Assignment - 2
Submission : 26th November -11:59pm

<u>Submission format</u>

A single zip file containing individual python files for each implementation and a brief report documenting and explaining your observations.

Part 1: Reliable Data Transfer (RDT)

Implement a simple stop-and-wait Reliable Data Transfer 2.2 protocol. A simulated data channel can be used for data transfer which has an associated packet loss probability as given in the **baseline.py** file. Implementation should cover sending data and displaying the received data, receiving ACK for successful transmission, and retransmission in case the packet is not received/ suitable ACK is not received. Print statements must be given as mentioned in the baseline.py file and the output must be documented in the report.

Part 2: Congestion Control

Implement a simple congestion control algorithm, such as Additive Increase Multiplicative Decrease (AIMD), to handle congestion in a network. Do print the congestion window size in order to verify the working of AIMD. To introduce congestion into the network, you can add 'loss probability' while sending and receiving packets. Try different variations of the slow start threshold to see what works best.

Part 3: Link State Algorithm

Implement the Link State Algorithm to determine the shortest path between nodes in a network. Your implementation should include the following functions:

a) build_topology(): This function constructs the network topology by reading the connectivity information from a file.

b) calculate_shortest_path(source, destination): This function calculates the shortest path between the given source and destination nodes using the Link State Algorithm.

Note: Follow the structure given in baseline.py for all three questions and implement the corresponding functions given. Ensure that you include comments in your code and briefly explain the algorithmic approach used for each component in the report.

After implementing the above algorithms, run the following test case:

Testing

- For the RDT implementation, send 10 packets which are given in the test_rdt file, and receive the data at the receiver.
- For the congestion control implementation, simulate the transmission of 100 data packets and receive acknowledgments. Display the congestion window size at each step.
- For the link state algorithm, use the network topology file given to build the network. Calculate and print the shortest path from Node A to Node I.