• Where did the bulk of logic occur? (3 pts)

The bulk of the logic occurs inside the for loop where each loop is processed. There are two processes. The first one is for acknowledgements in segments. It updates the last acknowledged sequence number and removes all segments from the send buffer that have been acknowledged, ensuring that only pending segments remain.The second one is the data segments. Here, it checks whether the segment has the correct checksum and is in order by comparing its sequence number with the expected value; if it is, the segment's payload is added to the received data and the expected sequence number is updated. If everything is correct, then the data is properly received and added to the dataReceived string (the final output).

• How were the timeouts resolved? What happened if timeouts have been resolved? (3+2 pts)

For each iteration, I checked whether the current iteration and the iteration when the segment was sent (seg.getStartIteration) was equal or greater than the timeout threshold I set (2). When a timeout happened, they were resolved by making sure I printed them out to the console to keep track of it. Then, I incremented my timeout counter for logging purposes, I made a new segment from the original data chunk and its corresponding sequence number (helps with checksum errors), I update the sentSegments with my new, original data segment, reset the segment's timer by setting its start iteration to the current iteration and finally retransmitted the segment over the channel.

Once the timeouts have been resolved, the receiver sends an acknowledgement that includes the next expected sequence number. When the sender processes this acknowledgement in processReceiveAndSendRespond, it removes the acknowledged segment from sentSegments and we don't need any more retransmissions.

• How was the packet dropping handled? (2 pts)

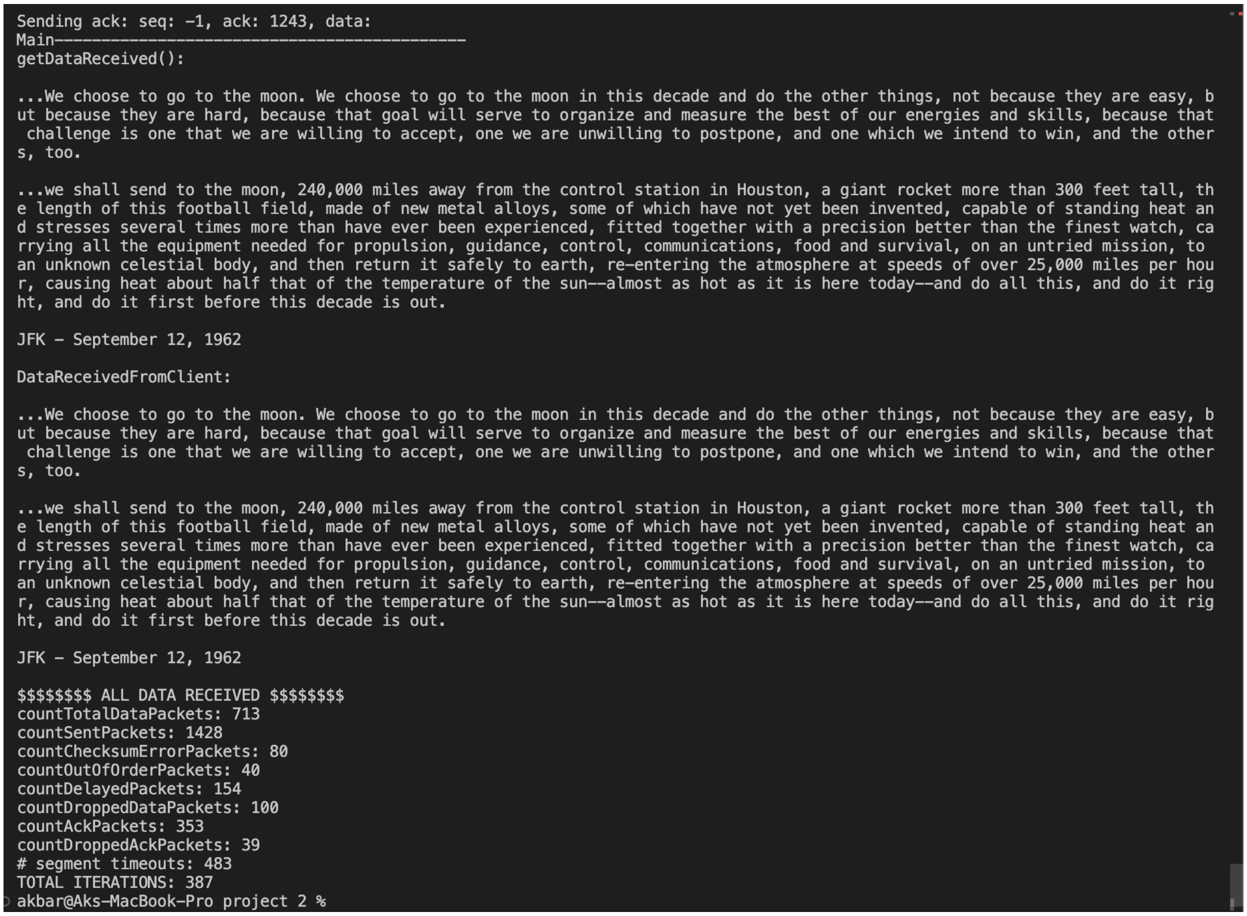
I handled packet dropping by utilizing acknowledgments and timeouts. When a packet is dropped, the sender would not receive an expected acknowledgment. A timeout will be detected because of the lack of an acknowledgement which makes it retransmit the segment.

