SS:
First I ran the file_receiver using the command

‘python file_receiver.py ss received_file.txt’,

where ‘received_file.txt is the name of the new file that will be created.

Next I ran the file_sender using the command

‘python file_sender.py ss Latin-Lipsum.txt’.

Here I am trying to copy the contents of ‘Latin-Lipsum.txt’ into the ‘received_file.txt’

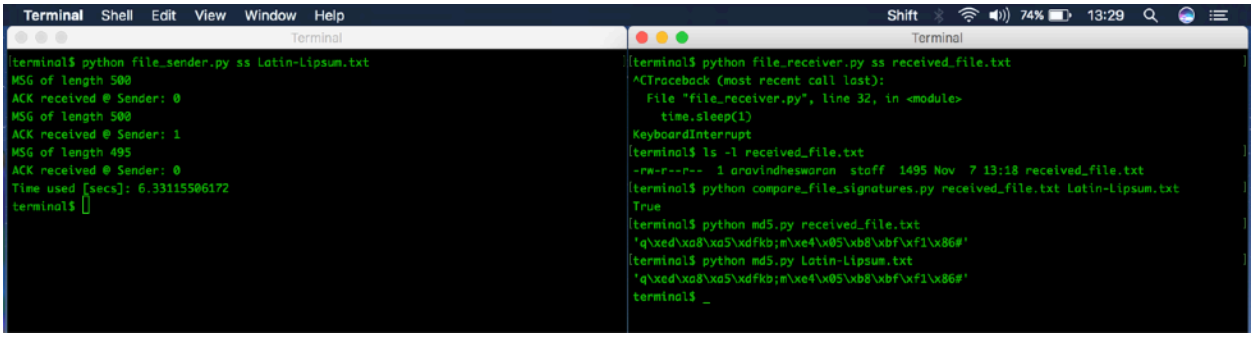
Then I shut down the file_receiver using a keyboard interrupt and did the ‘ls’ command. The file is created.

I wrote a custom program ‘compare_file_signatures.py’ to compare the signatures of the two files using the md5 hash. Next I ran the ‘compare_file_signatures.py’ using the command

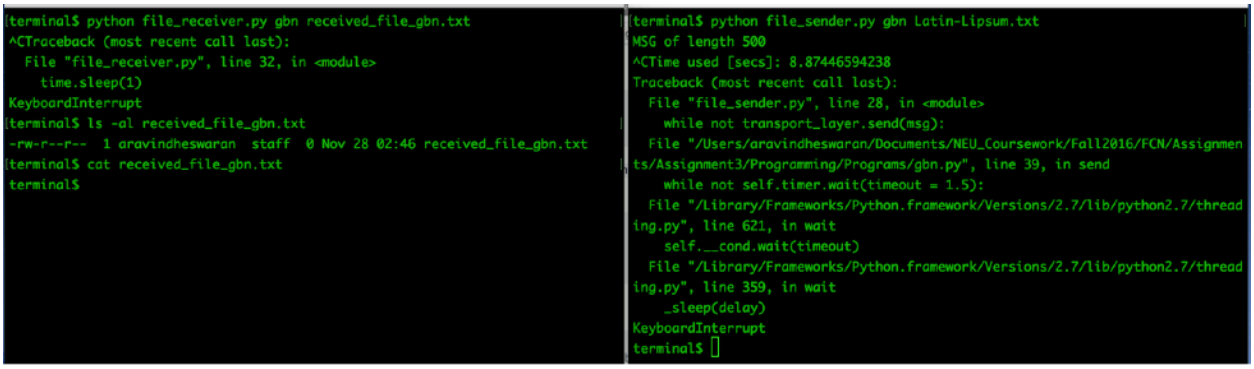
‘python compare_file_signatures.py Latin-Lispum.txt received_file.txt’

The output returned so, I concluded that the file was exactly replicated at the receiver.

Though, there was no negative ACK in the screenshot provided, I was able to produce correct results when there was negative ACK returned or when there was a timeout.



GBN:
Since, I was not able to completely able to resolve the question myself, I had a discussion with my classmate on how to approach this problem.



I first started the file_receiver using the command

‘python file_receiver.py gbn received_file_gbn.txt’,

where ‘received_file_gbn.txt’ will be the new file created.

Similarly, I started the file_sender using the command

‘python file_sender.py gbn Latin-Lipsum.txt’

A part of the text (the first 500 bytes of data) was transferred from the sender to the receiver. I tried to implement as it was given in the question but for some reason, it would not fully execute. The program just gets struck after sending the initial 500 bytes of data. I had to give the keyboard interrupt command (ctrl - c) to come out of execution.