Files submitted:

- udp_select.c
- udp_select_timeout.c
- client.c

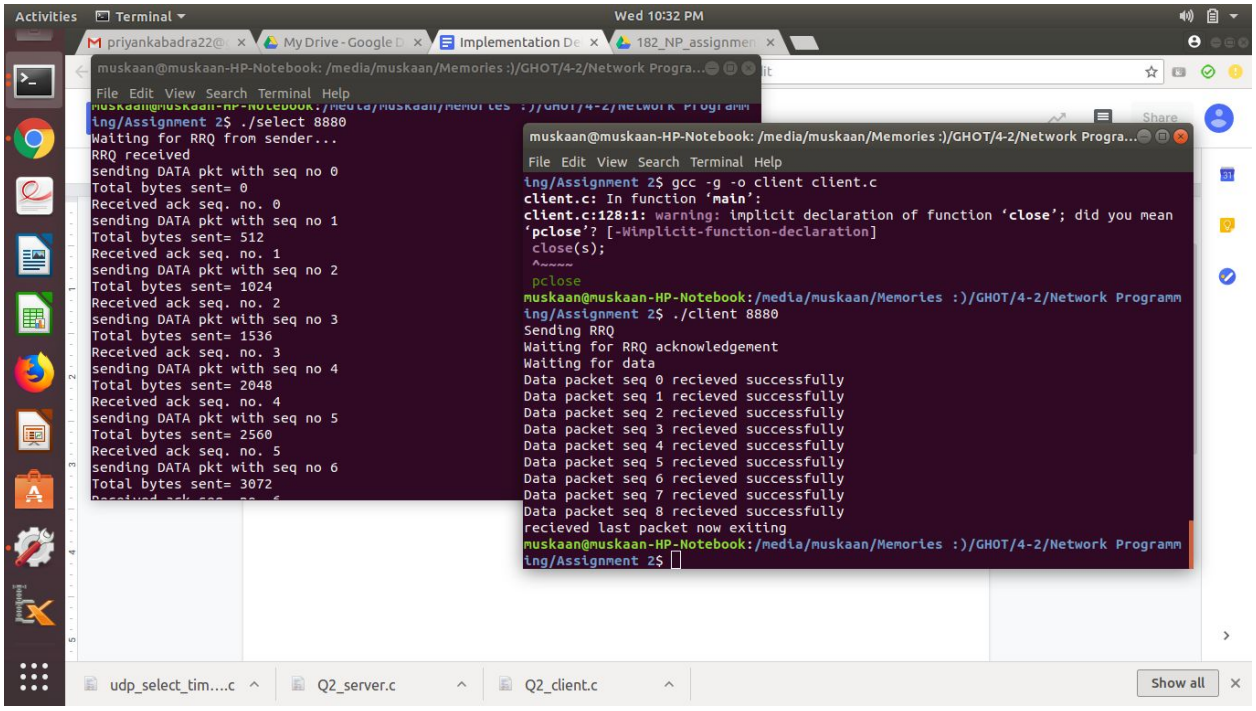
The client.c and the server file udp_select.c initially require the same port for establishing the connection. This port is used for sending the RRQ by the client. The port number is provided in the command line arguments. Example: ./server 8880 and ./client 8880.

After receiving the RRQ from the client, the server creates a new port “PORT2” (fixed as 8882 in this code), and send an acknowledgement to the client using this port. The client sends any other message using this new port only.

The server then sends 512 bytes of data to the client. The client recieves the data and sends an ACK packet to the server. The server waits for the ACK packet for the desired sequence number and after recieving an appropriate ACK packet, it sends next DATA packet. The process is continued till a DATA packet has data of length less than 512 bytes. A zero DATA packet is sent if the file is in multiple of 512.

The execution of code clearly displays the total number of packets and total number of bytes sent.

A screenshot after the execution of the code has been displayed below for reference.



The udp_select_timeout.c displays the dynamically calculated timeout values as asked in the question. Two functions calculateRTT and calculateTimeout implement the algorithms as asked.