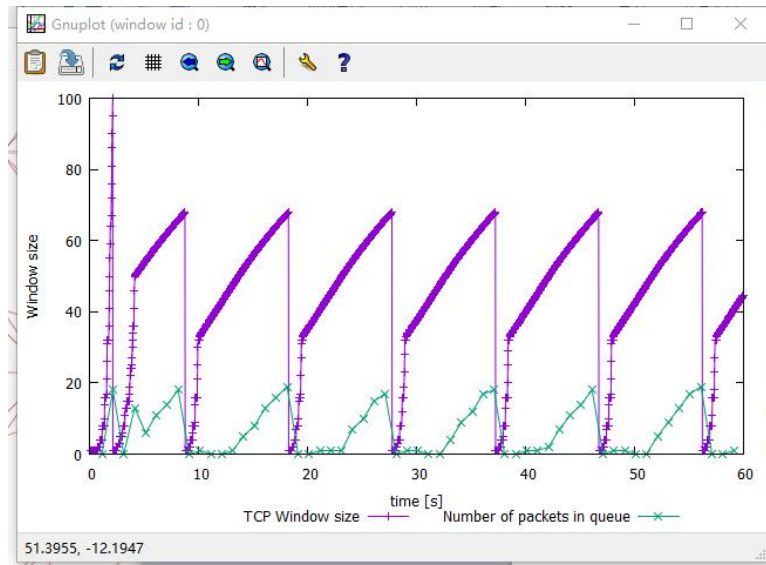# Exercise 1

Question1

What is the maximum size of the congestion window that the TCP flow reaches in this case? What does the TCP flow do when the congestion window reaches this value? Why? What happens next?


The maximum size of the congestion window is 100.

When congestion window reaches 100, TCP flow stop increase the window size and reset it from 1. At the same time, it set a new ssthresh to control the transmission and the value of ssthresh is the half of the maximum congestion window. So when next time congestion window reach ssthresh it will decrease the growing speed. And next the congestion window start from 1 and increase fast until it reach 50 which is the value of ssthresh and it become grow slowly until the congestion happen, the ssthresh will be reset to the half of the congestion window size.

Question2

From the simulation script we used, we know that the payload of the packet is 500 Bytes. Keep in mind that the size of the IP and TCP headers is 20 Bytes, each. Neglect any other headers. What is the average throughput of TCP in this case? (both in number of packets per second and bps)
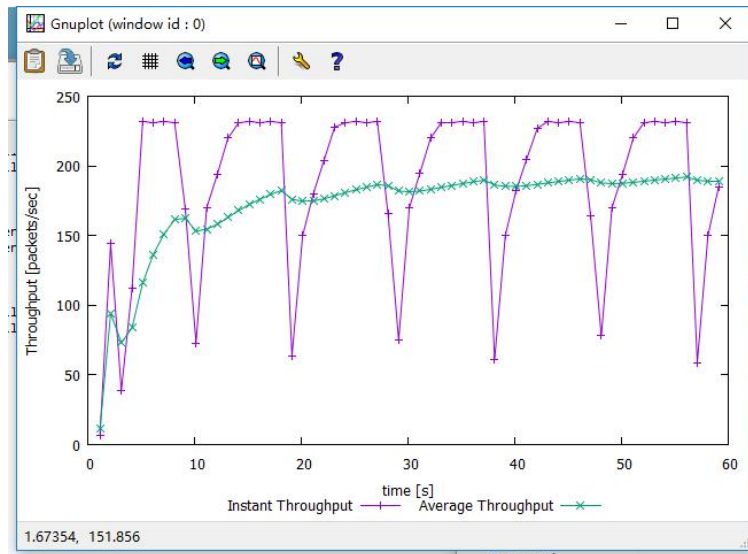
It show that the average is nearly keep in 190 packets per second.

