Assignment - Simple Transport Protocol Report

1) **Implementation of STP protocol – Using python3**

   a) **A Three Way Handshake (works)**

      i) Because the STP segment for connection establishment or teardown is directly pass to UDP without going to PLD module which means these segment won't loss, duplicated, corrupted, delayed or reorder. I can assume that the transmission between receiver and sender is fully reliable. Hence the implementation of the three way handshake part is by hardcoding

      ii) In the sender part, I set up the socket first and connect to receiver host IP and receiver port by UDP. Then I created a function call *handShakeInitSend()* which is send SYN to receiver, In addition, there also has *handShakeRev()* and *handShckeSend()* function which is used for receive the SYN+ACK segment send from the receiver and send back the ACK segment.

      iii) In the receiver part which is similar to the sender part. First set up the socket listen for the first SYN flag segment from receiver. After this, connect socket with sender side and start sending ACK to sender.

   b) **A four-segment connection termination (works)**

      i) Similar to the three-way handshake, the implementation of connection teardown is also hardcoding and it happens after the file been successfully transmitted.

      ii) In sender, there has function called *terminateFinSend()* and terminateACKsend which send the FIN flag segment and ACK flag segment to receiver. At the end, close the socket.

      iii) In receiver, function *terminateRcv()* used for receive segment from sender and function *terminateSend()* used for sending ACK and FIN flag segments to sender. At the end, close the socket.

   c) **Segment**

      i) Segment header (details see below)

      ii) Segment structure, segment structure is a class implemented by using ctype struct

      iii) When passing segment to UDP, convert segment structure to byte string by using memory move and use byte join to combine the segment with the data the segment need to carried.

   d) **Timer (works)**

i) I created a class called SenderTimer, has attributes EstimatedRTT, devRTT, timeoutInterval and timer, which timer is a threading.

ii) Timer has five functions. Function *cal-timeoutInterval(*) used for calculate timeout interval by passing into sample RTT

iii) *startTimer()* used for initiate new Timer for self.timer and start timing, set the timeout time using the timeout interval calculated by function before, when the timeout, call *timeout()* function

iv) *timeout()* function handle fast retransmission and call function stopTimer() to stop the current timer

e) PLD module (works)

i) I created a class called PLD, PLD initiate by passing in pDrop, pDuplicate, pCorrupt, pOrder, pDelay and keep tracked of the number of segments passed in and serval different static count like number of segments drop, number of segment delay and so on.

ii) It also *has self.savedSeg* to store the packet which is reorder, and *self.currReorder* to keep track of the number of packets been send after the reordered packet been saved.

iii) Function *make_decision()* it used for make decision of if a segment should be drop, duplicate, corrupt, reorder, delay or successfully send it to UDP

iv) Function *dropSegment(), duplicateSegment(), corruptedSegment(), reOrderSegment(), delaySegment()* are used for actually conduct the behaviour. And function forwardingSegment() will be used in the sender for make a decision of the behaviour of a segment and actually conduct it.

f) Sender

i) Has simple timer called STPtimer

ii) Has PLD module called PLD

iii) Can record the information of segment been received or sent to log file consistently

iv) Can deal with different MSS

(1) Before sending the file, read the whole file in bytes and store it in buffer called *fileBytes*. Function *getDataChunk()* can help to get the data chunk need to been send start from the index lastByteAck, the size is depends, it can be MSS or smaller than MSS (only happen in last chunk)

v) Maximum Window Size

(1) The segment can only been send when lastByteSent – lastByteAck <= MWS. First get the data chunk need to send by using *getDataChunk(lastByteAck)*, then calculate the checksum (here involved function *changeByteToInt()* to change the byte string to integer), generate the header and segment, update the global variables, pass segment and data to the PLD module. Inside the PLD module, it will encapsulated the packet and determine the conduction of them.

