National University of Computer and Emerging Sciences, Lahore Campus

Course:	Advanced Database Concepts	Course Code:	CS451
Program:	BS(Computer Science)	Semester:	Spring 2017
Duration:	60 Minutes	Total Marks:	30
Paper Date:	21-Feb-17	Weight	12.5%
Section:	ALL	Page(s):	1
Exam:	Midterm-I		

Instruction/Notes:

Q1. *(4 points)* Consider the university enrollment database schema:

Student(snum, sname, major, level, age) Class(name, meets_at, room, fid)
Enrolled(snum, cname) Faculty(fid, fname, deptid)

The meaning of these relations is straightforward; for example, Enrolled has one record per student-class pair such that the student is enrolled in the class. For each of the following transactions, state the SQL isolation level (lowest possible) you would use and explain why you chose it.

a) Enroll a student identified by her snum into the class named 'Database Systems'.
b) Change enrollment for a student identified by her snum from one class to another class.
c) Assign a new faculty member identified by his fid to the class with the least number of students.
d) For each class, show the number of students enrolled in the class.

Ans:
a. Because we are inserting a new row in the table Enrolled, we do not need any lock on the existing rows. So we would use READ UNCOMMITTED.
b. Because we are updating one existing row in the table Enrolled, we need an exclusive lock on the row which we are updating. So we would use READ COMMITTED.
c. To prevent other transactions from inserting or updating the table Enrolled while we are reading from it (known as the phantom problem), we would need to use SERIALIZABLE.
d. same as above.

Q2. *(10 points)* Consider the following classes of schedules: conflict-serializable, view-serializable, recoverable, cascadeless, and strict. For each of the following schedules, state which of the preceding classes it belongs to. If you cannot decide whether a schedule belongs in a certain class based on the listed actions, explain briefly. The actions are listed in the order they are scheduled and prefixed with the transaction name. If a commit or abort is not shown, the schedule is incomplete; assume that abort or commit must follow all the listed actions.

a) S1. T1:R(X), T2:R(Y), T3:W(X), T2:R(X), T1:R(Y)

b) S2. T2: R(X), T3:W(X), T3:Commit, T1:W(Y), T1:Commit, T2:R(Y), T2:W(Z), T2:Commit

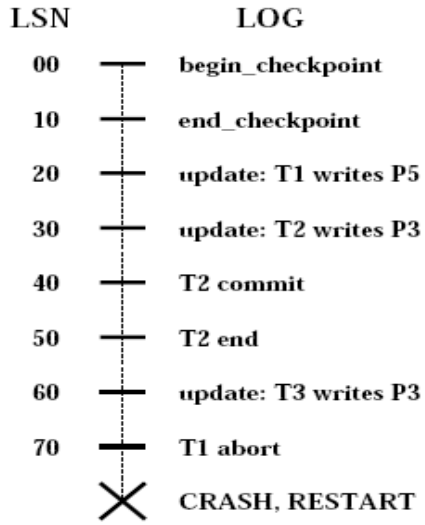
Ans: S1. It is conflict-serializable, view serializable, not cascadeless, not strict. We cannot decide whether it's recoverable or not, since the abort/commit sequence of these two transactions are not specified. Equivalent serial schedule is T1→T3→T2.
S2. It is conflict-serializable, view serializable, recoverable, cascadeless, strict. Equivalent serial schedule is T1→T2→T3.

Q3. *(10 points)* Consider the following concurrency control protocols: 2PL, Optimistic, Basic Timestamp, Strict Timestamp, and Timestamp with Thomas Write Rule.
For the schedule S2. T2: R(X), T3:W(X), T3:Commit, T1:W(Y), T1:Commit, T2:R(Y), T2:W(Z), T2:Commit,
state which of these protocols allows it, that is, allows the actions to occur in exactly the order shown. Justify your answer.

Ans:
2PL(N), Optimistic(N), Basic TS(Y), S-TS(Y), and TS W/TWR(Y)

Q4. *(1+3+1+1= 6 points)* Consider the execution shown in below Figure. Assume that the Dirty Page Table and Transaction Table were empty before the start of the log.

a) What is the value of the LSN stored in the master log record?
b) What is done during Analysis? (Be precise about the points at which Analysis begins and ends and describe the contents of any tables constructed in this phase.)
c) What is done during Redo? (Be precise about the points at which Redo begins and ends.)
d) What is done during Undo? (Be precise about the points at which Undo begins and ends.)



Ans:
a) LSN 00 is stored in the master log record as it is the LSN of the begin checkpoint record.
b) Analysis determines that the last begin checkpoint was at LSN 00 and starts at the corresponding end checkpoint
(LSN 10). We will denote Transaction Table records as (transID, lastLSN) and Dirty Page Table records as (pageID, recLSN) sets.
Then Analysis phase runs until LSN 70, and does the following:

LSN 20 Adds (T1, 20) to TT and (P5, 20) to DPT
LSN 30 Adds (T2, 30) to TT and (P3, 30) to DPT
LSN 40 Changes status of T2 to ”C” from ”U”
LSN 50 Deletes entry for T2 from Transaction Table
LSN 60 Adds (T3, 60) to TT. Does not change P3 entry in DPT
LSN 70 Changes (T1, 20) to (T1, 70)
The final Transaction Table has two entries: (T1, 70), and (T3, 60). The final Dirty Page Table has two entries: (P5, 20), and (P3, 30).

c) Redo Phase: Redo starts at LSN 20 (smallest recLSN in DPT).
LSN 20: Changes to P5 are redone.
LSN 30: P3 is retrieved and its pageLSN is checked. If the page had been written to disk before the crash (i.e. if pageLSN >= 30), nothing is re-done otherwise the changes are re-done.
LSN 40,50: No action
LSN 60: Changes to P3 are redone
LSN 70: No action

d) Undo Phase: Undo starts at LSN 70 (highest lastLSN in TT). The Loser Set consists of LSNs 70 and 60. LSN 70: Adds LSN 20 to the Loser Set. Loser Set = (60, 20). LSN 60: Undoes the change on P3 and adds a CLR indicating this
Undo. Loser Set = (20). LSN 20: Undoes the change on P5 and adds a CLR indicating this Undo.