NETWORKS LAB

JEREMIAH THOMAS | 106119055

ASSIGNMENT 2

Do note:-

- Kindly find all the codes in the zip folder as requested
- Do cd into the right directory while running the programs.
- It is preferred to run the programs on a Linux environment.
- Also, **compiled.py** is a convenient one stop solution that runs all programs at once.
- So it is sufficient to only run: python3 compiled.py

Question 1: -

I. Begin by examining the dynamics of a simple TCP connection between two endpoints on a wired network.First, download this simple ns script in Assignment Folder and examine it

(a) Draw a very simple diagram of the network that the script simulates, including the one-way latency and bandwidth of the links.

(b) What kind of queueing discipline does the simulation's router use?

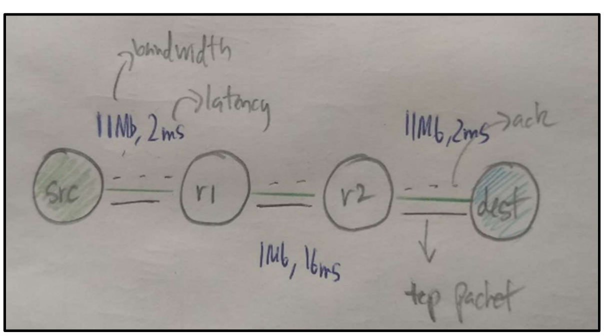
(c) Run the script and plot the evolution of the sender's TCP window over time. Identify where TCPslow-start ends and where congestion avoidance begins.

(d) What is the average throughput of the transfer?

Answer

A)

Diagram including the one-way latency and bandwidth of the links: -



B)

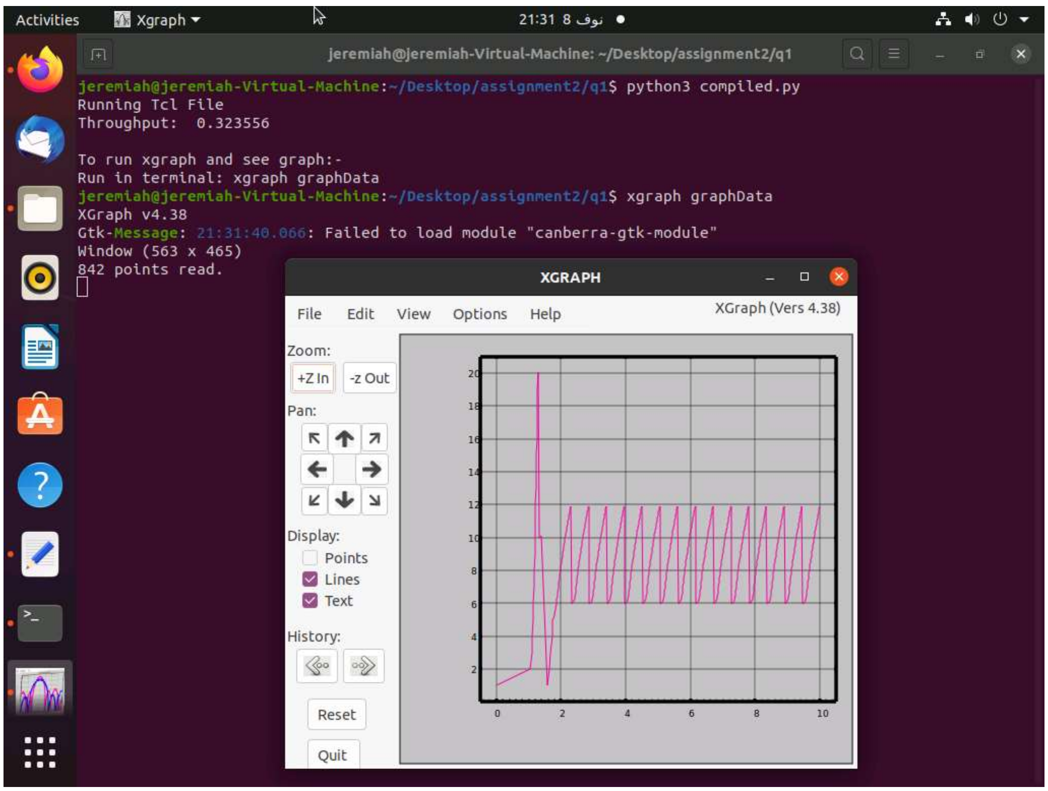
The Queuing discipline followed here is **FIFO** and is taken by default. In this discipline, packets that arrive first are given higher priority that the packets arrive later.

