1) Suppose in a distributed system, the synchronization error $\epsilon$ is 0, the processor-to-processor datagram delay $\delta$ is 1 minute, the processor-to-processor atomic broadcast delay $\Delta$ is 10 minutes, and the check-in period $\pi$ is 30 minutes. Suppose a server-group consists of five members. (a) For each of the three protocols in Cristian's paper, give the reconfiguration latency $D$ when a single failure occurs. (b) Suppose the five members in the above group are $P_1, \ldots, P_5$, and their successors are, respectively, $P_2, \ldots, P_5, P_1$. The group was initiated by $P_1$ at time 7:00. If at time 7:20, $P_2$ crashes, when will $P_3$ and $P_4$ detect that a processor crash failure might have occurred according to the attendance list membership protocol and the neighbor surveillance protocol? (Assume no joins and other failures occur between 7:00 and 12:00).

2) What is the worst-case multiple-failure reconfiguration latency of the attendance list protocol if the last member of the ring (whose successor initiates the attendance list) does not fail? Justify your answer and give a worst-case scenario.

3) Consider the following criteria in choosing a membership protocol in a distributed system: (i) Assume at most one failure can occur between the formation of successive server-groups. (ii) The message overhead should be as low as possible. (iii) The reconfiguration latency $D$ must be bounded by a constant no matter how many processors are in a group. Which of the three protocols discussed in Cristian's paper would you adopt? Justify your answer.

4) In the case of a single failure, the neighbor surveillance protocol has a shorter reconfiguration latency than the attendance list protocol, but the reverse is true in the case of multiple failures. We want to design a new protocol that combines these two protocols as follows. Suppose there are $n$ processors: $P_0, \ldots P_{n-1}$, and let $n = km$, for some $k$ and $m$. The $n$ processors are further divided into $k$ subgroups such that $G_i$, the $i^{th}$ subgroup consists of the processors $\{P_{(i-1)m}, P_{(i-1)m+1}, \ldots, P_{(i-1)m+m-1}\}$. Processors in a subgroup use the neighbor surveillance protocol to maintain membership. The attendance list protocol is used by the leaders of the $k$ subgroups to maintain membership, where the leader of the $i^{th}$ subgroup is the lowest ranked surviving processor in the subgroup, i.e., when there is no failure, $P_0, P_m, \ldots P_{(k-1)m}$ form a virtual ring under the attendance list protocol. Death of a leader is detected by the neighbor surveillance protocol and the new leader initiates a new group under the attendance list protocol. You may assume that the datagram delay is in $(0, \delta]$, and the broadcast delay is $\Delta$. What is the reconfiguration latency of this protocol (a) when there is a single failure? (b) when there are multiple failures? Justify your answer. (Note: If you think that there is some flexibility in the interpretation of this protocol, make sure you state your additional assumptions clearly in your answer.)

5) Suppose in a distributed system of $n$ processors, the synchronization error is $\epsilon$. Processors can communicate by synchronous atomic broadcast as well as by datagram. The datagram delay is upper bounded by $\delta_1$ and the atomic broadcast delay is upper bounded by $\delta_2$. Consider the following membership protocol: Join handling is the same as in Cristian's paper. Failure handling is done as follows. Every $\pi$ time units after a new group is formed, each member sends a "present" message to the leader (highest-ranked member). The leader checks the "present" messages to determine if someone is missing. If nobody is missing, the leader broadcasts an "all is well" message to the group. Otherwise, it broadcasts the membership of the remaining members who then constitute a new group. If the members do not receive a
broadcast message from the leader within a certain deadline, then all members will send "present" messages to the second highest-ranked member who is the new leader, and so on. For this protocol to work, (a) when is the earliest time that the leader may broadcast to the group? What is the deadline for the leader’s broadcast message to reach all members? (b) what is the reconfiguration latencies of this protocol under single and multiple failures, assuming \( k \) is the number of failures and \( n > k > 0 \)? (c) what is maximum/minimum number of messages that are needed to achieve membership agreement when multiple failures can happen? (d) suppose we let any member that does not receive the broadcast from the leader by the deadline issue a "new group" request, how do the answers to parts (b) and (c) change? For this problem, assume that the message count for each datagram is 1 while the message count for a broadcast is the number of datagrams that are received by the remaining good processors.