# COM S 461: ASSIGNMENT 5

Percentage in your final grade: 10%

Maximum score for this assignment: 100 points

Objectives:

1. To enhance your understanding of transaction management

Questions

1. (15 points): Write pseudo java code to implement locking mechanisms for strict 2PL.
    a. Provide data structure
    b. Provide pseudo code for *read(int tid, object o)*, *write (int tid object o)*, *commit(int tid)*, and *abort(int tid)*, where *tid* is the id of the transaction that calls the method.
2. (15 points): Consider the following actions taken by transaction T1 on database objects X and Y: R(X), W(X), R(Y), W(Y).
    a. Give an example of another transaction T2 that, if run concurrently to transaction T1 without some form of concurrency control, could interfere with T1.
    b. Explain how the use strict 2PL would prevent interference between the two transactions.
3. (20 points): Consider the following classes of schedules: *serializable*, *conflict-serializable*, *recoverable*, *avoids-cascading-aborts*, and *strict*. For each of the following schedules, state which of the above classes the schedule belongs to. If you cannot decide whether a schedule belongs in a certain class based on the listed actions, briefly explain.
    a. S1: $R_{T1}(X)$, $W_{T2}(X)$, $W_{T1}(X)$, $Abort_{T1}$, $Commit_{T1}$
    b. S2: $R_{T1}(X)$, $R_{T2}(X)$, $W_{T1}(X)$, $W_{T2}(X)$, $Commit_{T2}$, $Commit_{T1}$
4. (20 points): Consider the three transactions T1, T2, and T3, and the schedule S1 and S2 given below. Draw the serializability graphs for S1 and S2, and state whether each schedule is serializable or not. If a schedule is serializable, write down the equivalent serial schedule(s).

T1: $R_1(X)$, $R_1(Z)$, $W_1(X)$

T2: $R_2(Z)$, $R_2(Y)$, $W_2(Z)$, $W_2(Y)$

T3: $R_3(X)$, $R_3(Y)$, $W_3(Y)$

a. S1: $R_1(X)$, $R_2(Z)$, $R_1(Z)$, $R_3(X)$, $R_3(Y)$, $W_1(X)$, $W_3(Y)$, $R_2(Y)$, $W_2(Z)$, $W_2(Y)$;

b. S2: $R_1(X)$, $R_2(Z)$, $R_3(X)$, $R_1(Z)$, $R_2(Y)$, $R_3(Y)$, $W_1(X)$, $W_2(Z)$, $W_3(Y)$, $W_2(Y)$;

5. (10 points): Prove that strict two-phase locking guarantees strict schedules.

6. (20 points): Consider a database organized in terms of the following hierarchy of objects. The database itself is an object (D), and it contains two files (F1 and F2), each of which contains 1000 pages (P1…P1000 and P1001…P2000, respectively). Each page contains 100 records, and records are identified as p:i, where p is the page identifier and i is the slot of record on that page.

   Multiple-granularity locking is used, with S, X, IS, IX and SIX locks, and database-level, file-level, page-level, and record-level locking. For each of the following operations, indicate the sequence of lock requests that must be generated by a transaction that wants to carry out (just) these operations:

   a. Read record P1200:5;

   b. Read records P1200:98 through P1205:2;

   c. Read all (records on all) pages in file F1;

   d. Read pages P500 through P520;

   e. Read pages P10 through P980;

   f. Read all pages in F1 and (based on the values read) modify 10 pages;

   g. Delete record P1200:98 (this is a blind write);

   h. Delete the first record from each page (again, these are blind writes);

   i. Delete all records.