Exam in Distributed Systems

Justin Pearson 1DT641 1TT835

March 31st, 2008

Cover Sheet

This sheet should be handed in together with the exam.

Each problem must be solved on a separate sheet. Write your name on each sheet. Indicate below which questions you have answered.

Problem no.	Solution provided	Max	Your points
1		5	
2		5	
3		8	
4		7	
5		9	
6		6	
7		6	
8		5	
Total:		40/51	

Name : . . . .

Pers.no. : . . . .

1

Exam Rubric

A mark of 60% is required for a pass, a 4 and 5 are distributed evenly. 85% is required for a VG. Answer can be in English or Swedish.

Each full answer should be started on a separate sheet. Please write you name and personal number on each sheet.

Students taking from F4Sy that is students on the 3 point version of the course only do questions 1-6.

Hjälpmedel:

Pen, pencil, ruler, rubber (eraser for people from the Antipodes or from across the pond) calculator.

General Comments

I like examples and pictures. If you give interesting examples and draw useful pictures to illustrate any points that make. If you do all this you are more likely to get full points for the question.

Always show your workings when ever you do a calculation. If you do not show your workings then even if the answer is correct you will get 0 points.

1. Replication. When building a distributed system it is desirable to replicate resources. Describe two strategies to manage the copies. (**5 points**)

2. Clock Synchronisation

   (a) What is a leap second and why is it used? (**1 points**)

   (b) Explain Cristian's method for synchronising clocks. (**2 points**)

   (c) Suppose that you have two clocks both with drift $\rho = 10^{-8} seconds/seconds$. Suppose that you want the clocks synchronised within 0.5 of a second. How often should the clocks be resynchronised to achieve this? (**2 points**) (Obs. Show your workings, no workings no points for a correct or incorrect answer)

3. Logical Clocks

   (a) Consider the following three processes p_1, p_2 and p_3 with the following pattern of communication:
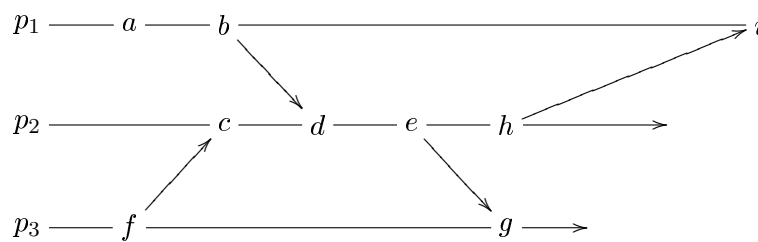
   Label the each event with a normal Lamport timestamp (**2 points**) and a vector timestamp (**2 points**)

   (b) What does it mean for two events to be concurrent and what is the relation of the Lamport timestamps of the two events. (**2 points**).

   (c) Is it possible for two events to have the same Lamport (non vector) timestamp? If it possible give an example, if it is not possible argue why it not possible. (**2 points**).

   (d) Why are vector timestamps required? (**2 points**)

4. Transactions.

   (a) With examples define what a transaction is. (**2 points**)

   (b) Explain what problems can happen there is no concurrency control where multiple transactions are being executed at the same time. (**3 points**) (Obs. **Give examples**).

   (c) Define what it means for an interleaving of two transactions to be serially equivalent? (**2 points**)

5. Two and Three Phase Commit

  - Explain the two phase commit algorithm. You should give the algorithms that the server and the clients implement and explain what happens. (**3 points**)

  - Suppose the server crashes during the two-phase commit algorithm. Explain if it possible and how the clients can come to an agreement. (**2 points**)

  - Explain the three phase commit algorithm and how it is an improvement over the two-phase algorithm. (**4 points**)

6. Consistent Cuts.

  - With examples define what a consistent cut is. (**3 points**)

  - Describe the snapshot algorithm and explain how it records consistent cuts. (**3 points**)

Stop here if you are a TF student.

7. Shared Memory

  (a) What is strict consistency and why is not possible to achieve in a distributed system? (**3 points**)

  (b) Define causal-consistency. You must give examples. (**2 points**)

  (c) Is the following data store causally consistent? Explain your answer. (**1 points**)

A	W(x)a		W(x)b		
B		R(x)a		W(x)c	
C				R(x)c	R(x)b

  (d) Explain with examples what release consistency is. (**2 points**)

8. Peer to peer and anonymous routing

  - Explain with a node in a Gnutella network finds the location of a file which is stored in another network. (**2 points**)

  - Describe some problems with Gnutella networks. (**1 points**)

  - Outline how onion routing works. (**2 points**)