Also, if a packet is dropped, then the cumulative acknowledgment number does not advance past the missing segment, prompting it to retransmit.

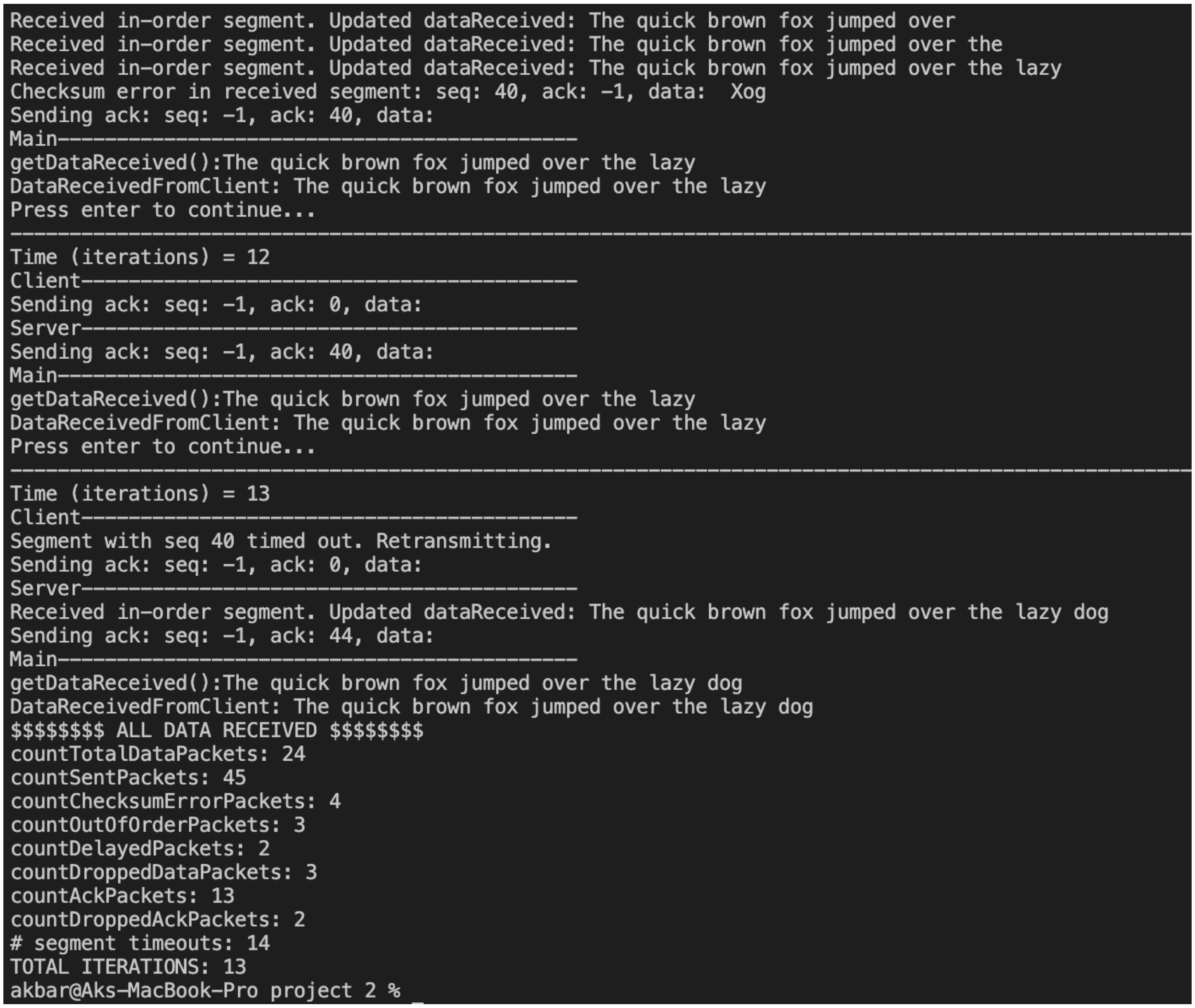
• How was the retransmission policy implemented (3+2 pts)?