Also, A DropTail attribute is added which causes the queue to drop packets beyond the droptail limit.

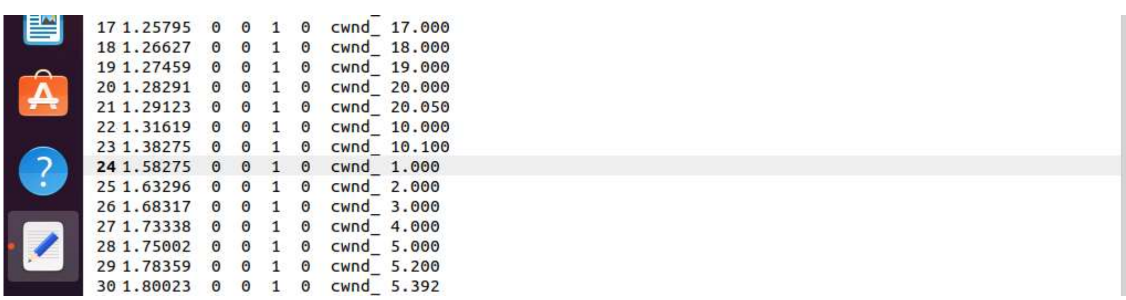
D)

Avg Throughput = 0.323556 as shown in the screenshot

Terminal:-



C)



In a nutshell, the point where slow start ends and congestion avoidance begins is**: 1.78359s.**

Reasoning:-

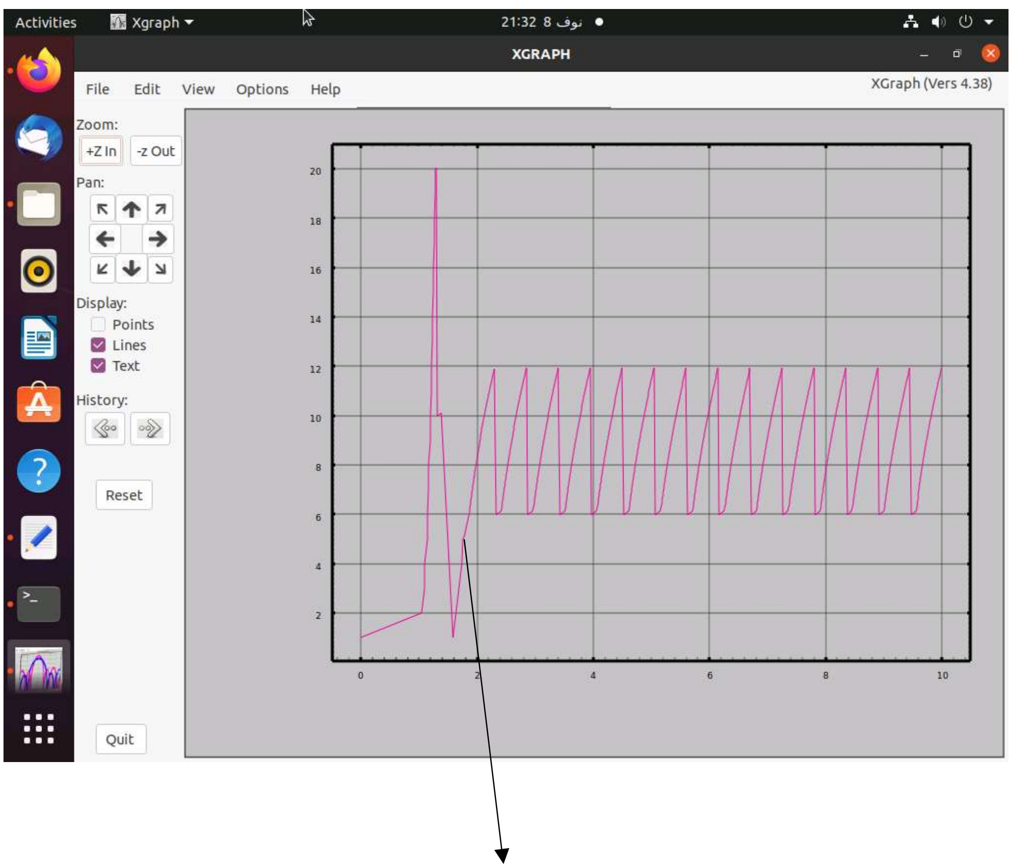
One can come to the conclusion that **1.58275** was the **beginning of the slow start** as if you observe in the above snippet, you come to realize that at line 24, cwnd dropped to 1, which can only happen if cwnd size is 2 or if a state of congestion detection with timeout was hit.

