

**Sample CS 542 questions**  
**(Shows example of topics covered for making final exam)**

What is two phase locking protocol for concurrency control? Prove that two phase locking is a sufficient condition for serializability. Show that it is not necessary (you may use a counter example).

Given two transactions T1 and T2 (with read and write sets R1, W1, R2, W2) as follows:

- a. T1: R1 – W1
- b. T2: R2 – W2

Assume T1 arrives before T2. Include some conflicts between (R1 - W2 and/or R2 - W1 and/or W1 - W2). Given an example of history that is serializable in optimistic concurrency control but not in locking based concurrency control.

When will optimistic Concurrency Control approach perform better than locking? Why?

Present ideas to implement in an efficient way:

Validation procedure in optimistic. Dealing with a transaction that repeatedly gets aborted.

How would you make changes to distributed concurrency control algorithms if communication delay was variable across participating sites?

How can you deal with out of order messages in a centralized concurrency control algorithm? Specifically explain how all slave (other) sites post the database updates for transactions (initiated at master site) in the same order for all transactions with unnecessary delay. You may use an example.

What are the benefits and overheads of various granularity (size) of database that is locked?

What are different measures for concurrency control algorithms?

What are the different ideas for increasing the degree of concurrency? You may give examples.

What different ideas can be used to ensure serializability?

What is a dead transaction in a history of concurrent transaction?

What is a flexible transaction?

What is the difference between centralized control and primary copy based concurrency control?

What is the difference in a lock and time stamp on a database item? When can a time stamp be assigned to a transaction? How can we order time stamps on all sites in a distributed system?

What do we mean by degrees of commitment of a transaction?

Give the state transitions and message sent/received in a Centralized two phase commit protocol.

What are the problems in a commit protocol if a failure of a participating site occurs? Give an example to illustrate the problem.

What is a committable state in a simple termination protocol? Show that the simple termination protocol given in the following quotes:

"If at least one committable message is received, then commit the transactions; otherwise, abort it"  
is not robust and will cause blocking.

How can the simple termination protocol be made robust? Give the summary of messages that are needed to abort, and commit the transaction.

Define the concurrency set of a local state in a commit protocol.

Instead of storing back-up copies of updated database stable storage (hard disk), one can store back-up copies at other sites. What characteristics of I/O cost and communication cost will allow this scheme to be practical?

For an N site distributed system, how many messages are exchanged in :

Centralized 2-phase commit protocol?

Decentralized 2-phase commit protocol?

Give the finite state automation (FSA or state/transition diagrams) for the local protocols for the two phase commit in a distributed database system. (Assume the states to be initial, wait, abort, and commit.)

(a) Centralized two phase commit

(b) Decentralized two phase commit

How would you make changes to distributed concurrency control algorithms if communication delay was high across participating sites?

What are the advantages of centralized concurrency control over distributed concurrency control? What are the disadvantages?