vi) when the non-acked segments size reach MWS, if the timer timeout, restart the timer, otherwise waiting for the ACK from receiver. Once the required ACK been received, recalculated the sample RTT, stop current timer.

vii) Fast retransformation

(1) Has a dictionary called *receivedACK*, which store the number of times of each ACK received, if the number of times a same ACK been received for 3 times, retransmit the segment has sequence number of the ACK

viii) Time out fast retransformation – handled by PLD module

ix) The while loop to calculate SampleRTT in sender file seems not correct.

g) Receiver

i) Open for listening the segments send from sender

ii) Can record the information of segment been received or sent to log file consistently

iii) Has function *getSegmentAndData()* to extract the segment header and data

iv) Has function *changeByteToInt()* and *rs232_checksum(data)* to detect if the received packet is corrupted or not, if the result is the same as the checksum stored in the header, data doesn't corrupted.

v) Has list called buff to store the reorder packet, another list called received used to store the sequence number of each segment whose data has been write to the file.

vi) After a segment been received, if the sequence number been received is the same as the expected one, update the ack number and send ACK to sender. Otherwise store the packet to buff, ack number keeps the same.

vii) At the end of the retransformation, will create a new file called file_r.pdf. First create a new file, then write byte to file when the segment received meet the requirement.

2) STP header and a quick explanation of all fields

<table>
<tr><td colspan="2">sequence number (8 bytes)</td></tr>
<tr><td colspan="2">acknowledgement number (8 bytes)</td></tr>
<tr><td>length (4 bytes)</td><td>checksum (4 bytes)</td></tr>
<tr><td>types(4 bytes)</td><td></td></tr>
</table>

Figure 1.1

i) Header

(a) Because the STP segment has header and data carried, I generate a new class called The reason I used structure type here is because I can change the structure type to byte string format and can send it through the socket. (details see the diagram below)

(b) checksum is an integer which calculated by using function *rs232_checksum()* by passing through the data carried in bytes string by using checksum function from python3 library, used for receiver to detect if the received segment is corrupted or not.

(c) type is same as the flags of each segment, in my implementation, there has only 6 types for the segment, which is S(SYN), A(ACK), F(FIN), D(DATA), SA(SYN+ACK), FA(FIN+ACK). I use different integer (base 2) to present each flag (1, 2, 4, 8), hence if the flags of a segment is combination of two types of flag, receiver can still detect it.

3) Design trade-offs considered and made. improvements and extensions to your program and indicate how you could realise them.

- My design currently has many duplicated place and very complex not flexible enough, like my three way handshake and teardown part is hardcoding and my sender and receiver has so many duplicated part. In addition, the way I create segment is to complex. If I have more time, I'll find a way to put actual data into the segment data filed instead of use byte join to link them together. And I'll make a separate file called PLD module instead of put it in the sender file.

4) Indicate any segments of code that you have borrowed from the Web or other books.

- Function *rs232_checksum(), changeByteToInt()*

5) **Answer the following questions**: (include any output as an appendix to the main report.pdf, appendix is not included in the 5-page limit)

(a) Run your protocol using pDrop = 0.1, MWS = 500 bytes, MSS = 100 bytes, seed = 100, gamma = 4, and pDuplicate, pCorrupt, pOrder, MaxOrder, pDelay, MaxDelay all set to 0. Transfer the file test0.pdf (available on the assignment webpage). Run an additional experiment with pdrop = 0.3, transferring the same file (test0.pdf).

- When pDrop = 0.1, I found out that the receiver doesn't receive the packet of sequence number 201, then it send duplicated ACK has ack number 201, which means segment has sequence number 201 was dropped. Similarly, packet with sequence number 2101 was dropped. Also packet of sequence number 2901 and 3001 were dropped. And the duplicate ACKs sent is 6 and 4 packets was dropped
- When pDrop = 0.3, I found out that segment with sequence number 1, 401, 601, 501, 901, 1001, 1301, 1901, 2101, 2201, 2501, 2901, 3001 were dropped, there are in total 13 packets been dropped. Which satisfy when the pDrop is larger, there has more packets going to be dropped.