In this case, the cwnd size was 10.1 and thus it is clear that **timeout has occurred at line 23** (1.38275s), causing cwnd to start from 1. Thus, the network goes into slow start state from 1.58275s.

Here at this point, the **sstresh value** changes to = (10.1)/2 = **5.05.**

At **1.78359s**, **cwnd breaches the sstresh mark** and thus enters into the **congestion avoidance phase**. (It can be clearly ascertained as congestion avoidance because whenever a network hits sstresh from a slow start state, it moves to congestion avoidance, increasing one unit for every RTT.)

Graph:-



1.78359s: Slow Start End, Congestion Avoidance Start

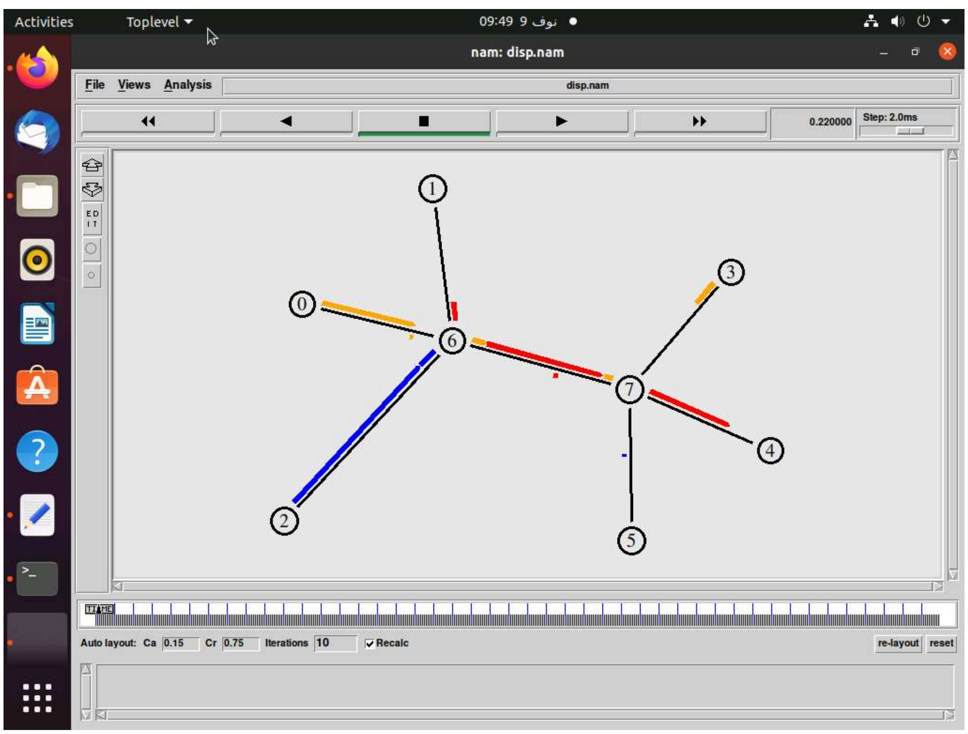
Question 2)

