1. (20 pts) Consider a system where processes can be dynamically created (or terminated). A process can generate a new process. For example, in Figure 1 $P_1$ generates both $P_2$ and $P_3$. Modify the happened-before relation and the linear logical clock scheme for events in such a dynamic set of processes.
2. (40 pts) Assume that up to **two processes** can enter a critical section simultaneously. Provide possible extensions to Lamport and the simple token-ring-based algorithms. (Do not write code.)
3. (40 pts) You are required to evaluate a polynomial

\[ a_0 + a_1x^1 + \ldots + a_{n-1}x^{n-1}. \]

Design a DCDL algorithm with \textbf{n processes}. Assume that initially \( P_i \) (0 \( \leq i < n \)) has \( a_i \) and \( x \). The final evaluation result should be placed in \( P_{n-1} \). Try to minimize the number of multiplications.
EXTRA POINTS:

4. (10 pts) To apply the Chang and Roberts’s election algorithm to a hypercube, one can first generate a spanning ring in the given hypercube (see Figure 2 for a 3-dimensional hypercube example.)

Assume that one process initiates an election process at a time. In the worst case, almost 2 rounds are needed to elect a winner. Enhance Chang and Robert’s algorithm to obtain a faster election process for the hypercube topology by using multiple paths provided by the hypercube. Assume that each node can send a message to multiple neighbors simultaneously. Only high-level description of your algorithm is needed. Use the 3-dimensional hypercube example to illustrate your approach.
Figure 1: $P_1$ generates $P_2$ and $P_3$

Figure 2: A spanning ring in a 3-dimensional hypercube