Chalmers Techniska Högskola
Data- och Informationsteknik

CTH TDA297
GU DIT290

Distribuerade system fk

Tentamen 2017-08-18

Dag, *Tid, Sal*: August 18th 2017, 14:00-18:00, M building

***Kursansvarig*:** Philippas Tsigas (Tel: 772 5409)

***Hjälpmedel*:** Inga

Totalt *Poängtal*: 60

***Betygsgränser*:**

CTH: 3:a 30 p, 4:a 38 p, 5:a 48 p

GU: Godkänd 30p, Väl godkänd 48 p

Instructions

- Please answer in English, if possible.
  If you have very big difficulty with that, though, you may answer in Swedish.
- **Do not forget to write your personal number and if you are a GU or CTH student and at which "linje".**
- Please start answering each assignment on a new page; number the pages and use only one side of each sheet of paper.
- Please write in a tidy manner and explain (briefly) your answers.
- Students must **not** write their personal number on the answer sheets since the exam is anonymous; they shall write that **only** on the name slip area that they will seal.

LYCKA TILL !!!!

1. (15 points) Moa is building a database that is replicated on N machines. To access the database, a client accesses any of the replicas. The communication between clients and the replicas and between the replicas themselves is reliable, point-to-point, and FIFO-ordered. However, the communication delays can vary significantly. Moa has designed the following variant of a totally-ordered multicast algorithm to be used on her system:

   > Each replica maintains Lamport's logical clock, and every inter-replica message is stamped with the unique id of the sender upon transmission. Whenever a replica receives a database update message from a client, it brodcasts the update in a message to all replicas (including itself). Whenever a replica receives an update message from another replica (or from itself) it puts the message into a local queue, and acknowledges the reception of the update by sending an acknowledgment message to all other replicas (including itself). The replica applies an update from its local queue to its local database if and only if:
   >
   > i) the update message has the lowest timestamp among the messages in the replica's local queue, and
   >
   > ii) the update message has been acknowledged by a quorum of N
   >
   > After a replica has applied the update to the local database, the update message and its acknowledgments are removed from the local queue. If later another acknowledgment arrives for the message, such a late acknowledgment is simply dropped from the queue.

   Prove that the above algorithm implements totally-ordered causal multicast (i.e. satisfies the following requirements: 1. Reliability: Integrity, Validity, Agreement 2. Ordering: Causal, Total-order) or produce a counter example.

2. (15 points) (8+7)

   - Describe a non-blocking algorithm that uses a special process that acts as a cordinator to solves the consensus problem on a system with process failures but no link failures.
   - Can you extend this algorithm to an algorithm that works also for systems with link failures? If yes describe the extension, if not describe a proof.

3. (10 points) Explain the difference between linearizability and sequential consistency.

4. (10 points) Describe an algorithm that computes a spanning tree of a network $G(V, E)$. How a node of the network can use the existence of such a spanning tree in order to broadcast information to all nodes of the network?

5. (10 points) Each statement is either true or false. A correct answer gives 1 point, a wrong answer gives -1 point, no answer gives 0 points. Overall you cannot get less than 0 points for this question.

   1. Leader election can be solved with a consensus algorithm.
      A. True B. False
   2. There exist a symmetric solution to solve leader election.
      A. True B. False
   3. The size of a quorum is always bigger or equal to the size of a majority set.
      A. True B. False

4. Atomic Broadcast is weaker than Consensus.
   A. True B. False
5. In an asynchronous system, where only one processor might crash, consensus is solvable.
   A. True B. False
6. Sequential Consistency is composable.
   A. True B. False
7. The Gossip architecture for replication was designed to provide highly available service.
   A. True B. False
8. The 3 Phase Commit protocol was introduced to improve the latency of the 2 Phase Commit protocol in executions where no faults take place.
   A. True B. False
9. There exist a consensus solution able to handle up to $f < n/3$ Byzantine failures.
   A. True B. False
10. If an algorithm is f-resilient, it must consist of at least $2f+1$ synchronous rounds, independently of the kind of system faults that it tolerates.
    A. True B. False