Distribuerade system fk

*Tentamen 1999-08-*19

Dag, *Tid, Sal*: Augusti 19 1999, 8:45-12:45, MG

***Kursansvarig*:** Philippas Tsigas(Tel: 772 5409, h. 7117763)

***Hjälpmedel*:** Inga

Totalt *Poängtal*: 60

***Betygsgränser*:**

CTH: 3:a 24 p, 4:a 36 p, 5:a 48 p

GU: Godkänd 28p, 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.

LYCKA TILL !!!!

1. *12 marks*

   (a) Describe the two-phase commit protocol.

   (b) Describe the use of the two-phase commit protocol and its drawbacks that led to the introduction of the three-phase commit protocol.

2. *12 marks*

   (a) Describe in few words the "Synchronous Doorway" mechanism that is used by the Choy and Singh algorithm for resource allocation. Point also why this mechanism is not enough to solve the resource allocation problem by itself.

   (b) Describe in few words the “Asynchronous Doorway” mechanism that is used by the Choy and Singh algorithm.

3. *12 marks*

   (a) In the State machine approach for replication each replica processes the stable request with the least unique id. What is the definition of stable request?

   (b) How can a replica use logical clocks to determine the stability of a request in a fail-stop system?

4. *12 marks*

   (a) Describe the simple Sequential Consistency Scheme that is used in IVY (IVY is the simple page-based DSM system).

   (b) In the above scheme what are the possible ways to find the page’s owner?

5. *2 marks*

   In a network G=(V,E) the algorithm below is used for broadcasting from processor P1 the information (X). What is the Time and Communication Complexity of this algorithm?

   Protocol for P1 :

   ---

   send(X) to all neighbouring processes.

   Protocol for All Other Processes:

   ---

   when receive(X) from a neighbour
   if first ***then*** /*initially first:=true */ first:=false send(X) to all neighbours

   ---