II. Consider a topology with "n" being 3, and thus three TCP connections.

1. For each of the TCP connections, calculate the *average* and *standard deviation* of the packet inter-arrival times.
2. Plot a *single* graph, but with three plots within it -- each plot corresponds to the inter-arrival time versus sequence number plot for a TCP connection (just as in the first question). Plot this just for the first 100 sequence numbers, as earlier. Submit this graph.
3. Explain the above average and standard deviation, as well as the graph above. Explain with respect to the "n=1" and "n=2" cases. (6-7 sentences max).

Answer:-

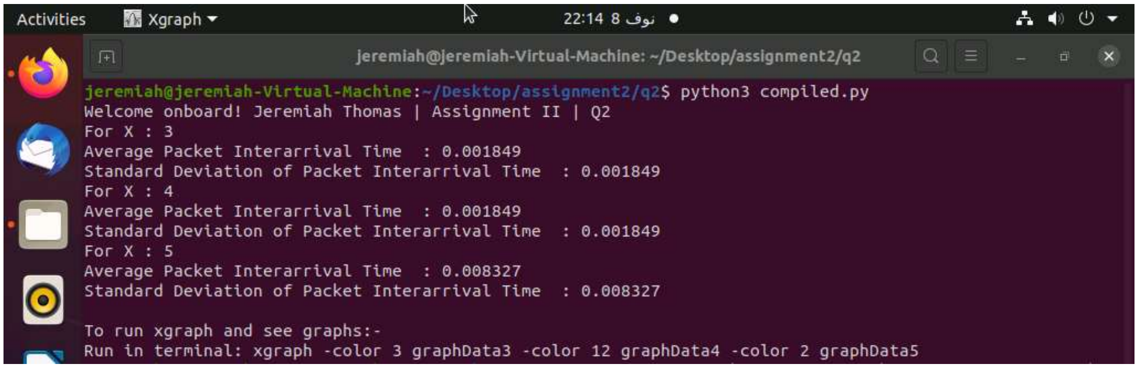
Simulation snapshot:-



---

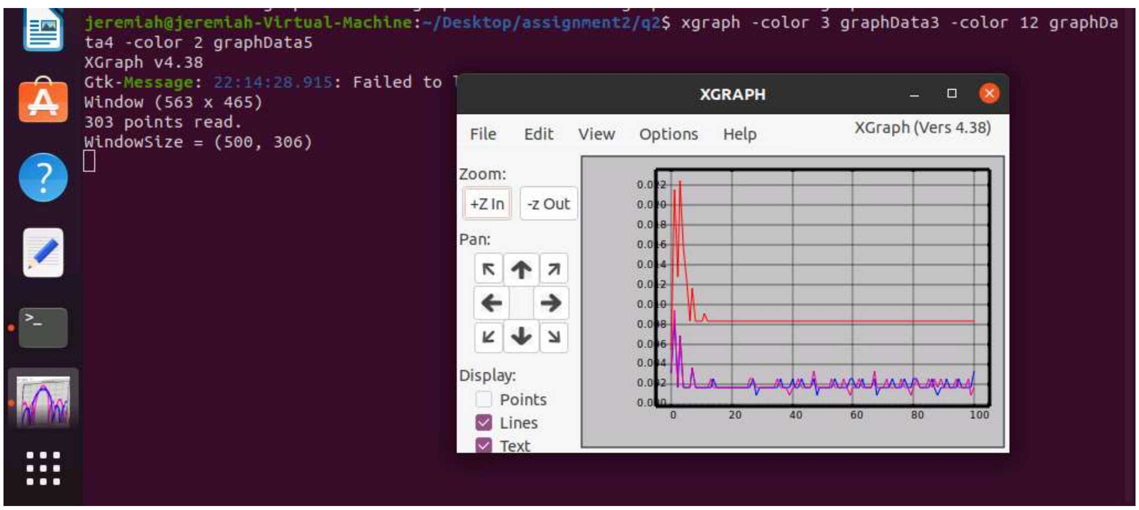
1)

