Assignment Guidelines:

Teamwork:

The assignment must be completed in pairs of two students. Only one team member (Member 1) must submit the assignment on behalf of the group.

PCAP File Selection:

Choose the appropriate ***X***.pcap file where: ***X***= (Team ID) % 9. TeamID is the Number from the Google Sheets. Find the pcap files here: PCAPs

Submission File Naming:

Report should be named as:***TeamID-RollNumber1-RollNumber2.pdf*** For example, For TeamID 1, and roll numbers 23210122 and 23210023, the file should be named: **1-*23210122_23210023.pdf*** and the submission must be made by student with roll number **23210122** only.

GitHub Repository:

Include a link to your GitHub repository containing all the programs and scripts used for the assignment. A common repository may be provided to you at later stages to upload all your project files.

Tools for Part One and Part Two: Packet Sniffer Development

Objective:

- Implement a raw packet sniffer in your preferred programming language. Extend the sniffer to analyze and compute the metrics requested in Part1 and Part2.

Data Source:

- Utilize a previously captured packet capture (pcap) file.
- Do not connect to the internet during the packet sniffing and analysis process.
- Analyze traffic solely based on the data within the pcap file.

Packet Replay:

- Replay the captured network traffic using a tool like **tcpreplay**.
  Key Feature: *tcpreplay* allows for manual speed configuration. This is crucial to prevent packet loss at the sniffer due to excessive replay speed.
- Alternatively, explore and utilize any other suitable packet replay tool.

Deliverables:

- i) Report, ii) Code, iii) test script and a iv) ReadMe file indicating the instructions to execute and reproduce your result. A functional packet sniffer was modified to keep account of metrics/contents as required in the questions.

Part 1: Metrics and Plots (40 pts)

From the chosen *X*.pcap file, extract and generate the following metrics for the data as captured by your program when you perform the pcap replay using tools like tcpreplay:

1. Find the total amount of data transferred (in bytes), the total number of packets transferred, and the minimum, maximum, and average packet sizes. Also, show the distribution of packet sizes (e.g., by plotting a histogram of packet sizes).
2. Find unique source-destination pairs (source IP:port and destination IP:port) in the captured data.
3. Display a dictionary where the key is the IP address and the value is the total flows for that IP address as the source. Similarly display a dictionary where the key is the IP address and the value is the total flows for that IP address as the destination. Find out which source-destination (source IP:port and destination IP:port) have transferred the most data.
4. List the top speed in terms of `pps` and `mbps` that your program is able to capture the content without any loss of data when i) running both tcpreplay and your program on the same machine (VM), and ii) when running on different machines: Two student group should run the program on two different machines eg. tcpreplay on physical-machine of student1 and sniffer program physical-machine of student2. Single students should run between two VMs.

Part 2: Catch Me If You Can (40 points)

For the designated *X*.pcap file, extend the program to sniff and answer the specific questions:
Find **pcap** Specific Questions here: PCAP specific Questions

Part 3: Capture the packets (20 points)

1. Run the Wireshark tool and capture the trace of the network packets on your host device. We expect you would be connected to the Internet and perform regular network activities.
   a. List at-least 5 different application layer protocols that we have not discussed so far in the classroom and describe in 1-2 sentences the operation/usage of protocol and its layer of operation and indicate the associated RFC number if any.

2. Analyze the following details by visiting the following websites in your favourite browser.
   i) canarabank.in
   ii) github.com
   iii) netflix.com
   a. Identify `request line` with the version of the application layer protocol and the IP address. Also, identify whether the connection(s) is/are persistent or not.
   b. For any one of the websites, list any three header field names and corresponding values in the request and response message. Any three HTTP error codes obtained while loading one of the pages with a brief description.
   c. Capture the Performance metrics that your browser records when a page is loaded and also report the list the cookies used and the associated flags in the request and response headers. Please report the browser name and screenshot of the performance metrics reported for any one of the page loads.