CS 342: Computer Networks Laboratory
Assignment – 3

This assignment is a programming assignment where you need to implement the given scenario in the C/C++ programming language.

Instructions:

- Each group needs to implement all three questions.
- Submit the assignment by the deadline of **11:55 pm on Monday, 16th Oct 2023 (hard deadline).**
- Write your source codes and do not copy from any source. Plagiarism detection tools will be used and any detection of unfair means will be penalized by awarding **NEGATIVE marks** (equal to the maximum marks for the assignment).

Q1) Implement a simplified version of the OSPF (Open Shortest Path First) routing protocol in C/C++. The goal of this assignment is to build a routing table for a network of routers and simulate the routing of packets between them.

Specifications:

1. Implement a Router class in C/C++ with the following attributes:

- router_id: A unique identifier for the router (an integer).
- neighbours: A list of neighboring routers, represented as pointers to other Router objects.
- routing_table: A data structure to store routing information.
- add_neighbor(Router* neighbour): A method to add a neighbouring router to the current router's list of neighbours.
- update_routing_table(): A method to calculate the routing table for the current router using the OSPF algorithm (For demonstration purposes, you can assume static routing here)
- print_routing_table(): A method to print the routing table for the current router.

2. Implement the OSPF algorithm within the update_routing_table() method to compute the shortest path to each router in the network using Dijkstra's algorithm or a similar approach.
3. Create a network of routers and connect them by adding neighbours to each router.
4. Simulate the routing of packets from one router to another by calling the update_routing_table() method and then using the routing table to determine the path.
5. Print the routing table for each router to demonstrate the routing information.

Q2.) Implement a simplified MAC layer simulation for a wireless network. Create a program that simulates the behavior of nodes in a network communicating using the CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) protocol.

Requirements:

a) Create a simulation environment with a configurable number of nodes (N) and shared communication channel.

b) Implement a backoff mechanism where nodes choose random backoff intervals before attempting to transmit data. The backoff interval should be adjustable.

c) Simulate the CSMA/CA protocol: Nodes should listen to the channel and only transmit when it's clear (no other nodes transmitting). If the channel is busy, they should back off and retry after the back-off period.

d) Implement a collision detection mechanism: If two or more nodes attempt to transmit simultaneously, a collision occurs, and they should back off for a random period before reattempting.

e) Record statistics: Keep track of successful transmissions, collisions, and the number of times nodes had to back off before successfully transmitting.

Q3) Write a C/C++ program to implement the Stop-and-Wait Flow control protocol.