1. [10 points] For each pair of timestamps below, indicate which of the listed properties could apply to the events corresponding to those timestamps (i.e., there is some execution in which the property holds), and which of the properties can never hold. Justify your answers. (Assume the timestamps are updated using the corresponding algorithms that we studied in class.)

(a) (Lamport timestamps): Event $a$ has timestamp 7 and event $b$ has timestamp 1.
   i) Event $a$ happens before event $b$.
   ii) Event $b$ happens before event $a$.
   iii) Events $a$ and $b$ are concurrent.

(b) (Vector timestamps): Event $c$ has timestamp (3,2,0) and event $d$ has timestamp (2,1,1).
   i) Event $c$ happens before event $d$.
   ii) Event $d$ happens before event $c$.
   iii) Events $c$ and $d$ are concurrent.

(c) (Matrix timestamps): Event $e$ has timestamp $\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 0 & 0 & 4 \end{bmatrix}$ and event $f$ has timestamp $\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 2 & 4 \end{bmatrix}$.
   i) Event $e$ happens before event $f$.
   ii) Event $f$ happens before event $e$.
   iii) Events $e$ and $f$ are concurrent.

2. [10 points] Consider an asynchronous distributed system with three processes $p_1$, $p_2$, and $p_3$. The network is fully connected (every process can send messages to every other process), messaging is reliable, and all communication channels are FIFO. The processes use Lamport’s mutual exclusion algorithm to grant access to a resource. Suppose that $p_1$ requests the resource and is granted access. While $p_1$ is accessing the resource, $p_2$ requests access to the resource, and then shortly after that, $p_3$ requests access to the resource. After waiting a while, $p_3$ wants to check on its request for the resource, so it initiates the Chandy-Lamport global snapshot algorithm to learn the state of the system, i.e., the contents of the request queue at each process and which process is accessing the resource, if any. At the same time, $p_1$ releases the resource and sends the RELEASE message to the other processes, triggering additional steps in the mutual exclusion algorithm. Describe the execution of the Chandy-Lamport algorithm in this setting. What messages are exchanged, and what process and channel states are recorded? (Note: there may be more than one correct answer to this question.)

3. [10 points] In class, we studied Maekawa’s grid quorums for mutual exclusion. To ensure safety, we required that every pair of quorums intersects, i.e., $S_i \cap S_j \neq \emptyset$ for all $i, j = 1 \ldots N$. Quorums are also used in databases to control read and write access to the data. In such systems, there are two types of quorums, read quorums (denoted $R_i$, $i = 1 \ldots N$) and write quorums (denoted $W_i$, $i = 1 \ldots N$). To read data from the database, a process needs permission from a read quorum, and to modify/write to the database, a process needs permission from a write quorum. The following quorum properties are required for safety in this setting (no other properties are required for safety):

- $W_i \cap W_j \neq \emptyset$ for $i, j = 1 \ldots N$
- $W_i \cap R_j \neq \emptyset$ for $i, j = 1 \ldots N$. 

Suppose that we use Maekawa’s grid quorum scheme to determine the write quorums. Design a scheme to determine the read quorums that generates read quorums that are smaller than the write quorums. In your answer, be sure to state the quorum size and show that your scheme satisfies the properties above.