TEKNISKA HÖGSKOLAN I LINKÖPING
Institutionen för datavetenskap
Petru Eles

Tentamen i kursen

Distribuerade System- TDDB 37

2000-03-13, kl. 14-18

Hjälpmedel:

Inga.

Poänggränser:

Maximal poäng är 40.
För godkänt krävs sammanlagt 21 poäng.

Resultat anslås:

Senast 2000-03-24 på IDAs anslagstavla för tentamensresultat.

Jourhavande lärare:

Petru Eles, tel 28 13 96

Good luck !!!

1. What means transparency in a distributed system? We have defined four aspects of transparency. Enumerate and explain at least three of them.

(2p)

2. Define the following three possible semantics for remote procedure calls:
   a. At least once semantics
   b. At most once semantics
   c. Exactly once semantics.

   Is it possible to achieve *exactly once semantics* in the case of lost messages? But in the case of server crashes? Explain.

(4p)

3. The Object Request Broker (ORB): what is its role in an object oriented distributed system? Illustrate also by a figure. (The question is not related directly to CORBA, but if you want to refer to the CORBA ORB it's fine).

(3p)

4. What is a cut of a distributed computation? What means a consistent and a strongly consistent cut? Consider the following set of events:
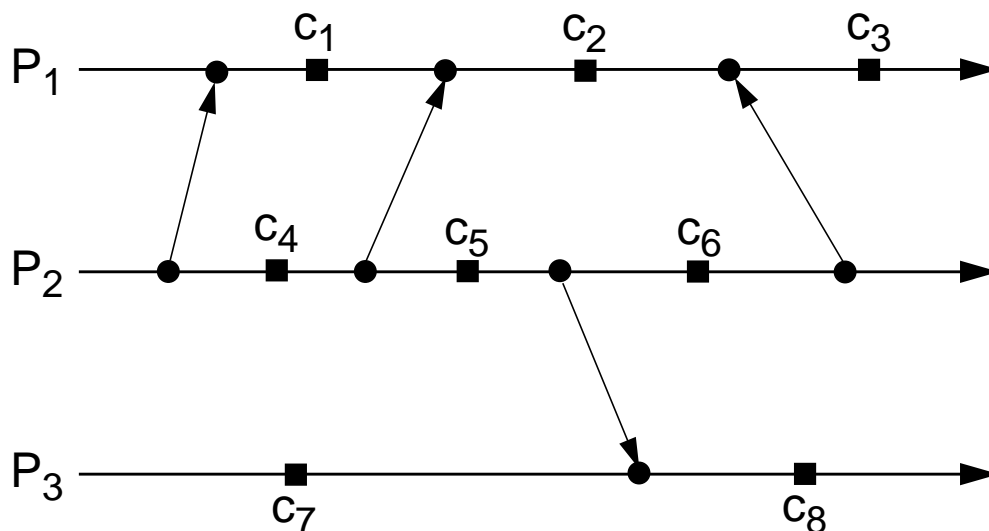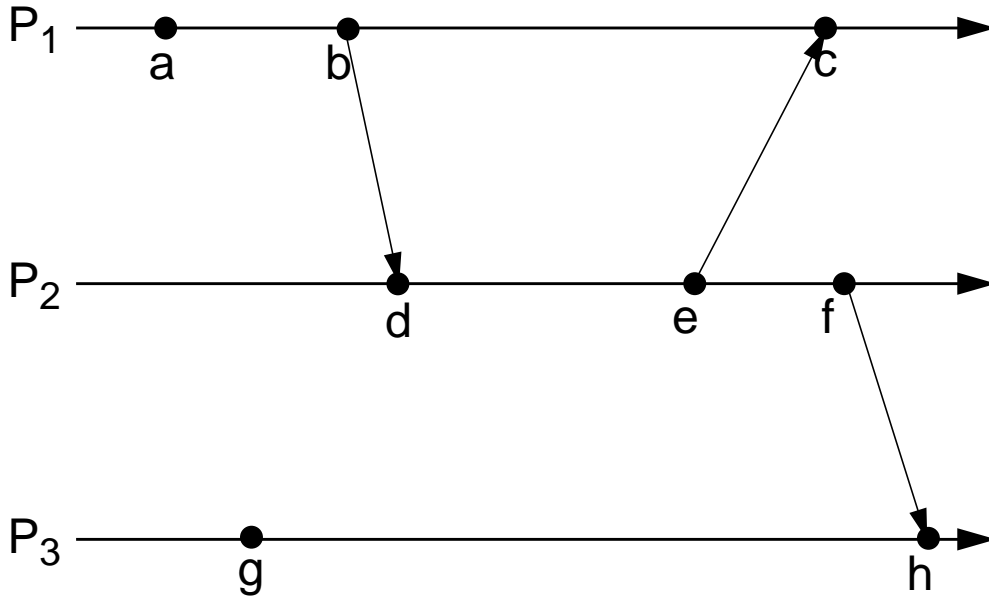

