**Question 1 (26 points):**

a. In class we saw 3 total ordering protocols, which we nicknames Sequencer, Orderer, and Rotating Token. For each such protocol, give one scenario in which it is likely to be better than the other two. Explain.

b. Extend the vector timestamp based causal broadcast protocol that we studied in class (in which each message is always broadcast to all group members) to enable causal ordering when each message can be sent to any subset of nodes, including point-to-point. Explain your solution.

**Question 2 (30 points):**

a. Describe a simple generic deterministic construction of a dissemination graph (overlay) in which the degree of each node as well as the diameter of the graph are both $\log(n)$. The construction should work for any value of $n$ (and not just powers of 2 like in a hypercube). Hint: start with a logical ring.

b. List 2 advantages and 1 disadvantage of a randomly constructed overlay like Araneola compared to an overlay like the one you were requested to construct in the previous question (2.a.).

**Question 3 (14 points):**

Consider the following caching scheme that is proposed for DHTs: Whenever a node $p$ that is responsible for a given key according to the DHT answers a request initiated by some other node $q$, cache the reply of $p$ in all nodes on the search path from $q$ to $p$. In which DHT do you expect this scheme to offer a more effective caching: Chord or Kademlia? Explain.

**Question 4 (30 points):**

a. In the quorum replication protocol we've studied, as well as in Dynamo, in order to survive $f$ failures, each write is being sent to $n=2f+1$ replicas. On the other hand, in GFS, as we talked in class, each write can be stored only on $f+1$ replicas. Explain this difference? Which design goal of GFS vs. Dynamo justifies one option vs. the other?

b. BigTable shards (partitions) tables into smaller tablets. From a distributed systems perspective, what is the main benefit of doing this? What does BigTable sacrifice because of this? Explain.

**Submission instructions:**

You must solve this exercise alone – submissions are individual. Solutions must be submitted through the course web site – either printed or a high-resolution scan. Solutions must be written in Hebrew unless you get an authorization from Prof. Friedman to submit in English.

The submission date is Monday 23/01/2017 before midnight.

**Good luck!**

© 2017 Computer Science Department, Technion – Prof. Roy Friedman and Dolev Adas