(b)

	gamma = 2	gamma = 4	gamma = 6
Number of STP packets been transmitted	9217	9217	9217
The time overall transfer took (s)	3437.08	6213.93	10450.24

- The number of STP packets been transmitted for all the three tests is the same, all of them is 9217 packets been transmitted (include RXT and drop), but if the gamma is bigger, the time took to transfer all the packets is longer. When gamma is 4, the transfer time is nearly the two times of the transfer time when set gamma to 2, similar to when the test has gamma value of 6.
- In conclusion, when gamma is bigger, the transformation time is longer because the timeout interval is longer.

(c) Yes, the file has been successfully transferred. (Because the file is too big and I do not have time because the deadline is coming soon, hence I change the timeout interval calculation to cut the transformation time), the most critical contributing most in the overall transfer time is pDrop, because the other 3 doesn't result in the timeout which means the packet still send to the receiver successfully.

Appendix

sender log file of the pDrop = 0.1, MWS = 500 bytes, MSS = 100 bytes, seed = 100, gamma = 4, and pDuplicate, pCorrupt, pOrder, MaxOrder, pDelay, MaxDelay all set to 0.

```
snd                  0.00       S          0          0          0
rcv                  0.00       SA         0          0          1
snd                  0.00       A          1          0          1
snd                  0.00       D          1          100        1
snd                  0.00       D          101        100        1
drop                 0.00       D          201        100        1
snd                  0.00       D          301        100        1
snd                  0.00       D          401        100        1
rcv                  0.00       A          1          0          101
snd                  0.01       D          501        100        1
rcv                  0.01       A          1          0          201
snd                  0.01       D          601        100        1
rcv/DA               0.01       A          1          0          201
rcv/DA               0.01       A          1          0          201
rcv/DA               0.02       A          1          0          201
snd/RXT              0.02       D          201        100        1
rcv                  0.02       A          1          0          701
snd                  0.02       D          701        100        1
snd                  0.02       D          801        100        1
snd                  0.02       D          901        100        1
snd                  0.02       D          1001       100        1
snd                  0.02       D          1101       100        1
rcv                  0.02       A          1          0          801
snd                  0.03       D          1201       100        1
rcv                  0.03       A          1          0          901
snd                  0.03       D          1301       100        1
rcv                  0.03       A          1          0
1001
snd                  0.04       D          1401       100        1
rcv                  0.04       A          1          0
1101
snd                  0.04       D          1501       100        1
rcv                  0.04       A          1          0
1201
snd                  0.04       D          1601       100        1
rcv                  0.04       A          1          0
1301
snd                  0.05       D          1701       100        1
rcv                  0.05       A          1          0
1401
snd                  0.05       D          1801       100        1
rcv                  0.05       A          1          0
1501
snd                  0.05       D          1901       100        1
rcv                  0.05       A          1          0
1601
snd                  0.06       D          2001       100        1
rcv                  0.06       A          1          0
```

```
1701
drop                 0.06       D          2101       100        1
rcv                  0.06       A          1          0
1801
snd                  0.07       D          2201       100        1
rcv                  0.07       A          1          0
1901
snd                  0.07       D          2301       100        1
rcv                  0.07       A          1          0
2001
snd                  0.07       D          2401       100        1
rcv                  0.07       A          1          0
2101
snd                  0.08       D          2501       100        1
rcv/DA               0.08       A          1          0
2101
rcv/DA               0.08       A          1          0
2101
rcv/DA               0.08       A          1          0
2101
snd/RXT              0.09       D          2101       100        1
rcv                  0.09       A          1          0
2601
snd                  0.09       D          2601       100        1
snd                  0.09       D          2701       100        1
snd                  0.09       D          2801       100        1
drop                 0.09       D          2901       100        1
drop                 0.09       D          3001       28         1
rcv                  0.09       A          1          0
2701
rcv                  0.09       A          1          0
2801
rcv                  0.10       A          1          0
2901
snd/RXT              1.79       D          2901       100        1
rcv                  1.79       A          1          0
3001
snd/RXT              4.38       D          3001       28         1
rcv                  4.39       A          1          0
3029
snd                  4.39       F          3029       0          1
rcv                  4.39       A          1          0
3030
rcv                  4.39       F          1          0
3030
snd                  4.39       A          3030       0          2
========================================================
Size of the file (in Bytes) 3028
Segments transmitted (including drop & RXT) 39
Number of Segments handled by PLD 35
Number of Segments dropped 4
Number of Segments Corrupted 0
Number of Segments Re-ordered 0
Number of Segments Duplicated 0
Number of Segments Delayed 0
Number of Retransmissions due to TIMEOUT 2
Number of FAST RETRANSMISSION 2
Number of DUP ACKS received 6
========================================================
```

