**DSA**

**Open Ended Lab**

**Proposal**



**Haroon | 2023231**

**Umar | 2023536**

**Rafay | 2023590**

### 

### 

### **Proposal: Network Packet Analyzer**

#### **Objective**

To develop a **Network Packet Analyzer** that simulates packet management in a network environment. The tool will track incoming and outgoing packets, prioritize critical packets, and efficiently search and analyze packets based on predefined criteria. This project addresses a real-world problem of managing network traffic in routers or servers.

### **Problem Statement**

In networking, managing data packets efficiently is critical for ensuring smooth communication between devices. Challenges like prioritizing critical packets, searching for specific packets in a large dataset, and analyzing packet data for patterns often arise. This project aims to design a solution that applies data structures and algorithms to handle these challenges effectively.

### **Core Features**

1. **Dynamic Packet Storage:**
   * Use linked lists to dynamically store incoming packets with attributes such as source, destination, priority, and size.
2. **Packet Prioritization:**
   * Use sorting algorithms to prioritize packets based on their importance (e.g., emergency packets before regular packets).
3. **Search for Packets:**
   * Efficiently search for packets by attributes like source or destination using a binary search tree.
4. **Packet History and Recovery:**
   * Implement undo/redo functionality for packet modifications (e.g., changing priority or deleting packets) using stacks.
5. **Traffic Statistics:**
   * Use recursion to compute statistics such as the total number of high-priority packets or the average packet size.

### 

### 

### 

### **Core Concepts Utilized**

1. **Linked Lists:**
   * Manage dynamic and continuous storage of incoming and outgoing packets.
2. **Stacks:**
   * Enable undo/redo operations for changes made to the packet list.
3. **Binary Search Tree (BST):**
   * Allow efficient searching and retrieval of packets based on attributes like priority or source.
4. **Sorting Algorithms:**
   * Prioritize packets using algorithms like quicksort or mergesort.
5. **Recursion:**
   * Analyze traffic patterns, compute averages, and extract other insights from the packet data.

### **Expected Outcomes**

* A functional **Network Packet Analyzer** capable of handling real-world challenges like dynamic packet management, prioritization, and efficient searching.
* Demonstration of practical applications of at least five core data structure and algorithm concepts.
* A clear understanding of how to address network traffic problems in computer engineering using fundamental principles.