(M.Sc.Eng.-course)

Written examination, 19 May 2016

Course: Distributed Systems Course no. 02220

Aids allowed: no aid

Exam duration: 4 hours

Weighting: TOPIC 1: 32%, TOPIC 2: 28%, TOPIC 3: 22%, TOPIC 4: 18%

Answers are evaluated according to correctness, completeness and conciseness (i.e., answers must be correct, complete and short). In case you think some specification is missing or ambiguous, just state your own specification/understanding and continue to solve the problem according to that.

The problems start on Page 2.

TOPIC 1 (32%): Logical Time and Global State

PROBLEM 1.1 (10%)

Let us consider a distributed system composed of three processes p_1, p_2 and p_3. An example of the progress of a possible computation of the system is shown in Figure 1, where each event e is described by 5 components $< p, s, s', M, c >$, meaning that process p goes from state s to state s' because a message M was sent or received on channel c.

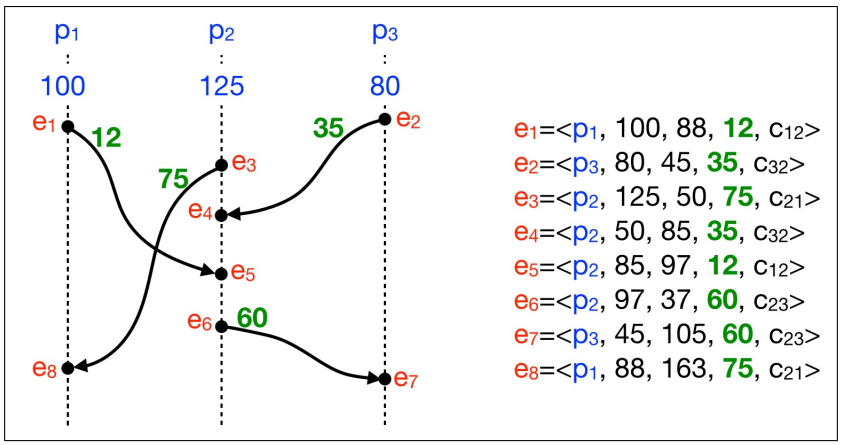

Figure 1: Computation of the Distributed System

Question 1 (2%): Using the computation in Figure 1, discuss an example of *inconsistent set of events* by:

a) drawing an *inconsistent cut* defining the inconsistent set

b) explaining why the set is *inconsistent*.

Let us suppose *Chandy and Lamport's snapshot algorithm* is initiated by process p_3 just before event e_2 in the computation in Figure 1.

Question 2 (2%): Sketch how markers would be exchanged during the execution of the algorithm in this case.

Question 3 (2%): Which events are included in the resulting consistent set H of events?

Question 4 (2%): Which state components are noted down in the various processes, as the execution of the algorithm proceeds?

Question 5 (2%): Which global state S^* is discovered by the algorithm in this case?

 ... continued on the next page

PROBLEM 1.2 (12%)

Let us consider the computation in Figure 2, where x and y are local variables of processes p_1 and p_2, respectively.

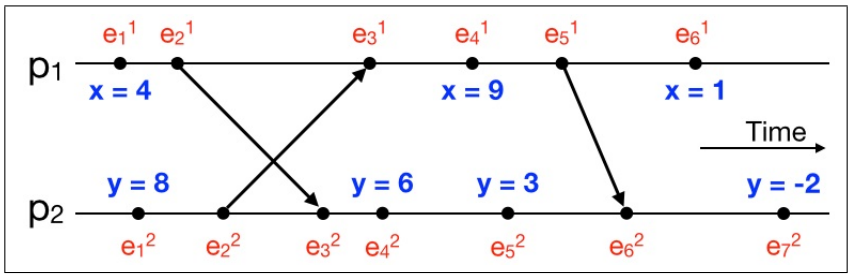

Figure 2: Computation of the Distributed System

Question 1 (8%): Construct the lattice of global states for the computation.

