Homework 3

1. Suppose your group member has developed a library that provides a function *consensus(v)* to propose a value v and participate in a consensus protocol that terminates at the end of $f + 1$ rounds (assuming all other machines also invoke the function in the first round). Now you need to reliably broadcast a value v to all machines and want to reuse your friend's code as much as possible. Using as few extra messages as possible, implement the (possibly multi-round) functions *trb_sender(v)* and *trb_other()* that respectively specify the protocol for the sender and other machines using the consensus function as a black box. You must show Validity, Agreement, Integrity, and Termination.

   (a) Develop a protocol that terminates in $f + 2$ rounds.

   (b) Develop a protocol that terminates in $f + 1$ rounds. Note: this is not trivial, so take care.

2. Electing a president in Paxos helps laws to be passed quicker. Explain why.

3. Why does *PBFT* use $2f + 1$ matching certificates for checkpointing and not $f + 1$ matching ones? How come *Separating* only uses $f + 1$ matching certificates then?

4. State precisely the assumptions on failures under which a system using *Proactive recovery...* works correctly.

5. A Java Virtual Machine is nice because you have to develop your Java-based system just once and it works on most operating systems. *BASE* is nice because it reduces the cost of N-version programming for Byzantine fault-tolerance. Suppose you had a vision for a "Semantic Web Sliced Bread Protocol" very different from HTTP today and you also wanted it Byzantine fault-tolerant. How does *BASE* help?

6. Suppose replicas in Bayou could be Byzantine. How will you maintain the prefix property?

7. What kind of consistency does Akamai ensure for the files it serves?