1. Team details: Clearly state the names and netids of your team members (there are 2 of you).

Olaolu Aina (ooa55) Adeola Adebanjo (aoa90)

2. Collaboration: Who did you collaborate with on this project? What resources and references did you consult? Please also specify on what aspect of the project you collaborated or Consulted.

Lecture Slides, Recitation notes, Piazza to understand the process of both stop and wait and pipelining.

3. Is there any portion of your code that does not work as required in the description above? Please explain.

There is currently no known portion of the code that is not working.

4. Did you encounter any difficulties? If so, explain.

One difficulty we encountered was updating the right edge of the window for pipelining. The sender would keep timeout and end sending one packet at a time making the transmission time of data very long.

5. Describe two technical observations or facts you learned while working on this project. Please answer in specific and precise terms. Your observations could relate to topics including reliable data delivery in general, your implementation of it in this project, reliable delivery in TCP, using UDP sockets, or other topics that are relevant to your software and implementation in this project. Please ensure your responses are clear, specific, and technical.

One observation we made was the fact that the udp sockets do not initially have reliability. The data will be sent from the sender and the sender will not ensure that the data was properly received by the sender.

Another observation was when you start the sender before the receiver, the sender keeps timing out and retransmitting the dropped packets until the receiver connects before finally transmitting the rest of the packets. This is because the sender waits and buffers data so as to not send out of order data.