

# *ECE 6102 Take-Home Exam 1*

Due Date: October 13, 2010; 4:30 PM

**Instructions:** This exam is to be completed on your own without discussing it with any one other than the Professor. You may use any reference materials you like in completing the exam but you must list the materials that you consulted below. It is not necessary to list class notes nor the papers distributed to the class as reference materials.

I, \_\_\_\_\_, hereby state that I did not discuss this exam with any person other than Professor Blough and that I used only the following reference materials in preparing my solution:

---

(Signature)

---

(Date)

1) 25 points

Consider a distributed system having 2 processes, P0 and P1, with 6 sequential events on each process. All pairs of events, where one event comes from each process, are concurrent, i.e. events are ordered only within one process but can be arbitrarily ordered between processes. How many *distinct* global states are there in this system, beginning from the initial state where no events have occurred and ending with the final state where all 12 events have occurred? Two states reached through the same sequence of events, but in different orders, are *not* considered distinct states. For example, the state reached by event 0 on P0 followed by event 0 on P1 is the same state reached by event 0 on P1 followed by event 0 on P0. Make sure to show how you calculated the number of distinct states – answers without sufficient work will not receive credit.

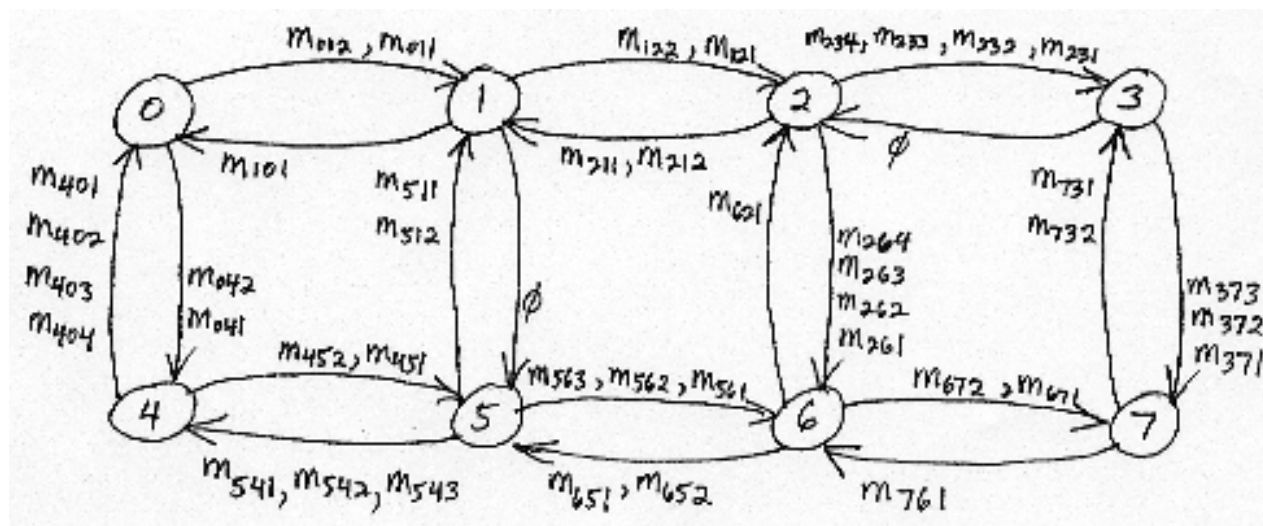
2) 25 points

Consider execution of the Chandy, Lamport distributed snapshot algorithm on a system with the following characteristics. The system operates in a synchronous fashion where during a single time step, each process experiences one event if at least one event is pending. An event is any of the following:

1. spontaneously initiating the snapshot algorithm without receiving a marker message (this event includes saving the state of the process that initiates on stable storage)
2. saving the state of the process on stable storage (after receiving a marker message)
3. saving the state of a channel on stable storage
4. sending a marker message on all outgoing channels
5. receipt of a message on a single channel

When multiple events are pending at a process, the process prioritizes the events in the order given, where lower numbers indicate higher priorities. Also, whenever incoming messages are pending on multiple channels, the channels are read in alternating fashion according to the following order which repeats in a cyclic fashion: left incoming channel, right incoming channel, vertical incoming channel.

Trace execution of the snapshot algorithm on the system shown below. At the end of the execution, show the state that was recorded for each channel in the system. Assume that process 1 spontaneously initiates the algorithm at time step 0 and that process 7 also spontaneously initiates at time step 2. Assume that no additional messages are sent by the processes during execution of the algorithm except for marker messages.



3) 25 points

Prove that the algorithm for atomic broadcast in synchronous systems presented in lecture, Algorithm A-Broadcast-Synchronized, provides both FIFO and causal message delivery under the assumptions stated, i.e. that:

- only crash faults occur,
- at any real time, clocks on two different correct processes do not differ by more than  $\epsilon$
- clocks on correct processes are monotonically increasing and do not skip any scheduled event times, and
- $\Delta \geq \epsilon + d_{\max}$ , where  $d_{\max}$  is the maximum delay between  $R_{i,j}$  broadcasting a message from one process and  $R_{i,j}$  delivering that message on any correct process

4) 25 points

Design an asymmetric threshold masking Byzantine quorum system with minimum load, assuming that the fraction of total operations that are reads is  $c$  (and therefore the fraction of write operations is  $1-c$ ). You must specify the read quorum and write quorum sizes for your system, state what the load is, and justify why the load is minimum. You must also prove that your system is a valid masking quorum system, i.e. that properties M-consistency and M-availability are satisfied.