Question 1 (20 points)
Consider a strongly connected distributed system consisting of n processes 0 through n-1. At the middle of the computation, the states of one or more processes were corrupted, so they tried to reset the system without stopping any process.

Assume that the state to which the system should be reset to \( R = (s_0, s_1, s_2, \ldots, s_{n-1}) \), where \( s_k \) is the fragment of the global state owned by process \( k \). If the reset operation is correctly performed, then all states reachable from \( R \) should be reachable after the reset is competed.

Modify the distributed snapshot protocol by Chandy and Lamport to perform a distributed reset.

Question (20 points)
Consider the following network where every link is unidirectional and FIFO, and messages always propagate along the direction of the links.
To record the global state, node A decided to send a robot. The robot will carry a briefcase, and will visit the processes in the order A-B-C-D-A. At each node, the robot will record the local state, and put it in its briefcase. When the robot returns to node A, the briefcase will contain the global state of the system.

(a) If the robot naively record the state of each node during its visit to a node, then will the snapshot be correct? Explain.
(b) What needs to be done so that the robot always collects a correct snapshot when it returns to node A?

**Question 3 (10 points)**
Revisit Dijkstra-Scholten’s termination detection algorithm.
(a) Explain with examples how different runs of the algorithm can generate different spanning trees.
(b) If the basic algorithm (whose termination is to be detected) uses $M$ messages, and each message is synonymous with a signal, then what is the message complexity of the termination detection algorithm? Briefly explain