UMEÅ UNIVERSITY, DEPARTMENT OF COMPUTING SCIENCE

Written Assignment 2 (40 points)

Distributed Systems - 5DV147

You need to complete and submit this assignment individually. Collaboration is permitted only as described in the Guidelines for compulsory assignments and the Honor code. You need to submit this assignment by December 13 at 23:59 (hard deadline). You can either email your solution to us (5dv147-staff(at)cs.umu.se) or drop off your solution in the box outside the department.

1 GLOBAL STATES (8 POINTS)

Two processes P and Q are connected in a ring using two channels, and they constantly rotate a message m. At any time, there is only one copy of m in the system. Each process's state consists of the number of times it has received m. Assume that P sends m first. At a certain point, P has the message and its state is **101**. Immediately after sending m, P, initiates the snapshot algorithm. Explain the operation of the algorithm in this case, giving the possible global state(s) reported by it.

2 MUTUAL EXCLUSION (14 POINTS)

1. In a certain system, each process typically uses a critical section many times before another process requires it. Explain why Ricart and Agrawala's multicast-based mutual exclusion algorithm is inefficient for this case, and describe how to improve its performance. Does your adaptation satisfy the liveness condition for mutual exclusion? (7 points)
2. Modify the Ricart and Agrawala algorithm to permit up to K simultaneous entries into the critical section. Does your algorithm satisfy the requirements for mutual exclusion (safety, liveness, and $\rightarrow$ ordering)?(7 points)

3 ELECTIONS (8 POINTS)

In class we studied the **Bully** algorithm for elections. You can also take a look at the course book, to know how the algorithm works. After taking a look at the book answer the following questions (4 points each).

1. In the **Bully** algorithm, a recovering process starts an election and will become the new coordinator if it has a higher identifier than the current coordinator. Is this a necessary feature of the algorithm? (Explain)

2. Suggest how to adapt the Bully algorithm to deal with temporary network partitions (slow communication) and slow processes.

[DS5] Coulouris G., Dollimore J., Kindberg T. and Blair G.: *Distributed Systems - Concept and Design.* Fifth edition. Addison Wesley (2005)

4 MESSAGE ORDERING (4 POINTS)

In a multi-user game, the players move figures around a common scene. The state of the game is replicated at the players' workstations and at a server, which contains services controlling the game as a whole, such as collision detection. Updates are multicast to all replicas. Consider the following conditions:

- The figures may throw projectiles at one another, and a hit debilitates the unfortunate recipient for a limited time. What type of update ordering is required here? *Hint*: consider the *throw*, *collide*, and *revive* events.

- The game incorporates magic devices that may be picked up by a player to assist them. What type of ordering should be applied to the *pick-up-device* operation?

5 GROUP COMMUNICATION (6 POINTS)

Show that the FIFO-ordered multicast algorithm presented in the course book does not work for overlapping groups, by considering two messages sent from the same source to two overlapping groups, and considering a process in the intersection of those groups. Adapt the protocol to work for this case. *Hint*: processes should include with their message the latest sequence numbers of messages sent to all groups. (6 points)

[DS5] Coulouris G., Dollimore J., Kindberg T. and Blair G.: *Distributed Systems - Concept and Design.* Fifth edition. Addison Wesley (2005)