- **Do note:** X below refers to the destination node of each tcp connection. So, for instance, X:3, is the destination B0 from A0 for tcp connection1 and so on



---

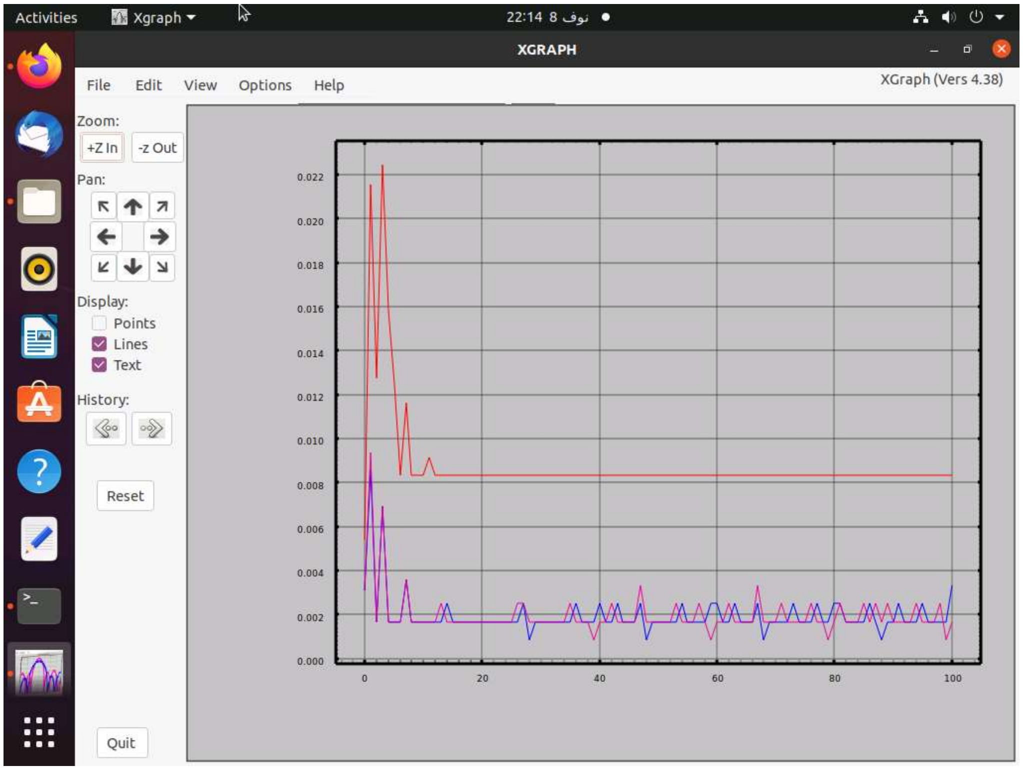
2) **Graph:-**



Color code:-

- Tcp connection 1: (0-3) -> Blue
- Tcp connection 2: (1-4) -> Purple
- Tcp connection 3: (2-5) -> Red

