Virginia Tech. **CS 4604 – Introduction to DBMS**
Computer Science **Spring 2014, Prakash**

Homework 7: Transactions, Logging and Recovery
(due April 24th, 2014, 3:30pm, in class—hard-copy please)

Reminders:

a. Out of 100 points. Contains 4 pages.
b. Rough time-estimates: 2~4 hours.
c. Please type your answers. Illegible handwriting may get no points, at the discretion of the grader. Only drawings may be hand-drawn, as long as they are neat and legible.
d. There could be more than one correct answer. We shall accept them all.
e. Whenever you are making an assumption, please state it clearly.
f. Lead TA for this HW: Pranav Nakate.

Q1. Serializability [30 points]

Consider the following schedules. The actions are listed in the order they are scheduled, and prefixed with the transaction name.

S1: T1:R(X), T2:R(Y), T1:W(X), T2:W(Y), T1:W(Y), T2:R(Z)

S2: T1:R(X), T2:W(X), T2:W(Y), T3:W(Y), T1:W(Y), T1:R(X), T3:R(Y)

Q1.1. (15 points) Consider the schedule S1.

A. (5 points) Draw the precedence graph for S1.
B. (5 points) Is S1 a conflict serializable schedule? If yes, what is the equivalent serial schedule? If no, explain in 1-2 lines.
C. (5 points) From the following options, choose which of the anomalies is present in S1 (if any). Explain in 1-2 lines which actions cause that anomaly.
    a. Dirty read (WR Conflict)
    b. Unrepeatable read (RW Conflict)
    c. Lost update (WW Conflict)

Q1.2. (15 points) Consider the schedule S2.

A. (5 points) Draw the precedence graph for S2.

B. (5 points) Is S2 a conflict serializable schedule? If yes, what is the equivalent serial schedule? If no, explain in 1-2 lines.
C. (5 points) From the following options, choose which of the anomalies is present in S2 (if any). Explain in 1-2 lines which actions cause that anomaly.
    a. Dirty read (WR Conflict)

b. Unrepeatable read (RW Conflict)
c. Lost update (WW Conflict)

Note: The anomalies due to interleaved execution of transactions are explained on slide 38-44 of the lecture 17 or page 526-529 of the textbook. The concept of conflict serializable schedules is explained on slide 46 of lecture 17 or page 550-551. The precedence graphs are explained on slide 51-52 of the lecture 17 or page 550-551 of the textbook.

Q2. Locking Protocols [20 points]

Consider the following schedules. The actions are listed in the order they are scheduled, and prefixed with the transaction name.

S1: T1:R(Z), T2:W(X), T2:R(Y), T1:W(X), T1:Commit, T2:Commit

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

S3:T2:W(A),T3:W(B),T1:W(C),T3:R(A),T2:R(B),T2:W(D),T1:R(D),T1:Commit, T2:Commit, T3:Commit.

Q2.1. (10 points) Consider schedule S1.

A. (5 points) Write the schedule S1 in a table format and draw the precedence graph.

B. (5 points) Will the actions of S1 be allowed by the following concurrency control protocols?
   a. 2PL
   b. Strict 2PL
   If YES, show in the table form where the lock requests can happen; If NO, explain briefly in 1-2 lines.

Q2.2. (10 points) Consider schedule S2.

A. (5 points) Write the schedule S2 in a table format and draw the precedence graph.
B. (5 points) Will the actions of S2 be allowed by the following concurrency control protocols?
   a. 2PL
   b. Strict 2PL
   If YES, show in the table form where the lock requests can happen; If NO, explain briefly in 1-2 lines.

Q2.3. (10 points) Consider schedule S3.

A. (5 points) Write the schedule S3 in a table format and draw the precedence graph.

B. (5 points) Will the actions of S3 be allowed by the following concurrency control protocols?

a. 2PL
b. Strict 2PL

If YES, show in the table form where the lock requests can happen; If NO, explain briefly in 1-2 lines.

Note: "2PL" and "Strict 2PL" protocols are explained on slide 18-33 of the lecture 18 or textbook page 551-552. "Lock Management" which explains how the lock requests are granted is explained on slide 36-41 of the lecture 18 or on page 553-554 of the textbook.

Q3. Deadlock Management [20 points]

Consider the following sequence of actions, listed in the order it is submitted to the DBMS (S is a shared lock, X is an exclusive lock):

S1: T1:S(A), T2:X(C), T3:S(A), T1:X(C), T3:X(B), T2: S(A), T3: X(A)
S2: T1:S(A), T2:X(B), T3:S(C), T2:S(C), T1:X(B), T3:S(A),T2:X(A), T3:X(A)

For S1 and S2 as given above, answer the following questions:

Q3.1. (5 points) For S1, write whether lock requests of its actions will be granted or blocked by the lock manager.

Q3.2. (5 points) Draw the waits-for graphs for S1 and write if the schedule will result in a deadlock condition. If there is no deadlock condition, write the order of completion of the schedule. Explain in 1-2 lines.

Q3.3. (5 points) For S2, write whether the lock requests of its actions will be granted or blocked by the lock manager.

Q3.4. (5 points) Draw the waits-for graphs for S2 and write if the schedule will result in a deadlock condition. Explain in 1-2 lines.

*Note***:** The implementation of lock and unlock requests is given in detail on slide 5-10 of lecture 18 or on page 554 of the textbook. The waits-for graphs are explained on slide 13 of lecture 18 or on page 556 of the textbook.

Q4. Logging [23 points]

Consider Exercise 18.4 in your textbook. Here we show a slightly easier to read Log table.

LSN	Log entry	prevLSN	undonextLSN
00	begin checkpoint		
10	end checkpoint		
20	update: T1 writes Page P2		
30	update: T1 writes Page P1		
40	update: T2 writes Page P5		
50	update: T3 writes Page P3		
60	T3 commit		
70	update: T2 writes Page P5		

80	update: T2 writes Page P3		
90	T2 abort		

Q4.1. (10 points) Write the appropriate values in the prevLSN and undonextLSN columns.

Q4.2. (5 points) Describe the actions taken to rollback transaction T2.

Q4.3. (8 points) Show the log after T2 is rolled back, including all the prevLSN and undonextLSN values in the log records.

Note: The Log record and CLR are explained in detail on page 582-585 of the textbook and also in Lecture 20.

Q5. Crashing now [5 points]

Consider the following log. It is the same one as given in Q4 above, but with a CRASH after LSN 90.

LSN	Log entry	prevLSN	undonextLSN
00	begin checkpoint		
10	end checkpoint		
20	update: T1 writes Page P2		
30	update: T1 writes Page P1		
40	update: T2 writes Page P5		
50	update: T3 writes Page P3		
60	T3 commit		
70	update: T2 writes Page P5		
80	update: T2 writes Page P3		
90	T2 abort		
	CRASH		

The recovery manager now sees this log (of course with the correct prevLSN and undonextLSN values filled in) after the CRASH. Which transactions will be redone and undone?

Note: The concept of crash recovery is explained in detail on page 588-592 of the textbook and also in Lecture 19.

Q6. Character fights [2 points]

Who would win the fight, if James Bond fought with Jason Bourne? Explain your answer. Is there anyone who can defeat both of them? ☺