Resit Distributed Algorithms

VU University Amsterdam, 12 February 2014, 18:30-21:15

(Answers can be given in English or Dutch. You may use the textbook Distributed Algorithms: An Intuitive Approach*. Use of slides, handouts, laptop is* not *allowed.)*

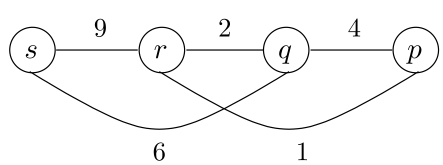
(The exercises in this exam sum up to 90 points; each student gets 10 points bonus.)

1. Propose an adaptation of the Chandy-Lamport snapshot algorithm (for directed networks with FIFO channels), in which basic messages may be buffered at the receiving processes, and the channel states of the snapshot are always empty.
(12 pts)

2. Suppose that the order in which resource requests are granted is predetermined. Give an example of a snapshot with a resource deadlock that is not discovered by the Bracha-Toueg deadlock detection algorithm.

   Show that in case of a nondeterministic selection which resource request is granted, the deadlock in your example may be avoided. (12 pts)

3. Run the Merlin-Segall algorithm on the following undirected weighted network, to compute all shortest paths toward process p. In your scenario, in the first round some node should first receive an improved value, and only then a message from its parent (emphasize this point in your description).



   Give the complete set of messages for the first round, and (only) the outcomes for the second and third round. (12 pts)

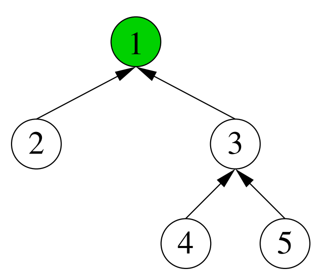
4. Argue why a congestion window may effectively double in size during every round trip time. (6 pts)

5. We adapt the Bracha-Toueg algorithm for t-Byzantine consensus by allowing a correct process to decide b if it accepts *at least* (instead of more than) $\frac{N+t}{2}$ b-votes in one update round.

   Consider a *complete* network G (i.e., there is a channel between each pair of different processes) of five processes. Let three processes hold the value 0, while two processes hold the value 1. Apply the adapted version of the Bracha-Toueg algorithm for 1-Byzantine consensus to G, and show that it can lead to inconsistent decisions. (12 pts)

6. In the t-Byzantine robust synchronizer of Lamport and Melliar-Smith, a correct process p accepts a local clock value of another process q if it differs no more than δ from its own clock value, at the moment of synchronization. Explain in detail why that synchronizer has precision $\frac{3t}{N}\delta$ (versus precision $\frac{2t}{N}\delta$ of the Mahaney-Schneider synchronizer). (14 pts)

7. Run Raymond's mutual exclusion algorithm on the graph below.



   Initially node 1 holds the token. Give a scenario (including all messages and message contents) in which first node 5, then node 4, and finally node 2 requests the token, but they receive the token in the opposite order. (12 pts)

8. Argue that in the Afek-Kutten-Yung self-stabilizing spanning tree algorithm, each join request eventually results in an acknowledgment. (10 pts)