I implemented the retransmission policy by using timeouts. For each iteration, it checks whether a timeout occurred. This timeout could have occurred because of dropped packets, incorrect checksums, etc. If a segment has timed out, the code prints a message and increments a timeout counter. I then retrieve the corresponding uncorrupted data chunk from self.dataToSend which I use to create a new segment with this data. This new segment is retransmitted through the send channel. This method ensures that any lost or corrupted segments are detected and retransmitted until they are successfully acknowledged.

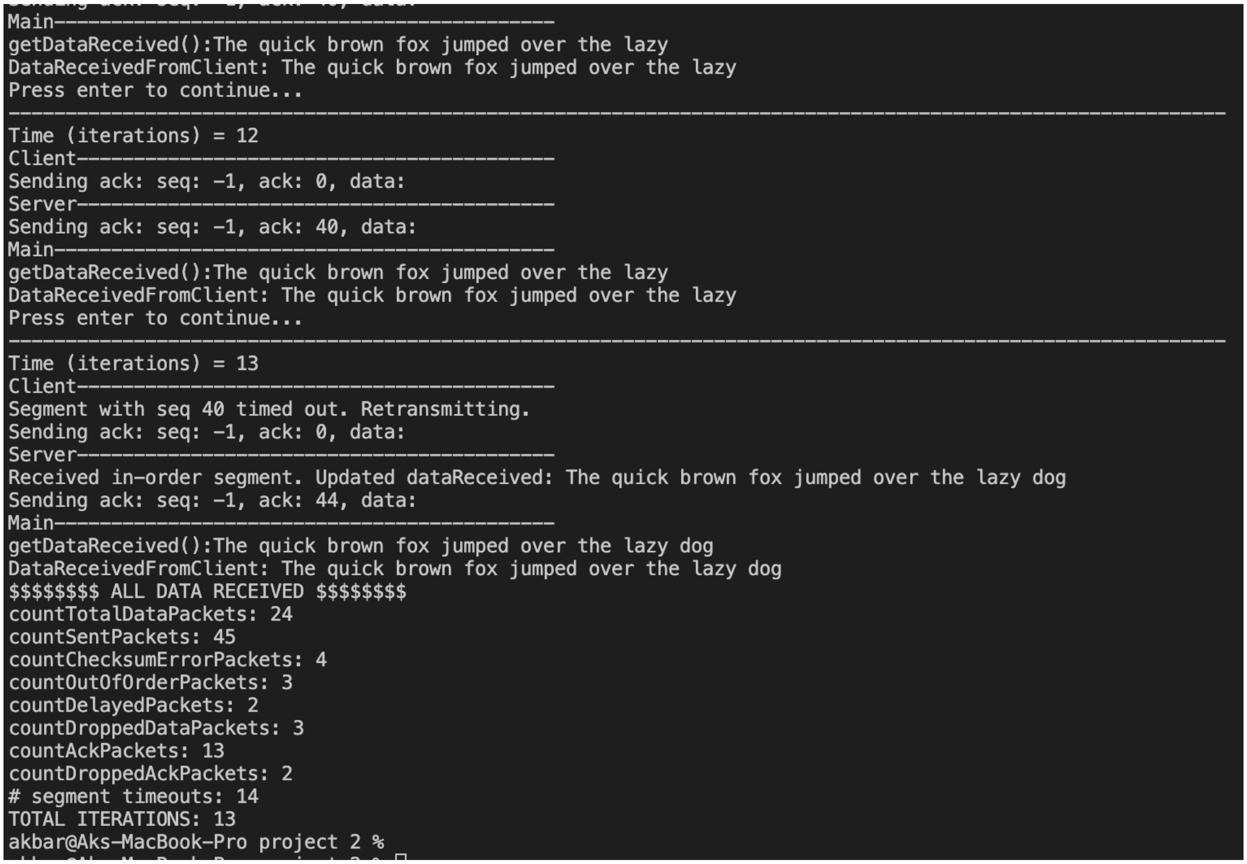
Screenshots:
Provide screenshots for short and long texts (5 pts)



```
Sending ack: seq: -1, ack: 1243, data:
Main----------------------------------------
getDataReceived():

...We choose to go to the moon. We choose to go to the moon in this decade and do the other things, not because they are easy, b
ut because they are hard, because that goal will serve to organize and measure the best of our energies and skills, because that
 challenge is one that we are willing to accept, one we are unwilling to postpone, and one which we intend to win, and the other
s, too.

...we shall send to the moon, 240,000 miles away from the control station in Houston, a giant rocket more than 300 feet tall, th
e length of this football field, made of new metal alloys, some of which have not yet been invented, capable of standing heat an
d stresses several times more than have ever been experienced, fitted together with a precision better than the finest watch, ca
rrying all the equipment needed for propulsion, guidance, control, communications, food and survival, on an untried mission, to
an unknown celestial body, and then return it safely to earth, re-entering the atmosphere at speeds of over 25,000 miles per hou
r, causing heat about half that of the temperature of the sun--almost as hot as it is here today--and do all this, and do it rig
ht, and do it first before this decade is out.

JFK - September 12, 1962

DataReceivedFromClient:

...We choose to go to the moon. We choose to go to the moon in this decade and do the other things, not because they are easy, b
ut because they are hard, because that goal will serve to organize and measure the best of our energies and skills, because that
 challenge is one that we are willing to accept, one we are unwilling to postpone, and one which we intend to win, and the other
