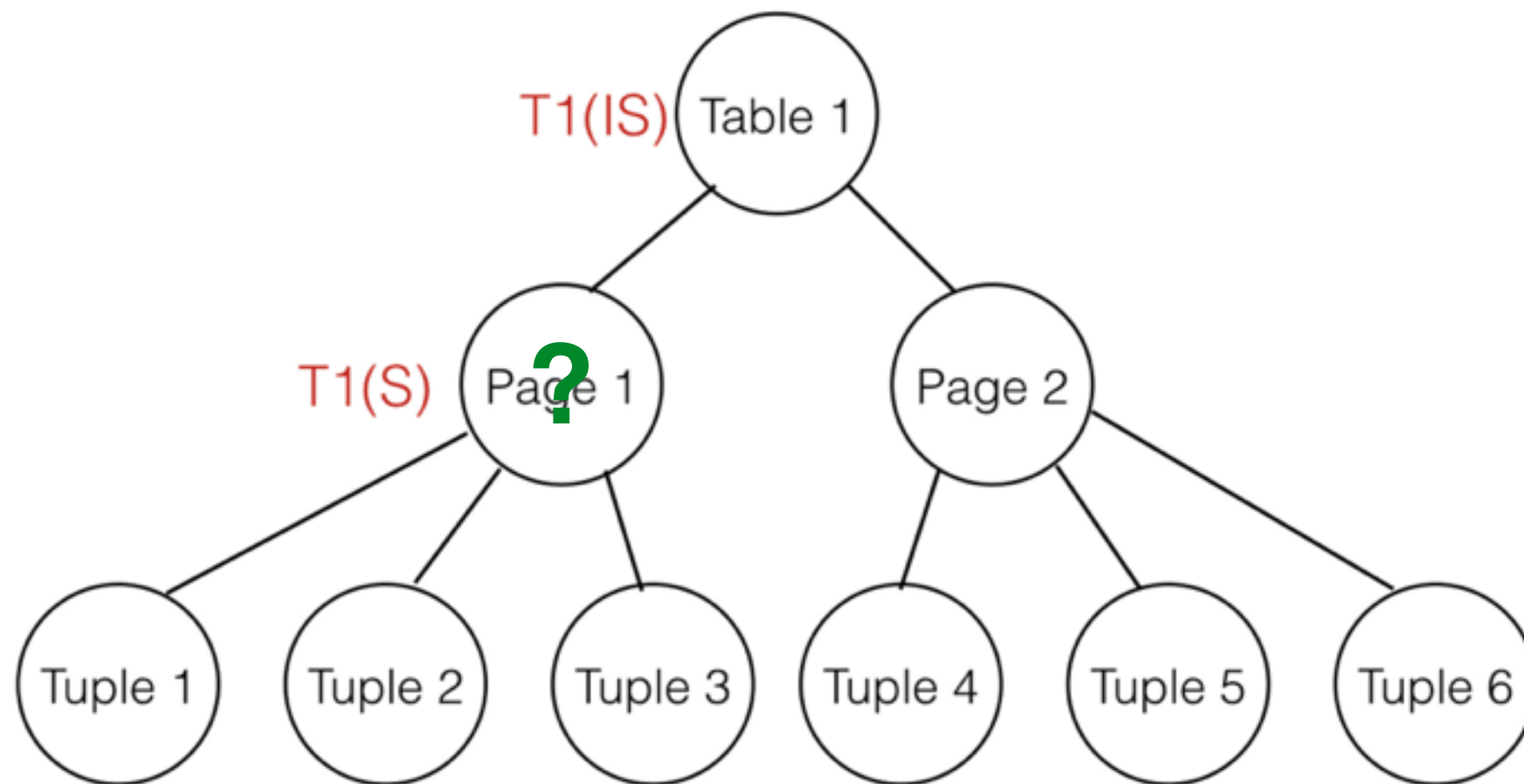
Worksheet - #3



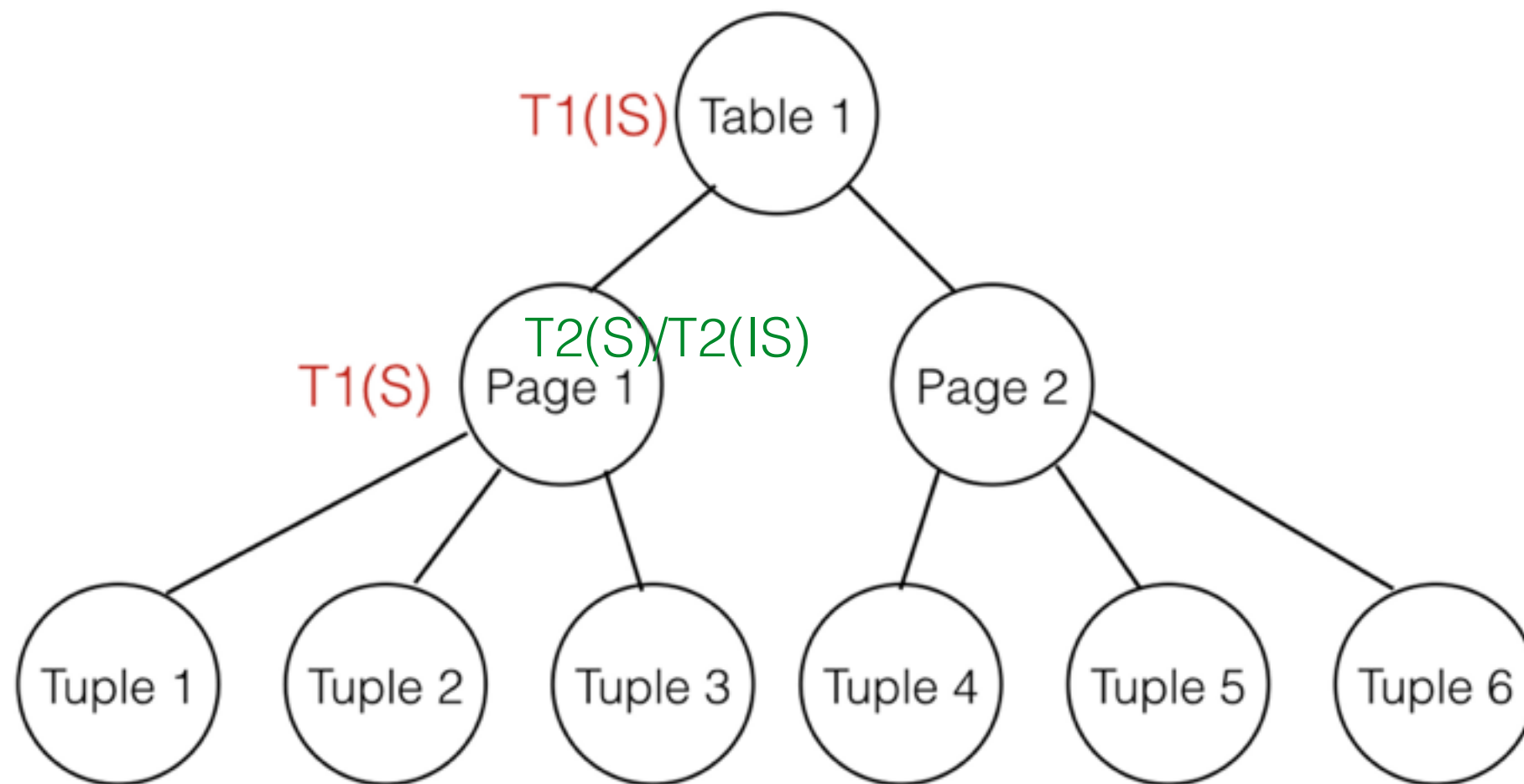
Worksheet - #3



Worksheet - #3



Worksheet - #3

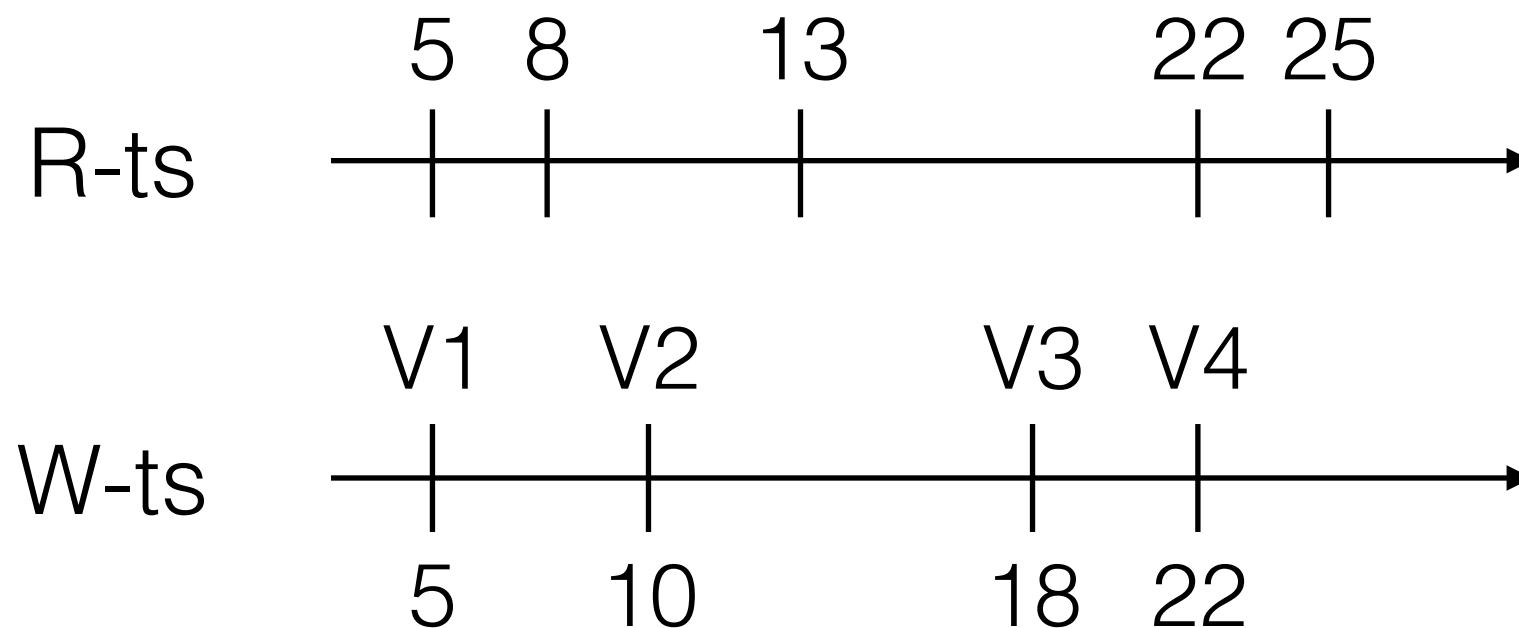


MVCC

- Each transaction:
 - is assigned a unique timestamp
