Network Packet Routing Simulator

Reg. Numbers. : 2023528, 2023534, 2023572
Name: Taha Farooq, Umair Afridi, Neshmiya

Network Packet Routing Simulator

Project Proposal & Conceptual Design

1. Project Title & Problem Statement

Problem Statement:

Modern computer networks rely on efficient routing mechanisms to ensure data packets reach their intended destinations with minimal latency and packet loss. The challenge lies in dynamically selecting the optimal path for packet transmission while adapting to network conditions such as congestion, latency, and link failures.

---

2. Data Structures Selection

To effectively simulate a network environment, the project will utilize the following **key data structures:**

1. **Graph (Adjacency List Representation)**
    - Nodes represent routers, and edges represent network links.
    - Edge weights correspond to latency/bandwidth.
2. **Priority Queue (Min-Heap)**

    - Used in **Dijkstra's algorithm** to efficiently select the next node with the shortest path.
3. **Queue (FIFO Structure)**
    - Used to simulate packet buffering and transmission order.
4. **Linked List**
    - Used to track packet retransmissions in case of loss.

---

3. Algorithmic Approach

The simulator will implement the following algorithms:

1. Dijkstra's Algorithm (For Shortest Path Routing):

- Used to compute the shortest path between routers based on link latency.
- Utilizes a **priority queue (min-heap)** to optimize node selection.

2. Bellman-Ford Algorithm (Alternative for Dynamic Routing):

- Used for **dynamic link weight adjustments** in cases of congestion.
- Handles **negative weight edges** (although uncommon in networking).

3. Packet Loss Simulation & Retransmission:

- A random function (e.g., 10% probability of loss) determines whether a packet is lost.
- If lost, the packet is retransmitted through the same route.

---

4. Input & Output Design

Input:

- Initialize a graph representing the network.
- User specifies source and destination routers.

Processing:

- Build the network graph with weighted edges.
- Compute the shortest path using Dijkstra's algorithm.

Output:

- Display the shortest path, total latency, hops, and retransmissions.
- Show real-time packet transmission logs.

---

5. Course Concepts Application

This project integrates several fundamental **Data Structures and Algorithms (DSA)** concepts, including:

1. **Graph Theory:** Representing and traversing a network topology using adjacency lists.
2. **Shortest Path Algorithms:** Implementing **Dijkstra's and Bellman-Ford algorithms**.
3. **Queue & Priority Queue:** Managing packet flow and priority-based routing.
4. **Dynamic Programming:** Optimizing packet routing decisions under changing network conditions.
5. **Randomization & Probabilistic Modeling:** Simulating **packet loss and network performance variations**.

By implementing these techniques, the project not only reinforces theoretical DSA concepts but also provides a **real-world networking application** that mirrors how routers handle packet transmission dynamically.

.