s, too.

...we shall send to the moon, 240,000 miles away from the control station in Houston, a giant rocket more than 300 feet tall, th
e length of this football field, made of new metal alloys, some of which have not yet been invented, capable of standing heat an
d stresses several times more than have ever been experienced, fitted together with a precision better than the finest watch, ca
rrying all the equipment needed for propulsion, guidance, control, communications, food and survival, on an untried mission, to
an unknown celestial body, and then return it safely to earth, re-entering the atmosphere at speeds of over 25,000 miles per hou
r, causing heat about half that of the temperature of the sun--almost as hot as it is here today--and do all this, and do it rig
ht, and do it first before this decade is out.

JFK - September 12, 1962

$$$$$$$$ ALL DATA RECEIVED $$$$$$$$
countTotalDataPackets: 713
countSentPackets: 1428
countChecksumErrorPackets: 80
countOutOfOrderPackets: 40
countDelayedPackets: 154
countDroppedDataPackets: 100
countAckPackets: 353
countDroppedAckPackets: 39
# segment timeouts: 483
TOTAL ITERATIONS: 387
akbar@Aks-MacBook-Pro project 2 %
```



```
Received in-order segment. Updated dataReceived: The quick brown fox jumped over
Received in-order segment. Updated dataReceived: The quick brown fox jumped over the
Received in-order segment. Updated dataReceived: The quick brown fox jumped over the lazy
Checksum error in received segment: seq: 40, ack: -1, data:  Xog
Sending ack: seq: -1, ack: 40, data:
Main--------------------------------------------
getDataReceived():The quick brown fox jumped over the lazy
DataReceivedFromClient: The quick brown fox jumped over the lazy
Press enter to continue...
----------------------------------------------------------------------------------------------------
Time (iterations) = 12
Client------------------------------------------
Sending ack: seq: -1, ack: 0, data:
Server------------------------------------------
Sending ack: seq: -1, ack: 40, data:
Main--------------------------------------------
getDataReceived():The quick brown fox jumped over the lazy
DataReceivedFromClient: The quick brown fox jumped over the lazy
Press enter to continue...
----------------------------------------------------------------------------------------------------
Time (iterations) = 13
Client------------------------------------------