Question 2 (2%): Check whether $Definitely(x + y = 12)$ is *True* or *False* in the computation. In case of *True*, indicate the global states in which the property holds. In case of *False*, explain why the property does not hold.

Question 2 (2%): Check whether $Possibly(x - y = 2y)$ is *True* or *False* in the computation. In case of *True*, indicate the global states in which the property holds. In case of *False*, explain why the property does not hold.

PROBLEM 1.3 (10%)

Let $\rightarrow$ be the Lamport *happened-before* relation, and $L(e)$ denote the Lamport timestamp of event e at whatever process it occurred at. By considering a chain of zero or more messages connecting events e_1 and e_2 and using induction on the length of any sequence of events relating e_1 and e_2, prove that $e_1 \rightarrow e_2 \Rightarrow L(e_1) < L(e_2)$.

... continued on the next page

TOPIC 2 (28%): Multicast Communication

Let us consider ordered multicast in distributed systems. Well known ordering delivery requirements are *total ordering*, *causal ordering* and *FIFO ordering*.

PROBLEM 2.1 (6%)

For each of the three ordering delivery requirements, define the delivery requirement and draw an example of computation in which the requirement is met.

PROBLEM 2.2 (12%)

For each of the following 6 statements indicate whether the statement is true or false. In the latter case (that is, the statement is false), justify your answer by means of a counterexample.

Statement 1 (2%): Total ordering implies FIFO ordering.

Statement 2 (2%): FIFO ordering implies total ordering.

Statement 3 (2%): Total ordering implies causal ordering.

Statement 4 (2%): Causal ordering implies total ordering.

Statement 5 (2%): FIFO ordering implies causal ordering.

Statement 6 (2%): Causal ordering implies FIFO ordering.

PROBLEM 2.3 (10%)

Show informally that if two processes use a FIFO-ordered variant of B-multicast, then the totally ordered multicast is also causally ordered.

 ... continued on the next page

TOPIC 3 (22%): Distributed Mutual Exclusion

PROBLEM 3.1 (4%)

Define the problem of mutual exclusion in distributed systems and provide an example.

PROBLEM 3.2 (6%)

Describe an algorithm solving the problem of mutual exclusion in distributed systems, providing also an example (that is, one execution of the chosen algorithm).

PROBLEM 3.3 (6%)

Let us consider the token-based central server algorithm for distributed mutual exclusion. Let us assume that the server is correct (that is, it does not crash and it well behaves) and that the server can use a *reliable failure detector* to determine whether a client is crashed or not.

Question 1 (2%): Comment on how to adapt the algorithm to handle the crash failure of any client (in any state).

Question 2 (2%): Comment on whether the resultant system can be considered fault-tolerant.

Question 3 (2%): Let us assume that the failure detector is *unreliable* instead of *reliable*. What would happen if a client that possesses the token is wrongly suspected to have failed? Discuss by means of an example.

PROBLEM 3.4 (6%)

Lamport's algorithm (1978) for distributed mutual exclusion requires communication channels to deliver messages in FIFO order. By means of a counterexample, discuss why the algorithm needs such assumption to work correctly.

 ... continued on the next page

TOPIC 4 (18%): Peer-to-Peer (P2P) Computing

PROBLEM 4.1 (6%)

Discuss the key differences between client-server and p2p systems, in particular in terms of symmetry/asymmetry, global/local knowledge of the system, centralization/decentralization, robustness, scalability, costs.

PROBLEM 4.2 (2%)

Describe the concept of overlay networks in p2p computing (what is an overlay network? What are the advantages of using overlay networks in p2p systems?).

PROBLEM 4.3 (2%)

Describe the data location (lookup) problem in p2p computing.

PROBLEM 4.4 (8%)

Describe one of the strategies to store and retrieve data in p2p computing (thus, solving the data lookup problem) and highlight advantages and disadvantages of the chosen strategy (for instance, in terms of search complexity, storage cost, scalability, ...).

End of the Exam