ASSIGNMENT 1

EE673A DIGITAL COMMUNICATION NETWORKS

2022-2023 (Odd)

19th August 2022 Deadline: 2th September 2022

Instructions:

1. Submissions are to be made through the HelloIITK portal
2. You are required to submit a single zip file containing well commented and easy to understand codes and a single PDF (containing answers to theory questions, if required). There should also be a readme file with clear instructions on how to run the codes.
3. Kindly name your submission file as <RollNo>_<Name>,
   eg: 18204269_LavishArora
4. Marks will be deducted for all submissions that do not follow the above guidelines.

Question 1: Network Queuing Simulation

In this exercise we will simulate a network queue and visualize the queue size under various network traffic conditions.

Consider a single user communicating through a network router with a fxed transmission capacity(R = 5 bits/second). At each time instant the user communicates with probability p. If the user is communicating at time t, it generates a packet of size either $a = 2, b = 4, c = 6$ or $d = 8$ bits with probability p_a, p_b, p_c and p_d respectively with $p_a + p_b + p_c + p_d = 1$

Part A: Simulate the above problem and plot [*15 marks*]

1. average queue length (in bits) vs p
2. average packet delay vs p

(varying p from 0 to 1) for the following cases:

(i) $p_a = p_b = p_c = p_d = 0.25$

(ii) $p_a = p_d = 0, p_b = p_c = 0.5$

(iii) $p_a = p_d = 0.5, p_b = p_c = 0$

(iv) $p_a = p_b = p_c = 0, p_d = 1$

Part B: Answer the following questions: [*5 marks*]

(a) For each of the above case, what is the expected number of incoming bits at time t as a function of p, i.e, if X is the number of arrival bits in a given time slot, what is $E[X]$ for each of the above cases as a function of p? Also, find the variance of X, for each of the above four cases.

(b) In which of above case the incoming communication traffic (i.e., expected number of arriving bits in a time slot) exceeds the network capacity for $p = 1$?

Question 2: Socket Programming with UDP and TCP

Part A: For this part refer to the UDP and TCP socket programming examples given in Section 2.7 of Computer Networking A Top-Down Approach 6th Edition, Kurose and Ross. Using the above examples create UDP and TCP client and server applications where:

- The client reads a line of characters from its keyboard and sends the data to the server
- The server receives the data and converts the characters into uppercase
- Server sends the modified data to the client

- The client receives the modified data and displays the line on its screen

[*10 marks*]

Part B: Create a UDP chat application for communicating with your phone using the UDP monitor

- Install the UDP monitor application on your android phone. Connect your phone and PC such that they are on the same network. This can be done either by
  - Connecting them to the same wifi network
  - Connecting your PC to phone's mobile hot-spot, or
  - Connecting phone and PC via USB tethering
- Once you ensure phone and PC are on the same network, open the UDP monitor application on the phone.
- The IPv4 address of your phone will be displayed on the top left corner. Also select the local port on the phone. Use this address while designing your UDP chat application.
- For two way communication give the remote address and port of your PC in the UDP monitor mobile application.

Design a two way chat application with the UDP monitor [*15 marks*]

HINT: For parallel two way communication, you may need to use the 'threading' feature in python that implements parallel processes.
UDP Monitor application link:(`https://tinyurl.com/udpMonitor`)

Question 3: UDP Pinger

For this question, write a client ping program in Python. Your client will send a simple ping message to a server, receive a corresponding pong message back from the server, and determine the delay between when the client sent the ping message and received the pong message. This delay is called the

Round Trip Time (RTT). Here we will create a nonstandard (but simple!) UDP-based ping program.

Your ping program is to send 10 ping messages to the target server over UDP. For each message, your client is to determine and print the RTT when the corresponding pong message is returned. Because UDP is an unreliable protocol, a packet sent by the client or server may be lost. For this reason, the client cannot wait indefinitely for a reply to a ping message. You should have the client wait up to one second for a reply from the server; if no reply is received, the client should assume that the packet was lost and print a message accordingly. The client application should also report maximum, minumum and average RRT and also the packet loss rate. Find the server application code here. [*10 marks*]

UDP Ping server: `https://tinyurl.com/udpPingServer`

Question 4: Mail Client

The goal of this question is to create a simple mail client that sends email to any recipient. Your client will need to establish a TCP connection with a mail server (e.g., iitk mail server: *mmtp.iitk.ac.in*), dialogue with the mail server using the SMTP protocol, send an email message to a recipient (e.g., your friend) via the mail server, and finally close the TCP connection with the mail server.

Your job is to code and test your client by sending email to different user accounts. [*15 marks*]

END OF ASSIGNMENT