Segment with seq 40 timed out. Retransmitting.
Sending ack: seq: -1, ack: 0, data:
Server------------------------------------------
Received in-order segment. Updated dataReceived: The quick brown fox jumped over the lazy dog
Sending ack: seq: -1, ack: 44, data:
Main--------------------------------------------
getDataReceived():The quick brown fox jumped over the lazy dog
DataReceivedFromClient: The quick brown fox jumped over the lazy dog
$$$$$$$$ ALL DATA RECEIVED $$$$$$$$
countTotalDataPackets: 24
countSentPackets: 45
countChecksumErrorPackets: 4
countOutOfOrderPackets: 3
countDelayedPackets: 2
countDroppedDataPackets: 3
countAckPackets: 13
countDroppedAckPackets: 2
# segment timeouts: 14
TOTAL ITERATIONS: 13
akbar@Aks-MacBook-Pro project 2 %
```



```
Sending ack: seq: -1, ack: 40, data:
Main--------------------------------------------
getDataReceived():The quick brown fox jumped over the lazy
DataReceivedFromClient: The quick brown fox jumped over the lazy
Press enter to continue...
-----------------------------------------------------------------------------------------------------------
Time (iterations) = 12
Client------------------------------------------
Sending ack: seq: -1, ack: 0, data:
Server------------------------------------------
Sending ack: seq: -1, ack: 40, data:
Main--------------------------------------------
getDataReceived():The quick brown fox jumped over the lazy
DataReceivedFromClient: The quick brown fox jumped over the lazy
Press enter to continue...
-----------------------------------------------------------------------------------------------------------
Time (iterations) = 13
Client------------------------------------------
Segment with seq 40 timed out. Retransmitting.
Sending ack: seq: -1, ack: 0, data:
Server------------------------------------------
Received in-order segment. Updated dataReceived: The quick brown fox jumped over the lazy dog
Sending ack: seq: -1, ack: 44, data:
Main--------------------------------------------
getDataReceived():The quick brown fox jumped over the lazy dog
DataReceivedFromClient: The quick brown fox jumped over the lazy dog
$$$$$$$$ ALL DATA RECEIVED $$$$$$$$
countTotalDataPackets: 24
countSentPackets: 45
countChecksumErrorPackets: 4
countOutOfOrderPackets: 3
countDelayedPackets: 2
countDroppedDataPackets: 3
countAckPackets: 13
countDroppedAckPackets: 2
# segment timeouts: 14
TOTAL ITERATIONS: 13
akbar@Aks-MacBook-Pro project 2 %
akbar@Aks-MacBook-Pro project 2 % 
```