log file of the pDrop = 0.1, MWS = 500 bytes, MSS = 100 bytes, seed = 100, gamma = 4, and pDuplicate, pCorrupt, pOrder, MaxOrder, pDelay, MaxDelay all set to 0.

```
rcv                  0.00       S          0          0          0
snd                  0.00       SA         0          0          1
rcv                  0.00       A          1          0          1
rcv                  0.00       D          1          100        1
snd                  0.00       A          1          0          101
rcv                  0.00       D          101        100        1
snd                  0.00       A          1          0          201
rcv                  0.00       D          301        100        1
snd/DA               0.00       A          1          0          201
rcv                  0.00       D          401        100        1
snd/DA               0.00       A          1          0          201
rcv                  0.01       D          501        100        1
snd/DA               0.01       A          1          0          201
rcv                  0.01       D          601        100        1
rcv                  0.02       D          201        100        1
snd                  0.02       A          1          0          701
rcv                  0.02       D          701        100        1
snd                  0.02       A          1          0          801
rcv                  0.02       D          801        100        1
snd                  0.02       A          1          0          901
rcv                  0.02       D          901        100        1
snd                  0.02       A          1          0
1001
rcv                  0.02       D          1001       100        1
snd                  0.02       A          1          0
1101
rcv                  0.02       D          1101       100        1
snd                  0.02       A          1          0
1201
rcv                  0.03       D          1201       100        1
snd                  0.03       A          1          0
1301
rcv                  0.03       D          1301       100        1
snd                  0.03       A          1          0
1401
rcv                  0.03       D          1401       100        1
snd                  0.03       A          1          0
1501
rcv                  0.04       D          1501       100        1
snd                  0.04       A          1          0
1601
rcv                  0.04       D          1601       100        1
snd                  0.04       A          1          0
1701
rcv                  0.05       D          1701       100        1
snd                  0.05       A          1          0
1801
rcv                  0.05       D          1801       100        1
snd                  0.05       A          1          0
1901
```

```
rcv                  0.05       D          1901       100        1
snd                  0.05       A          1          0
2001
rcv                  0.06       D          2001       100        1
snd                  0.06       A          1          0
2101
rcv                  0.06       D          2201       100        1
snd/DA               0.06       A          1          0
2101
rcv                  0.07       D          2301       100        1
snd/DA               0.07       A          1          0
2101
rcv                  0.07       D          2401       100        1
snd/DA               0.07       A          1          0
2101
rcv                  0.08       D          2501       100        1
rcv                  0.09       D          2101       100        1
snd                  0.09       A          1          0
2601
rcv                  0.09       D          2601       100        1
snd                  0.09       A          1          0
2701
rcv                  0.09       D          2701       100        1
snd                  0.09       A          1          0
2801
rcv                  0.09       D          2801       100        1
snd                  0.09       A          1          0
2901
rcv                  1.79       D          2901       100        1
snd                  1.79       A          1          0
3001
rcv                  4.38       D          3001       28         1
snd                  4.38       A          1          0
3029
rcv                  4.39       F          3029       0          1
snd                  4.39       A          1          0
3030
snd                  4.39       F          1          0
3030
rcv                  4.39       A          3030       0          2
========================================================
Amount of data received (bytes) 3028
Total Segments Received 35
Data segments received 31
Data segments with Bit Errors 0
Duplicate data segments received 0
Duplicate ACKs sent 6
========================================================
```

