CMPT-454 Fall 2009
Instructor: Martin Ester
TA: Yi Liu

Assignment 5

Total marks: 200 (20 % of the assignments)
Due date: December 2, 2009

Problem 5.1 (50 marks)

For each of the following schedules, state whether they are serializable and whether they are conflict-serializable. For each schedule, draw the corresponding precedence graph. If the schedule is conflict-serializable, show all the conflict-equivalent serial schedules. If the schedule is not serializable, provide an initial DB state and some semantics (pseudo-code) for the transactions for which no serial schedule has the same net effect.

a) S = r1(A) r2(A) w1(A) w2(A)

b) S = r1(A) r2(B) w3(A) r2(A) r1(B)

c) S = r1(A) w2(A) w1(A) r3(A)

Problem 5.2 (50 marks)

Show all the conflict-serializable schedules of the following transactions T1 and T2. Explain why these are all such schedules.

a) T1: r1(A) w1(A) r1(B) w1(B)
T2: r2(B) w2(B) r2(A) w2(A).

b) T1: r1(A) w1(A) r1(B) w1(B)
T2: r2(A) w2(A) r2(B) w2(B)

Problem 5.3 (40 marks)

Consider the following schedules of transactions T1, T2 and T3. Insert shared and exclusive lock actions as well as unlock actions and modify the schedule where necessary. Place a shared or exclusive lock immediately before the read or write action that requires it. Place the necessary unlocks at the end of a transaction. If a deadlock occurs, abort the participating transaction that started last, release all its locks and re-start that transaction after all the other participating transactions have committed. Show the complete schedules satisfying the 2PL protocol.

a) r1(A) r2(B) r3(C) r1(B) r2(C) r3(D) w1(A) w2(B) w3(C) w1(A)

b) r1(A) r2(B) r3(C) r1(B) r2(C) r3(A) w1(A) w2(B) w3(C) w1(A)

Problem 5.4 (60 marks)

Provide one example schedule (including the commits) of two transactions (that both commit, i.e. no aborts involved) with each of the following properties. In concurrency control by validation, suppose that validation is performed at the time of the commit. To show that a schedule validates or does not validate, show the read and write sets for the two transactions and show that both validation rules are satisfied or list the validation rule violated. To show that a schedule satisfies or does not satisfy the 2PL protocol, show the modified schedule after inserting shared and exclusive lock actions right before the corresponding read / write action and unlock actions right before the commit.

a) The schedule is allowed by 2PL, but not by concurrency control by validation.

b) The schedule is allowed by concurrency control by validation, but not by 2PL.