Problem Set 1: Sanpshots, Mutual Election, Leader Election, Multicast

1. In the central server algorithm for mutual exclusion, describe a situation in which two requests are not processed in happened-before order.

2. Adapt the central server algorithm for mutual exclusion to handle the crash failure of any client (in any state), assuming that the server is correct and given a reliable failure detector. Comment on whether the resultant system is fault tolerant. What would happen if a client that possesses the token is wrongly suspected to have failed?

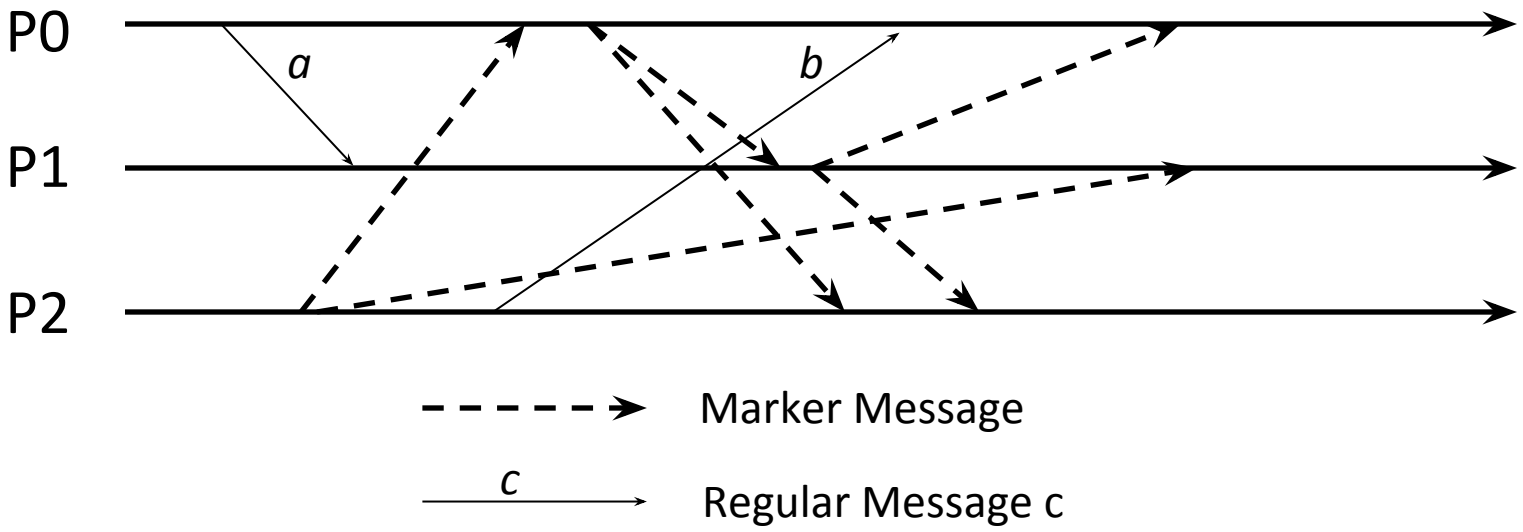
3. In the Bully algorithm, a recovering process starts an election and will become the new coordinator if it has a higher identifier than the current incumbent. Is this a necessary feature of the algorithm?

4. Show that the FIFO-ordered multicast algorithm 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 messages the latest sequence numbers of messages sent to *all* groups.

5. Suggest how to adapt the causally ordered multicast protocol to handle overlapping groups.

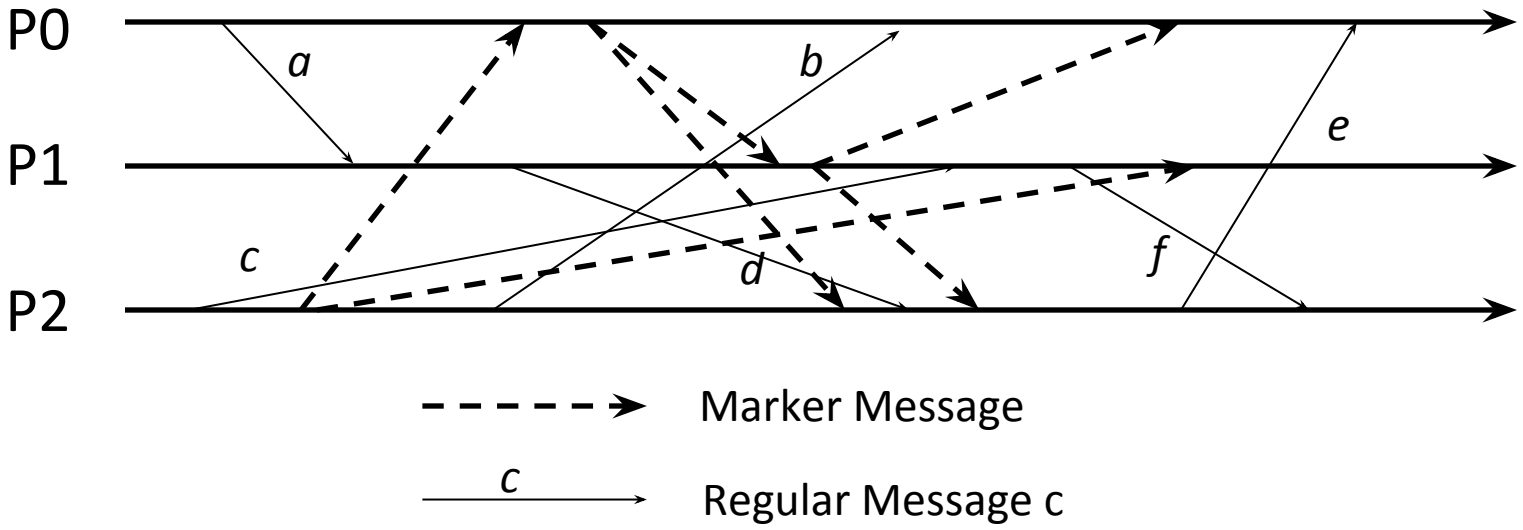
I. Chandy-Lamport Global Snapshots Algorithm

- Mark the entire global snapshot collected.



II. Chandy-Lamport Global Snapshots Algorithm

- Mark the entire global snapshot collected.



III. Mark all Lamport timestamps for application messages on this figure for all events.
All Lamport timestamps start from zero.