CIS720, Homework 2, Fall 2013, Due Dec 12th, 2013

1. Do the following executions satisfy sequential consistency

   a. P_1: $w_1(y)2$, $r_1(x)1$
   P_2: $w_2(x)2$, $r_2(z)1$
   P_3: $w_3(z)2$, $r_3(y)1$

   b. P_1: $r_1(x)1$, $w_1(y)2$, $r_1(x)2$
   P_2: $r_2(y)1$, $w_2(z)3$, $w_2(x)2$
   P_3: $r_3(y)2$, $r_3(z)3$, $r_3(x)1$

   All variables are initialized to 1. Give reasons for your answer.

2. Assume that you have three processes, A, B and C and two variables x and y. Process A writes x, B reads x and writes y and C only reads x and y. Each process may perform the operations above multiple times. Each process which reads a variable maintains a local copy. When A writes x, it sends the updated value to both B and C, who update their local copies. When B writes y, it sends the updated value to C which updates its local copy. Whenever B or C wants to perform a read, they simply read their local copies. Does this implementation ensure sequential consistency? If not, present a scenario that violates sequential consistency.

3. In the programming assignment, we have looked at operations on a queue. Consider a queue which is replicated at N sites. The operations to be supported are Enqueue, Dequeue, and Peek (which reads the element at the front of the queue without removing it). Assume that each replica has weight 1 and N = 23. Assume that the queue length is always greater than two.
   a. Give a possible quorum assignment for Enqueue, Dequeue and Peek operations to implement replication efficiently which ensure the following (a) Concurrent enqueue operations are disallowed, (b) Concurrent dequeue operations are disallowed, (c) At most 3 concurrent Peek operations are allowed, (d) Dequeue and Peek are not allowed at the same time.
   b. If the operation distribution is 40% Enqueue, 40% Dequeue and 20% Peek, how would adjust the quorum assignment for part (a) to improve efficiency. Give reasons for your answer.

4. In the class, we studied the two-phase commit protocol and found that the cohorts might have to block in certain cases. Assuming that the coordinator cannot fail, is the two-phase commit protocol non-blocking. Explain your answer.

5. The following transactions issue operations in the given order. All data items have initial value of 0.

T_1:	$r_1(x)$		$w_1(z)5$	$w_1(y)4$
T_2:		$r_2(x)$		$w_2(y)3$
T_3:	$r_3(y)$	$w_3(z)2$		$r_3(y)$

   (a) Assume that time-stamped concurrency control is being followed and transaction T_i is given timestamp i. Indicate the transactions which will complete successfully and those that will abort. Also, give the value returned by each read operation.

   (b) If two-phase locking is being used, what will be the result of the execution? What is the order in which operations will execute and what will be the values returned by the read operations?

6. Assume that there are three types of threads T1, T2 and T3 performing transactions on two arrays A[0..N] and B[0..M]. Type T1 threads perform transactions which only read values in array A. Type T2 threads perform transaction which first reads a set of array locations in A and then write a single array location in B. Finally, Type T3 threads perform transactions which may write multiple locations in B (no read operations).
    a. If timestamp concurrency control is used, state whether each of the following is true or false
        i. Transactions of type T1 never abort
        ii. Transactions of type T2 may have to abort
        iii. Transaction of type T3 never abort
    b. If two-phase locking is used, state whether each of the following is true or false:
        i. Transactions of type T1 may have to wait for a lock acquired by another transaction
        ii. Transactions of type T2 may have to wait for a lock acquired by transaction of type T1.
    c. For the execution of transactions described above, simplify two-phase locking as much as possible to improve throughput.

    For each case above, support each answer with reasons.