Brandon Dotson
CS 550
Homework 4
11/27/17

2) The design issues of distributed scheduling are clock synchronization of nodes, ensuring mutual exclusion when it comes to shared resources, and having global positioning in place for nodes. Clock synchronization is important because many different machines are working together, and the order of processes and other events should be known. Mutual exclusion is necessary because there is a finite supply of shared resources that can only be handled by one machine at a time. Getting this wrong could affect performance. Finally, global positioning is important because the layout of our distributed system helps us decide on things like routing, effective methods of communication, and data placement.

3) Message logging is useful for checkpointing because it reduces the number of checkpoints required. Rather than make many checkpoints, a smaller number can be created and used in conjunction with message logs. The logs can then be used to trace from a checkpoint to another globally consistent state of the system.

4) A Byzantine Failure is when a server produces arbitrary responses at arbitrary times. In order to survive a k component fail in Byzantine Failure, 2k + 1 replicas are required.

5) In the Bully algorithm, having a recovering process start an election to become a new coordinate is a necessary feature of the algorithm. If this were not a feature, a process with lower priority might remain as coordinator after the higher process recovers. If the higher process that previously failed comes back online, there could be an issue if the current coordinator fails. If the higher process notices the failure, it will be unable to call for an election since processes may only notify those higher than themselves.

6) If two processes detect the failure of the coordinator at the same time, they will both call for an election. One of the processes will be higher in priority than the other, and it will tell the lower process to stop its election. Then, it will proceed with its own election messages until a new coordinator is found.

7) The two camps of file systems represented by Parallel File Systems and Hadoop File Systems come down to the organization of data and data access. In the Parallel File System, the data is spread among many different servers. In the Hadoop File System, the data resides in one node. The Parallel File System allows parallel, concurrent data access. On the other hand, Hadoop only allows access from one location at a time.

8) To implement an electronic stock market, I would use the monotonic-write consistency. Stock brokers will generally check the status of their stocks very often, and the individual changes to the stocks are just as important as the overall trends of the stock. With the monotonic-write consistency, every single update will be visible to the users.

9) In the case of a personal mailbox, I would use the monotonic-read consistency. When checking a personal mailbox, you always want to be able to read your most recent emails. The monotonic-read consistency allows the user to read all the most recent emails.

10) An example of client-centric consistency leading to write-write conflicts is the following: When using monotonic-writes, two different clients could make an update to the same data file at different. When this happens, they would both go through, but only the later of the two writes would be shown.

11) Since there are 10 servers: Nw must be > 5 and Nr + Nw > 10

Nr = 1, Nw = 10	Nr = 1, Nw = 9	Nr = 1, Nw = 8	Nr = 1, Nw = 7	Nr = 1, Nw = 6
Nr = 2, Nw = 10	Nr = 2, Nw = 9	Nr = 2, Nw = 8	Nr = 2, Nw = 7	Nr = 2, Nw = 6
Nr = 3, Nw = 10	Nr = 3, Nw = 9	Nr = 3, Nw = 8	Nr = 3, Nw = 7	Nr = 3, Nw = 6
Nr = 4, Nw = 10	Nr = 4, Nw = 9	Nr = 4, Nw = 8	Nr = 4, Nw = 7	Nr = 4, Nw = 6
Nr = 5, Nw = 10	Nr = 5, Nw = 9	Nr = 5, Nw = 8	Nr = 5, Nw = 7	Nr = 5, Nw = 6
Nr = 6, Nw = 10	Nr = 6, Nw = 9	Nr = 6, Nw = 8	Nr = 6, Nw = 7	Nr = 6, Nw = 6
Nr = 7, Nw = 10	Nr = 7, Nw = 9	Nr = 7, Nw = 8	Nr = 7, Nw = 7	Nr = 7, Nw = 6
Nr = 8, Nw = 10	Nr = 8, Nw = 9	Nr = 8, Nw = 8	Nr = 8, Nw = 7	Nr = 8, Nw = 6
Nr = 9, Nw = 10	Nr = 9, Nw = 9	Nr = 9, Nw = 8	Nr = 9, Nw = 7	Nr = 9, Nw = 6
Nr = 10, Nw = 10	Nr = 10, Nw = 9	Nr = 10, Nw = 8	Nr = 10, Nw = 7	Nr = 10, Nw = 6

12) Linearizability and sequential consistency are very similar in that they both require all shared accesses to be seen in the same order for processes. For linearizability, there is a timestamp for shared accesses while there isn't for sequential consistency. Sequential consistency is more practical to implement because you do not have to worry about timing. Timing would surely complicate things because it is difficult to synchronize different machines with varying hardware and clocks.

13) A web browser with an outdated cache page could be considered a response failure. In this case, we are receiving the outdated cache page in place of the recent version.

14) In reliable multicasting, it is not always necessary that the communication layer keeps a copy of a message for retransmission purposes. This is usually only required when we know the multicasting is unreliable. If this is the case, there might be times where the messages are not received by every receiver, meaning a stored copy is required to ensure everyone receives the original message.

15) In a two-phase commit protocol, blocking can never be completely eliminated, even when the participants elect a new coordinator because the participants are not only waiting on the coordinator but also the other participants. Since the coordinator must wait for a response for all the others, causing the coordinator to block causes the others to block by extension.

16) The totally-ordered multicasting using Lamport's logical clocks does not scale because of the messages, including the acknowledgments, that are received by the original sender. The sheer number of messages that return to the sender can bog down the system.

17) The authentication protocol is the following:

Person A sends data encrypted with public key and uses signature to verify identity
Person B receives, decrypts it with public key and uses signature to verify identity