<u>CS171- HW1</u>

1) If events corresponding to vector timestamps Vt_1, Vt_2, ..., Vt_n are mutually concurrent, then prove that,

   $(Vt_1[1], Vt_2[2], ..., Vt_n[n]) = max(Vt_1, Vt_2, ..., Vt_n)$.

2) If events ei and ej respectively occurred at processes pi and pj and are assigned vector timestamps VT_{ei} and VT_{ej} , respectively, then show that

   $ei \rightarrow ej \iff VT_{ei}[i] < VT_{ej}[i]$

3) Consider the following simple method to collect a global snapshot (it may not always collect a consistent global snapshot): an initiator process takes its snapshot and broadcasts a request to take snapshot. When some other process receives this request, it takes a snapshot. Channels are not FIFO.

   Prove that such a collected distributed snapshot will be consistent iff the following holds (assume there are n processes in the system and Vt_i denotes the vector timestamp of the snapshot taken process p_i):

   $(Vt_1[1], Vt_2[2], ..., Vt_n[n]) = max(Vt_1, Vt_2, ..., Vt_n)$.

   Don't worry about the channel states.

4) Consider a distributed system where every node has its physical clock and all physical clocks are perfectly synchronized. Give an algorithm to record global state assuming the communication network is reliable. (Note that your algorithm should be simpler than the Chandy-Lamport algorithm).
5) What modifications should be done to the Chandy-Lamport snapshot algorithm so that it records a strongly consistent snapshot (I.e., all channel states are recorded empty).