Question3

Rerun the above script, each time with different values for the max congestion window size but the same RTT (i.e. 100ms). How does TCP respond to the variation of this parameter? Find the value of the maximum congestion window at which TCP stops oscillating (i.e., does not move up and down again) to reach a stable behaviour. What is the average throughput (in packets and bps) at this point? How does the actual average throughput compare to the link capacity (1Mbps)?
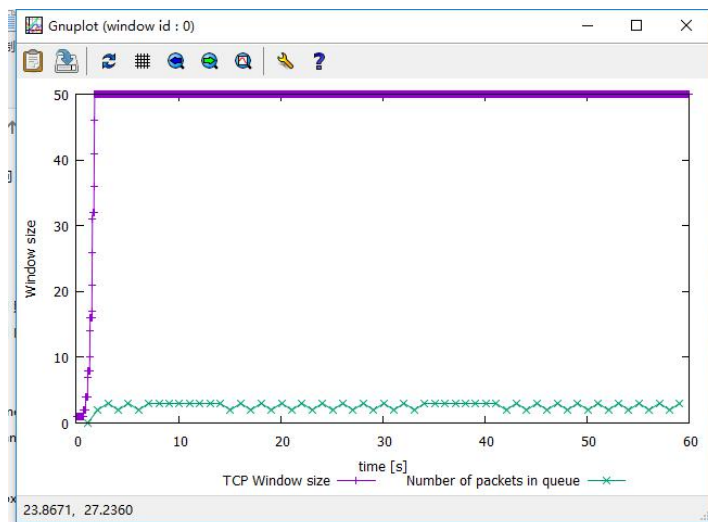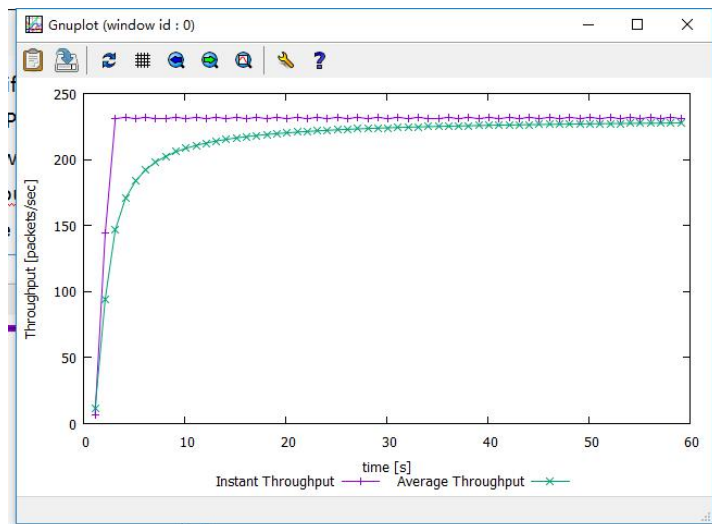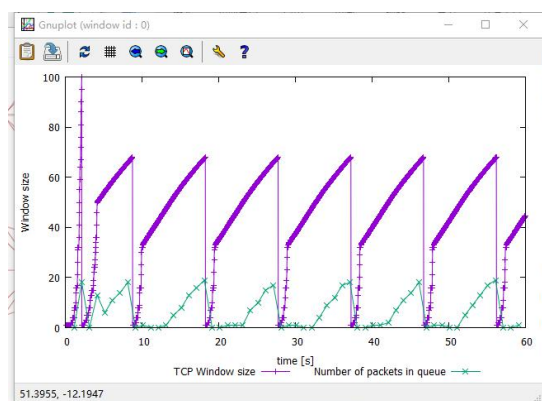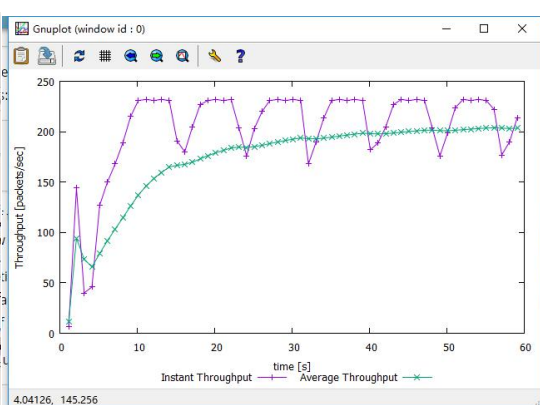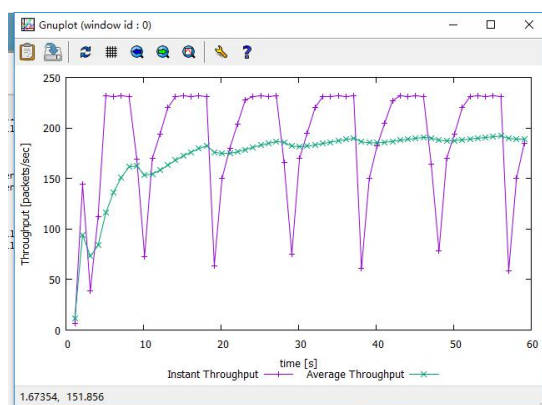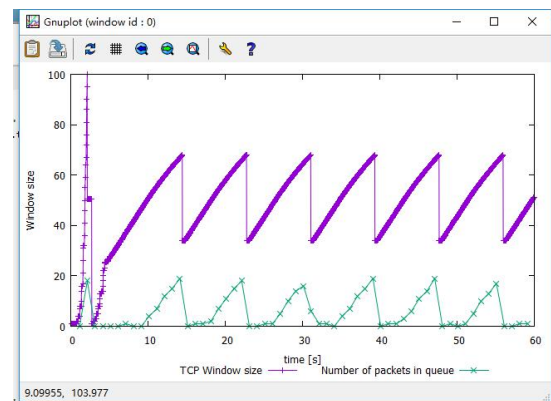
‑As two graph show above, when the max congestion window size be set at 50. At the beginning the congestion window size increase from 1 but when the value increase at 50 it stop increase and keep in 50, the TCP stops oscillating. The average throughput is nearly keep at 235 packets per second.

