Final Problem Set Due 5/15

Instructions: Do all 5 problems using your own ideaa. You may use slides, textbooks, and internet resources, but you will not receive credit for an answer if you copy and paste the solution. If a question is confusing, email me and I will clarify.

Question 1

Describe a scenario with 2 servers where Basic Paxos algorithm livelocks.

Question 2

In Basic Paxos, describe a scenario with 6 servers where 3 different servers accept 3 different values at some point during execution. Next, is it possible for a 4^{th} value that is different from the 3 that have been accepted so far is ultimately chosen? If so, describe the scenario. If not, why not?

Question 3

In Basic Paxos, when a proposer receives responses to a prepare request from a majority of nodes these respones may contain an acceptedValue and acceptedProposal number. If any acceptedValues are returned, the proposer's value is replaced with the acceptedValue for highest acceptedProposal. Describe a scenario where at least two different acceptedProposals are receive. In other words, describe a scenario where a proposer must actually find the highest acceptedProposal.

Question 4

As described in Question 3, in Basic Paxos, when a proposer receives responses to a prepare request from a majority of nodes these responses may contain an acceptedValue and acceptedProposal number. If any acceptedValues are returned, the proposer's value is replaced with the acceptedValue for *highest* acceptedProposal. Would the algorithm still work if instead of choosing the *highest* acceptedProposal, each proposer chooses the *lowest* acceptedProposal. Either give you reasoning why the algorithm still works or give an example execution where the algorithm fails.

Question 5

In what ways is the Basic Paxos algorithm similar to 2 phase commit? How are the two algorithms different?