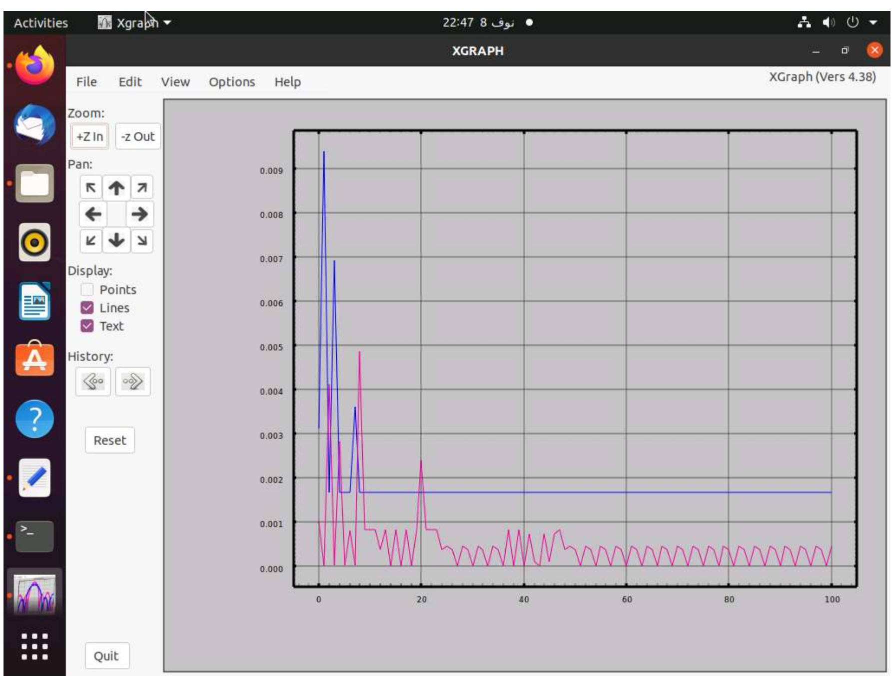
(DO SCROLL DOWN)



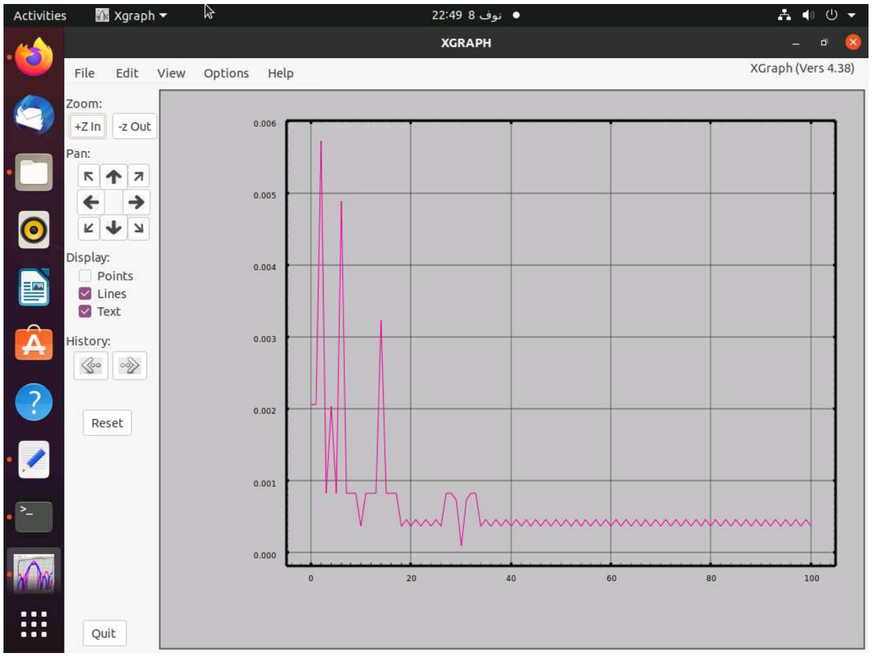
---

3)

When N=2:-



When N=1:-



Inference :-

- The general rule is that for large STDV, interarrival times are inconsistent. Here, in this scenario you can see tcp connection 3 has the greatest STDV and is the most inconsistent graph.
- Also, while tcp connections 1 and 2 nearly overlap and have the same interarrival times, tcp connection 3 is way above the rest.
- An interesting phenomenon to note in n=2 and n=3 is that there is one connection which is above the rest and becomes constant somewhere after 15s. This could possibly be one tcp connection being given preference over the other.
  Thus, the preferred connections have lesser interarrival times while the other will have much longer time and possibly approximates when bandwidth capacity is reached.
- As n increases from 1->3 the peak interarrival time (from 0.006 to 0.009 to 0.022) rises which can be intuitively attributed to a rise in congestion as packets pass through the X-Y interlinking nodes.
- The uniform initial rise across the 3 slopes could be a result of packet losses happening in the initial start due to all connections beginning to send data at the same time, with the same priority.

THANK YOU!