Determine for each of the following cuts if it inconsistent, consistent or strongly consistent:
$\{c_2, c_6, c_8\}$, $\{c_1, c_4, c_7\}$, $\{c_1, c_5, c_7\}$, $\{c_1, c_6, c_8\}$, $\{c_1, c_6, c_7\}$, $\{c_3, c_6, c_8\}$.

(3p)

5. Define the *happened before* relation (Lamport).
    When are two events concurrent?

What is the relation between events a - b, a - d, b - g, g - e, d - c, b - c, a - g, f - g, e - h, a - h?
(3p)

6. What is the basic idea behind the token based distributed mutual exclusion algorithm by Ricart-Agrawala (the second algorithm)? Consider how mutual exclusion is guaranteed and how the token is passed after a process has left the critical section. How many messages are passed in order a process to get permission to a critical section? Compare to the first algorithm by Ricart-Agrawala (which is not using a token).
(4p)

7.

a. Define total and causal ordering of requests. Illustrate by an example.

b. How can total ordering be implemented using a central sequencer?

c. Consider total ordering based on distributed agreement with one front end and several replication managers, like below:
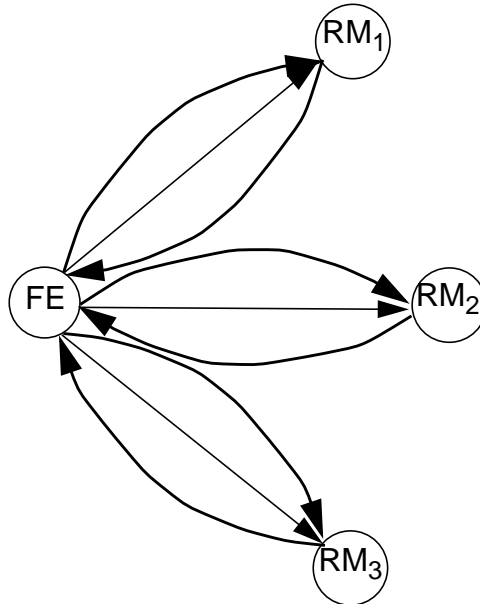
Three messages have to be exchanged by the FE with each RM. Explain. What do this messages contain?

(4p)

8. What is a fault tolerant system?
    Explain the following fault types (fault models):
    - fail-stop
    - slowdown
    - byzantine.

(3p)

9. What is the basic idea with voting protocols for updating replicated data? How do they work?

    Consider a set of 15 replica managers. Define two voting protocols. One for a situation when the number of writes is relatively large compared to that of reads, and the other for the reverse situation. Give examples of read and write quorums (use figures).

(3p)

10. The Byzantine Generals Problem: show how agreement is not or is possible for three and for four generals respectively, in the case one of the generals (not the commander) is a traitor (illustrate the exchange of messages with figures).

(3p)

11. What does it mean by external and internal synchronization of physical clocks?

(2p)

12. Both the Berkeley algorithm and distributed algorithms for clock synchronization are based on collecting current values of clocks and calculating an average. What is the difference between them?

(2p)

13. You now the maximum drift rate of the clocks on two processors and the maximal allowed skew between them. How do you determine the maximum interval between two successive synchronizations between the clocks?

(2p)

14. Compare the Ethernet protocol and the CAN protocol from the point of view of predictability. Explain.

(2p)