- Each object:
 - has a set of read timestamps
 - has a set of write timestamps (versions)

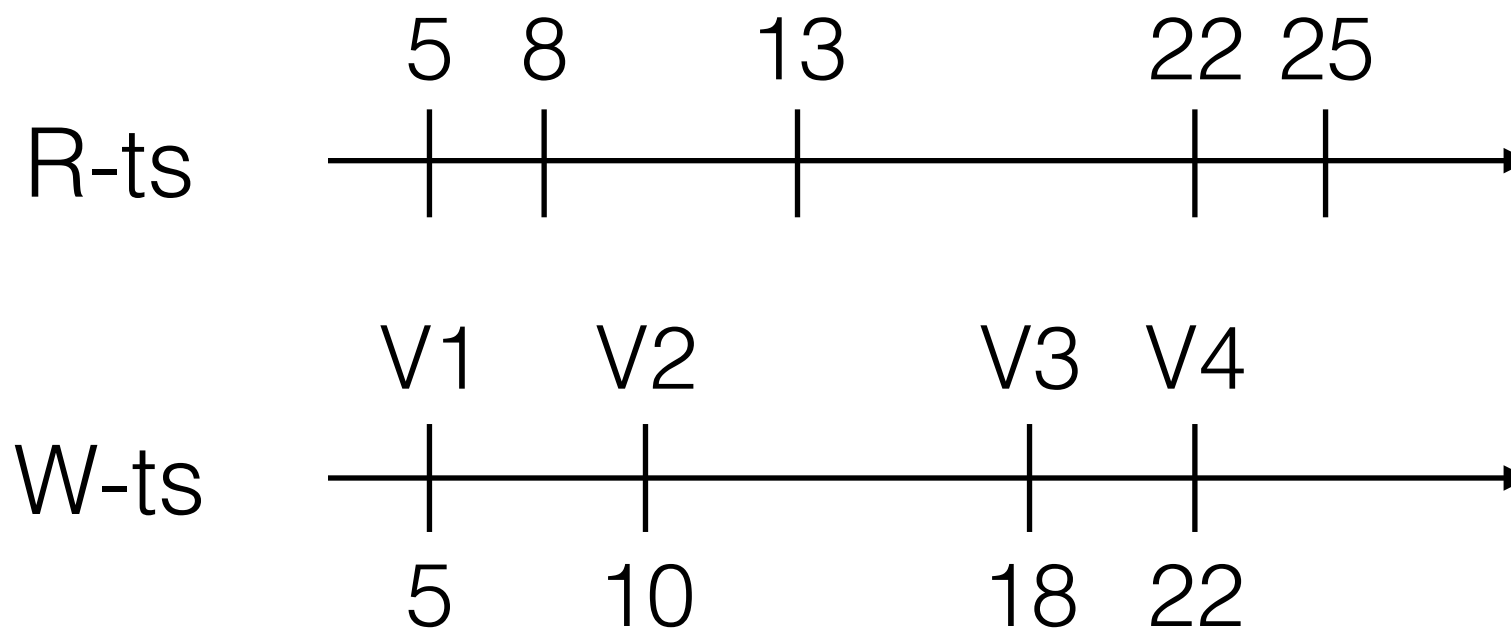
MVCC - Reads

- Reads will always succeed, and never have to wait
 - Can do this because multiple versions are stored



MVCC - Writes

- First check if xact needs to abort
 - Would write change another xact's read value?



MVCC - Writes

- First check if xact needs to abort
 - Would write change another xact's read value?
- If no abort, write a new version

