**CS542: Professor Bhargava**   
**Homework 3**   
**Due Date: April 4, 2013**

Q1. Give two examples of the application of commit and termination protocols in a distributed system.   
       (Hint: Mail facility is one such example. If you use this example, explain how would you make changes in the current mail facility that exists on our system across ITAP and CS machines.) (10)

Q2. The Read-one write-all-available protocol for site failure/recovery requires aborting of the   
       transactions who find differing session numbers of other sites. (15)

    a) How could you modify the protocol to avoid aborts? (It may not be easy). (2)

    b) What other ideas do you have to recover the database on the recovering sites that are different than fail-locks? (2)

    c) What are the windows of vulnerability during which if some other failure occurs, the   
        recovery may not be complete in your protocol? (2)

    d) How can the ideas from site failure protocol be extended to the problem of network partitions?   
        Under what conditions the transaction processing can continue and how the database can be   
        recovered in all partitions? (3)

    e) What type of performance measures can you think of for such protocols and how would you   
        measure them in a distributed database system such as RAID? Give some implementation ideas for continuing transactions and recovering the database. (3)

    f) Identify the problems when multiple sites may fail or recover during a transaction's life time. A clear statement will be needed. Suggest a solution to one of them. (3)

Q3. What are session numbers and nominal session numbers? What role they play in recovery? What are control transactions? What is their role? (10)

Q5. What is meant by the notion of 'fail lock'? Explain the how transaction processing takes place at the operation sites and recovery process at the failed site. (10)

Q6. Explain the various ideas of determining a 'unique majority partition' that can continue to process transactions during a network partition. Consider cases where multiple partitions and merging may take place. Suggest what useful processing can be done in minority partitions so that the merge process is quick and efficient. How can the size of the partition be determined at the time of partition and how can we determine if the mutual consistency has been violated at the time of merge? (20)