Chalmers University of Technology

Computer Science and Engineering
Elad Michael Schiller

2020/08/27

Written exam in EDA387/DIT663 Computer Networks 2020-08-27. Exam time: 4 hours.

A remote exam via Canvas.

Examiner: Elad Michael Schiller, phone: 073-6439754

Credits:	30-38	39-47	48-Max
Grade:	3	4	5
Grade (GU)	G	G	VG

1. Exam problems that require text-based answers should be written and submitted in a text document, e.g. Word or PDF.
   - Create one text document for each such exam problem.
   - Name your text document Question_YY. *Example:* Question_01.pdf
   - Submit your answers by uploading the text documents via Canvas before the due date/time.
2. Exam problem solutions involving calculations, figures, diagrams etc. should be solved on paper as in a normal exam.
   - Make sure that each paper is clearly marked with your name, exam problem number and page number.
   - Scan or photograph your solutions. Make sure to have a good lightning and preferably use a document scanning app, e.g. even CamScanner or Genius Scan.
   - Name your image files Question_YY_Page_XX. *Example:* Question_01_Page_02.jpg.
3. If you want, you can combine images for the same problem into a single document (e.g. Word or PDF) named Question_YY.
4. Submit your solutions by uploading the image files or documents via Canvas before the due date/time.
5. The answer must be written in English (also for Swedish students). Use proper grammar and punctuation.
6. All answers need to be motivated, unless otherwise stated. Correct answers without motivation or with wrong motivation will not be given full credit. Answer concisely but explain all reasoning. Draw figures and diagrams when appropriate. Write clearly. Unreadable or hard-to-read handwriting will not be given any credit.
7. Do not hand in anything that is not part of the solution.

Question 1: Networking and design criteria (34 points in total)

1.a (6 points) In class we learned about self-stabilizing algorithms for constructing a distributed spanning tree. **(2 pt)** What rule does this algorithm serve in the construction of routing tables over the Internet? **(2 pt)** Please name the exact self-stabilizing algorithm and the routing protocol. **(2 pt)** Please explain how the algorithm is used and what for.

1.b (8 points) This part focuses on software-defined networks (SDNs). **(2 pt)** Explain what is the functionality that packet forwarding rules serve in SDNs. **(6 pt)** Do any of the algorithms considered in item 1.a can help to satisfy the task of constructing in-band routing in SDNs? If not, please explain why. If it can help, please provide detail regarding how it can be done.

1.c (4 points) IPv4 address has 32-bits whereas IPv6 address has 128-bits. **(2 pt)** Please explain the motivation for this change. **(2 pt)** Please clarify the limitations of the Internet without this change.

1.d (8 points) Suppose that the Internet address had 65536-bits already in version 1. **(2 pt)** In your opinion, how would the Internet have developed? **(2 pt)** What limitations the Internet would have had and what limitation it would not have had. **(2 pt)** Are there any new protocols that would be needed? **(2 pt)** Are there any existing protocols that would not be needed? Please specify these protocols, their names, and their functionality.

1.e (8 points) (4 pt) In what sense a solution for super-self-stabilizing systems is better than a self-stabilizing one? **(4 pt)** Should we just focus on super-self-stabilization instead of self-stabilization?

....

Question 2: designing algorithms for computer networks (26 points in total)

Consider a computer network that its topology is of a general graph. Assume that the network includes a single distinguished node. Also, the system is synchronous. That is, upon a pulse, all processors invoke a procedure simultaneously. These pulses mark the start of a new synchronous round.

This question considers the problem of self-stabilizing firing squad. The definition of the firing squad problem assumes that at any given round a processor may receive an external “go” signal, which is considered a request for the processors to simultaneously “fire.” The problem requires that: (a) if some processor fires in round r, all processors fire simultaneously in round r; (b) if a processor receives a go input in round r', it will fire at some later round $r > r'$, and (c) a processor fires in round r only if some processor received a go input in some round $r' < r$. It is also disallowed for a single go input to induce multiple firing events.

Your task is to design a self-stabilizing algorithm for solving the problem of the firing squad problem. For the sake of a simpler presentation, assume that shared memory model in which for every processor $p_i \in P$ can either have (i) a single register r_i that it can read and write and all neighbors $p_j \in N_i$ can only read r_i, or (ii) a register $r_{i,j}$ per neighbor $p_j \in N_i$.

2.a (10 points) Please provide a clear solution to the firing squad problem. Your algorithm can partially use other solutions learned in the course as long as you specify your sources clearly. Also, your proposed solution should include a pseudo-code for any part that does not use a solution that was learned during the course.

2.b (8 points) Please write a proof that shows recovery after the last occurrence of a transient fault. How long is the recovery period?

2.c (8 points) Please write a proof that shows after the recovery period, the system satisfies the problem requirements.

....