sender log file of the pDrop = 0.3, MWS = 500 bytes, MSS = 100 bytes, seed = 100, gamma = 4, and pDuplicate, pCorrupt, pOrder, MaxOrder, pDelay, MaxDelay all set to 0.

```
snd                  0.00       S          0          0          0
rcv                  0.00       SA         0          0          1
snd                  0.00       A          1          0          1
drop                 0.00       D          1          100        1
snd                  0.00       D          101        100        1
snd                  0.00       D          201        100        1
snd                  0.00       D          301        100        1
drop                 0.00       D          401        100        1
rcv                  0.00       A          1          0          1
rcv/DA               0.01       A          1          0          1
rcv/DA               0.01       A          1          0          1
snd/RXT              1.70       D          1          100        1
rcv                  1.70       A          1          0          401
drop                 1.70       D          501        100        1
drop                 1.70       D          601        100        1
snd                  1.70       D          701        100        1
snd                  1.70       D          801        100        1
rcv/DA               1.70       A          1          0          401
rcv/DA               1.71       A          1          0          401
snd/RXT              3.40       D          401        100        1
rcv                  3.40       A          1          0          501
drop                 3.41       D          901        100        1
snd/RXT              6.00       D          501        100        1
rcv                  6.00       A          1          0          601
drop                 6.01       D          1001       100        1
snd/RXT              9.62       D          601        100        1
rcv                  9.62       A          1          0          901
snd                  9.62       D          1101       100        1
snd                  9.62       D          1201       100        1
drop                 9.62       D          1301       100        1
rcv/DA               9.62       A          1          0          901
rcv/DA               9.63       A          1          0          901
snd/RXT              11.32      D          901        100        1
rcv                  11.32      A          1          0
1001
snd                  11.32      D          1401       100        1
rcv/DA               11.32      A          1          0
1001
snd/RXT              13.02      D          1001       100        1
rcv                  13.02      A          1          0
1301
snd                  13.02      D          1501       100        1
snd                  13.02      D          1601       100        1
snd                  13.02      D          1701       100        1
rcv/DA               13.02      A          1          0
1301
rcv/DA               13.02      A          1          0
1301
rcv/DA               13.03      A          1          0
1301
```

```
snd/RXT              13.03      D          1301       100        1
rcv                  13.03      A          1          0
1801
snd                  13.03      D          1801       100        1
drop                 13.03      D          1901       100        1
snd                  13.03      D          2001       100        1
drop                 13.03      D          2101       100        1
drop                 13.03      D          2201       100        1
rcv                  13.03      A          1          0
1901
snd                  13.03      D          2301       100        1
rcv/DA               13.03      A          1          0
1901
rcv/DA               13.04      A          1          0
1901
snd/RXT              14.73      D          1901       100        1
rcv                  14.73      A          1          0
2101
snd                  14.74      D          2401       100        1
drop                 14.74      D          2501       100        1
rcv/DA               14.74      A          1          0
2101
snd/RXT              16.43      D          2101       100        1
rcv                  16.43      A          1          0
2201
snd                  16.43      D          2601       100        1
rcv/DA               16.43      A          1          0
2201
snd/RXT              18.13      D          2201       100        1
rcv                  18.13      A          1          0
2501
snd                  18.13      D          2701       100        1
snd                  18.13      D          2801       100        1
drop                 18.13      D          2901       100        1
rcv/DA               18.13      A          1          0
2501
rcv/DA               18.13      A          1          0
2501
snd/RXT              19.83      D          2501       100        1
rcv                  19.83      A          1          0
2901
drop                 19.83      D          3001       28         1
snd/RXT              22.43      D          2901       100        1
rcv                  22.43      A          1          0
3001
snd/RXT              26.04      D          3001       28         1
rcv                  26.04      A          1          0
3029
snd                  26.04      F          3029       0          1
rcv                  26.04      A          1          0
3030
rcv                  26.04      F          1          0
3030
snd                  26.04      A          3030       0          2
```

```
========================================================
Size of the file (in Bytes) 3028
Segments transmitted (including drop & RXT) 48
Number of Segments handled by PLD 44
Number of Segments dropped 13
Number of Segments Corrupted 0
Number of Segments Re-ordered 0
Number of Segments Duplicated 0
Number of Segments Delayed 0
Number of Retransmissions due to TIMEOUT 12
Number of FAST RETRANSMISSION 1
Number of DUP ACKS received 16
========================================================
```

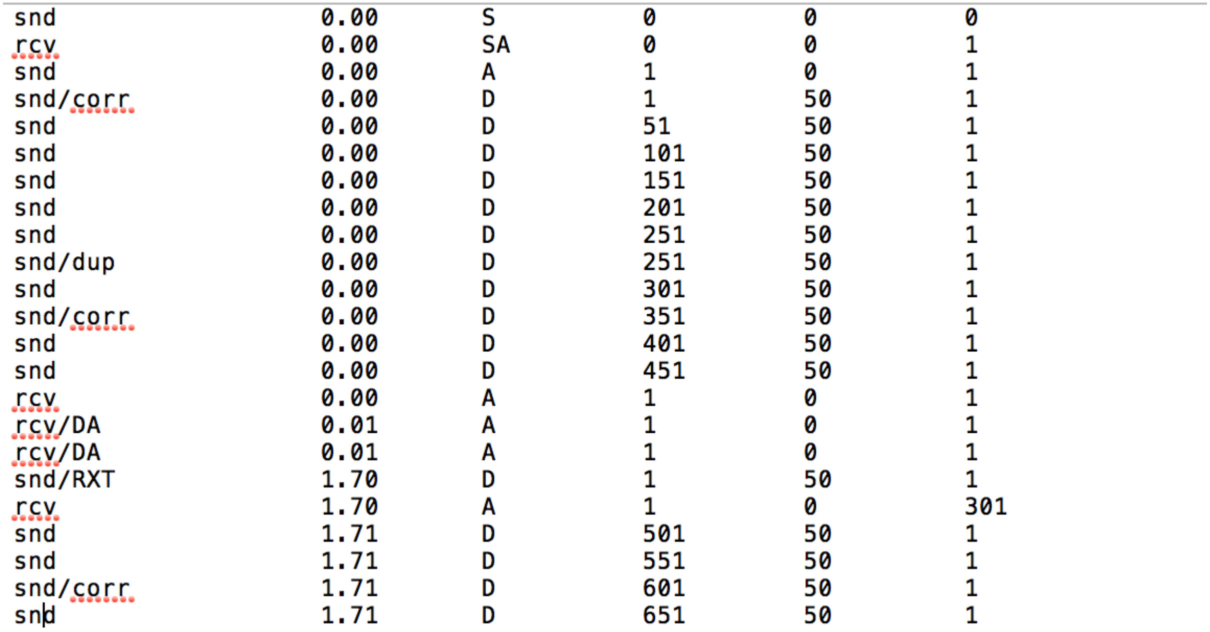
receiver log file of the pDrop = 0.3, MWS = 500 bytes, MSS = 100 bytes, seed = 100, gamma = 4, and pDuplicate, pCorrupt, pOrder, MaxOrder, pDelay, MaxDelay all set to 0.

```
rcv                  0.00       S          0          0          0
snd                  0.00       SA         0          0          1
rcv                  0.00       A          1          0          1
rcv                  0.00       D          101        100        1
snd/DA               0.00       A          1          0          1
rcv                  0.00       D          201        100        1
snd/DA               0.00       A          1          0          1
rcv                  0.00       D          301        100        1
snd/DA               0.00       A          1          0          1
rcv                  1.70       D          1          100        1
snd                  1.70       A          1          0          401
rcv                  1.70       D          701        100        1
snd/DA               1.70       A          1          0          401
rcv                  1.70       D          801        100        1
snd/DA               1.70       A          1          0          401
rcv                  3.40       D          401        100        1
snd                  3.40       A          1          0          501
rcv                  6.00       D          501        100        1
snd                  6.00       A          1          0          601
rcv                  9.62       D          601        100        1
snd                  9.62       A          1          0          901
rcv                  9.62       D          1101       100        1
snd/DA               9.62       A          1          0          901
rcv                  9.62       D          1201       100        1
snd/DA               9.62       A          1          0          901
rcv                  11.32      D          901        100        1
snd                  11.32      A          1          0
1001
rcv                  11.32      D          1401       100        1
snd/DA               11.32      A          1          0
1001
rcv                  13.01      D          1001       100        1
snd                  13.02      A          1          0
1301
rcv                  13.02      D          1501       100        1
snd/DA               13.02      A          1          0
1301
rcv                  13.02      D          1601       100        1
snd/DA               13.02      A          1          0
1301
rcv                  13.02      D          1701       100        1
snd/DA               13.02      A          1          0
1301
rcv                  13.03      D          1301       100        1
snd                  13.03      A          1          0
1801
rcv                  13.03      D          1801       100        1
snd                  13.03      A          1          0
1901
```

```
rcv                  13.03      D          2001       100        1
snd/DA               13.03      A          1          0
1901
rcv                  13.03      D          2301       100        1
snd/DA               13.03      A          1          0
1901
rcv                  14.73      D          1901       100        1
snd                  14.73      A          1          0
2101
rcv                  14.74      D          2401       100        1
snd/DA               14.74      A          1          0
2101
rcv                  16.43      D          2101       100        1
snd                  16.43      A          1          0
2201
rcv                  16.43      D          2601       100        1
snd/DA               16.43      A          1          0
2201
rcv                  18.13      D          2201       100        1
snd                  18.13      A          1          0
2501
rcv                  18.13      D          2701       100        1
snd/DA               18.13      A          1          0
2501
rcv                  18.13      D          2801       100        1
snd/DA               18.13      A          1          0
2501
rcv                  19.83      D          2501       100        1
snd                  19.83      A          1          0
2901
rcv                  22.43      D          2901       100        1
snd                  22.43      A          1          0
3001
rcv                  26.04      D          3001       28         1
snd                  26.04      A          1          0
3029
rcv                  26.04      F          3029       0          1
snd                  26.04      A          1          0
3030
snd                  26.04      F          1          0
3030
rcv                  26.04      A          3030       0          2
========================================================
Amount of data received (bytes) 3028
Total Segments Received 35
Data segments received 31
Data segments with Bit Errors 0
Duplicate data segments received 0
Duplicate ACKs sent 17
========================================================
```

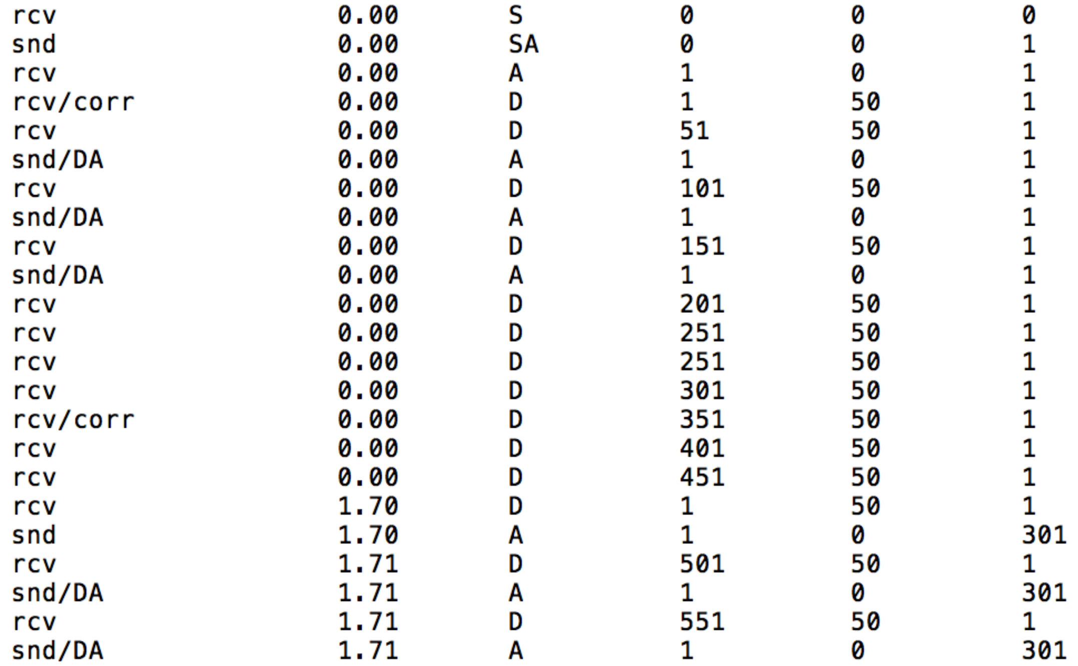
Part (c)

Connection + first 20 segments in sender_log.txt



```
snd         0.00    S    0     0     0
rcv         0.00    SA   0     0     1
snd         0.00    A    1     0     1
snd/corr    0.00    D    1     50    1
snd         0.00    D    51    50    1
snd         0.00    D    101   50    1
snd         0.00    D    151   50    1
snd         0.00    D    201   50    1
snd         0.00    D    251   50    1
snd/dup     0.00    D    251   50    1
snd         0.00    D    301   50    1
snd/corr    0.00    D    351   50    1
snd         0.00    D    401   50    1
snd         0.00    D    451   50    1
rcv         0.00    A    1     0     1
rcv/DA      0.01    A    1     0     1
rcv/DA      0.01    A    1     0     1
snd/RXT     1.70    D    1     50    1
rcv         1.70    A    1     0     301
snd         1.71    D    501   50    1
snd         1.71    D    551   50    1
snd/corr    1.71    D    601   50    1
snd         1.71    D    651   50    1
```

Connection + first 20 segments in receiver_log.txt



```
rcv         0.00    S    0     0     0
snd         0.00    SA   0     0     1
rcv         0.00    A    1     0     1
rcv/corr    0.00    D    1     50    1
rcv         0.00    D    51    50    1
snd/DA      0.00    A    1     0     1
rcv         0.00    D    101   50    1
snd/DA      0.00    A    1     0     1
rcv         0.00    D    151   50    1
snd/DA      0.00    A    1     0     1
rcv         0.00    D    201   50    1
rcv         0.00    D    251   50    1
rcv         0.00    D    251   50    1
rcv         0.00    D    301   50    1
rcv/corr    0.00    D    351   50    1
rcv         0.00    D    401   50    1
rcv         0.00    D    451   50    1
rcv         1.70    D    1     50    1
snd         1.70    A    1     0     301
rcv         1.71    D    501   50    1
snd/DA      1.71    A    1     0     301
rcv         1.71    D    551   50    1
snd/DA      1.71    A    1     0     301
```