Question4

TCP Tahoe   TCP Reno

As shown above, when TCP Reno get the max congestion window size, the congestion window

size will be restart from 1, and ssthresh will be set to be 1/4 of the max congestion window size. When next congestion happen, the TCP do not reset the congestion window size, just set the ssthresh to the half of congestion window size and the congestion window size will restart from it.

Exercise 2
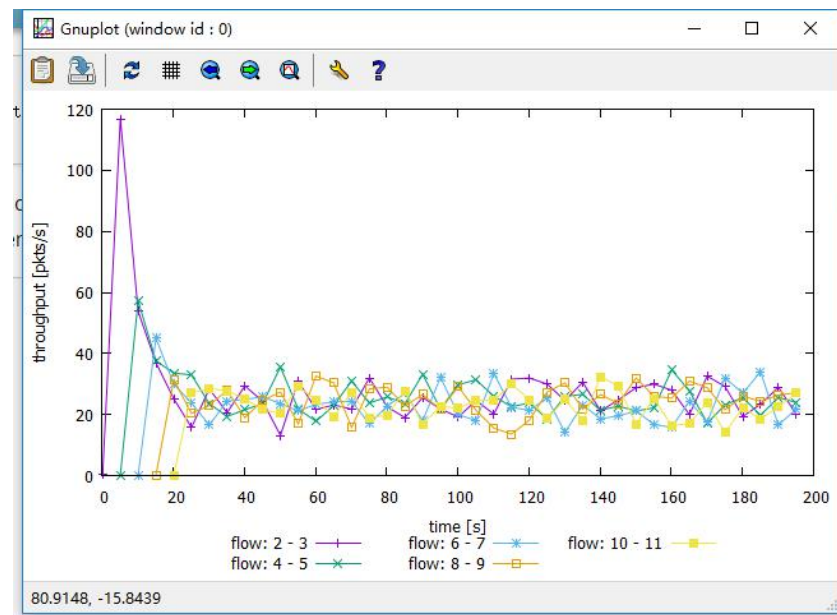
Question1

As the graph shows, each flow not get an equal share of the capacity. Because each flow always try to get largest throughput until the congestion happen and it will jump down after that it will still keep try get the largest throughput. Although the flow seems get similar capacity, but it is not stable. The capacity be restrained just because the congestion window size.
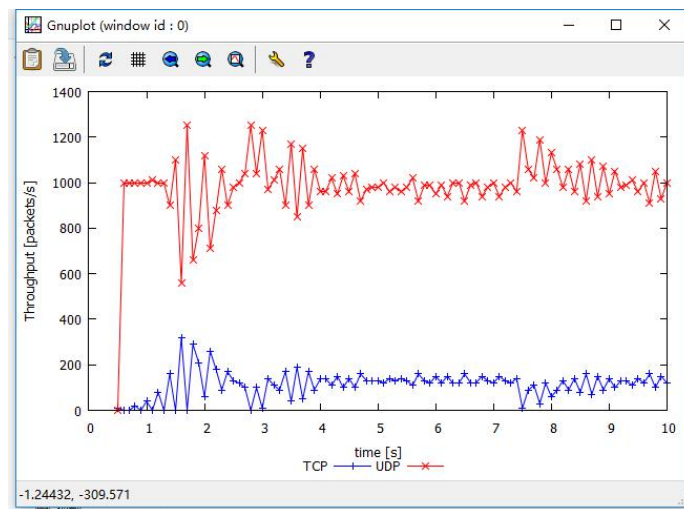
Question2

When the new flow is created, the throughput of the pre-existing TCP flows will decrease immediately and share the throughput to the new flow. The TCP is try to let each flow take the same percentage of the whole throughput. For this is a fair behaviour to make sure the new flow can get transmission ability, otherwise the transmission will be always take by pre-existing TCP flows until it finish transmission.

Exercise3

Question1
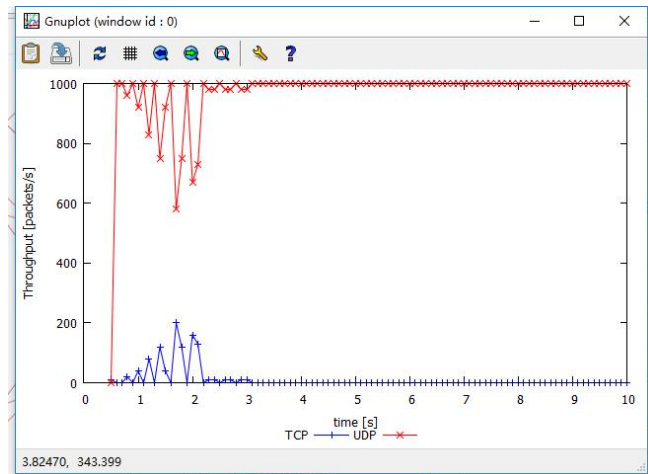
When the link capacity is 4Mb, the UDP transmission is be restrain by the capacity so when the capacity increase it will have a higher throughput. The throughput of TCP will be increase too.

Question2

Since TCP is a stable and reliable transmission, so it have more limit of it. UDP just to send the data not need to make a connection between the source and destination. The main constrain of TCP is the congestion window size and the link capacity. But UDP is rely on the capacity of the source computer which means the data production rate, it just need to put data to internet.

Question3

TCP：

Advantages: reliable, stable.

Disadvantage: too many constrain, transmission speed.

UDP:

Advantages:fast.

Disadvantage: easily been influence.

If everybody started using UDP instead of TCP it will hard to download an entire file from the internet. The data would not be exchange exactly.