CS5570: Advanced Database Management Systems

Test topics for test 1. Monday, Oct. 5, 2015. Class time.

1. Transaction properties and their interrelationship. You must be very clear about this relationship. You should know the interrelationship among these attributes. You should know how they are dependent on each other, that is, what could happen to other properties if one is not satisfied during the execution of concurrent transactions.
2. Questions on serializability and recoverability. Given a set of schedules or histories or both identify which is recoverable, which is serializable and which is both. You must know how many different ways a set of transactions can be executed. You should know the strengths and weaknesses of each way of executing transactions. You must know types of operation a transaction can apply on a data item.
3. Question on serializability theorem. Also you may be given a few schedules and you may be asked to serialize them. If you cannot serialize a schedule then you have to explain the reason.
4. Questions on categories of concurrency control mechanisms, two-phase locking, dirty read, unrepeatable read, lost update, deadlock, phantom problem and cascade roll-back.
5. Variations of two-phase locking schemes.

Test format: You will be asked to answer 3 or 4 questions out of 5 or 6 or 7 questions.

Question format: a question may have multiple parts, e.g., Q1a, Q1b, Q1c. etc.

Test time: Class time (1 hour 15 minutes)

Test type: Everything closed (book, internet, notes, etc.)

**Student Name**:

CS570 Advanced Database Management Systems V. Kumar

**Question 1 continues**:

Student Name:

CS570 Advanced Database Management Systems V. Kumar

**Q2a**. What is a *serializable* schedule and what is a *recoverable* schedule? Give examples.

**b.** Consider the following schedules:

* + 1. w1(x,2) r2(x) w2(y,3) c2
    2. w1(x,2) r2(x) w2(y,3) c1 c2
    3. w1(x,2) r2(x) w2(y,3) c2 c1
    4. w1(x,2) w2(x,3) a1
    5. w1(x,2) w2(x,3) a1 a2
    6. w1(x,2) w2(x,3) a2 a1
    7. w1(x,2) w2(x,3) c2 a1

Answer the following questions. Explain your answer.

* Which are *recoverable* and which are not,
* which are *serializable* and which are not,

**Student Name**:

CS570 Advanced Database Management Systems V. Kumar

**Question 2 continues**:

* which are both, i.e., *recoverable* and *serializable*

Student Name:

CS570 Advanced Database Management Systems V. Kumar

**Q3a**. Consier the following history:

***rl1(x)******r1(x) ru1(x) wl2(x) w2(x) wl2(y) w2(y) wu2(x) wu2(y) c2 wl1(y) w1(y) wu1(y) c1***

Is this history serializable? Give the serialization graph if it is or it is not serializable. If it is not serializable then do you think a serializable (not serial) history can be produced? Explain your answer. (**Note**: ***l*** represents locking and ***u*** represents unlocking)

**b.** How many different types of lock modes are there in multigranularity locking and what are they? Draw a compatibility matrix for these lock modes.

Student Name:

CS570 Advanced Database Management Systems V. Kumar

**Q4**. Explain the *phantom* problem. Why does it exist only in dynamic databases and not in static databases? Show with a simple example (execution of transactions) how does it exist. Suggest a solution for phantom problem.

Student Name:

CS570 Advanced Database Management Systems V. Kumar

**Q5a**. Show that timestamp (TO) rule enforces serializability but not necessarily recoverability. Give a suitable example (serializable but unrecoverable history) in support of your answer.

**b**. Give a serializable history (schedule) which is produced by Basic TO and show that this history cannot be produced by any 2PL scheduler. Explain your answer.

**Student Name**:

CS570 Advanced Database Management Systems V. Kumar

**Question 5 continues**:

Student Name:

CS570 Advanced Database Management Systems V. Kumar

**Q6**. Complete the following and give suitable example:

1. A transaction *Ti* reads x from transaction *Tj* in an execution, if,
2. An execution is **recoverable** if,
3. An execution **cascades** if,
4. An execution is **strict**, if,
5. A history is **serializable** if,
6. Two histories **H** and **H’** are equivalent if,

**Student Name**:

CS570 Advanced Database Management Systems V. Kumar

**Question 6 continues**: