

## Report COMP3331

1. A brief discussion of how you have implemented the STP protocol. Provide a list of features that you have successfully implemented. In case you have not been able to get certain features of STP working, you should also mention that in your report.

I implemented my STP protocol using two java files. The arguments for running each of the program are stored and used to define how it functions. The main functions of the file are broken into stages of handshaking, closing the connection and sending data. The data sending section is further broken up to process segments that are received, determining what kind of errors have occurred and the responses required. A separate PLDmodule method is called in the data processing section, it takes in defining arguments to model errors occurring.

Several special classes were used to implement special functionality. The files use PrintStreams to write strings containing transfer metadata to generate the text logs. Writes were written after each call of socket.send and socket.receive. FileInputStreams and FileOutputStreams were used to read and write the files that were being transferred. ByteBuffers allowed for the header and file data to be placed inside a buffer to be transferred. Timeouts are measured using the Timer and TimerTasks classes, these handle timeouts by retransmitting the previous packet. Various getters were also added for ease of use.

Features Successfully implemented:

- Three way handshake
- Four segment tear down
- Header which contains relevant metatimer
- Single parallel thread timer that determines timeouts
- Proper sequence numbers and acknowledgement number calculations
- Log Files for sender and receiver
- Using random seed to allow rerun of tests
- Error generation and processing with PLDmodule implementation, including dropped packets, duplicated packets, corrupted packets and delayed packets. Reordered packets were not implemented as pipelining was not implemented
- Functioning checksum for flipped bits
- Correct transfer of file

2. A detailed diagram of your STP header and a quick explanation of all fields (similar to the diagrams that we have used in the lectures to understand TCP/UDP headers).

The ByteBuffer allowed different data types to be stored into the segment along with the data. Due to the fact that ByteBuffers were used, bits were unable to be used as only bytes and other primitive data could be stored within the ByteBuffer. The header contains the data as shown in the table.

32bits		32bits	
Sequence Number (32bits)		Acknowledgement Number (32 bits)	
Flags (8 bits)	MWS (32 bits)		MSS (24 bits)
MSS (8 bits)	Checksum (32 bits)		Data (24 bits)

Data (up to MSS – 24 bits)
----------------------------

The sequence number and acknowledgement number function as defined in theory. The flags are SYN\_FLAG = 0x8 and ACK\_FLAG = 0x4 which can be bitwise OR'd to form the SYNACK\_FLAG.

3. Discuss any design trade-offs considered and made. Describe possible improvements and Extensions to your program and indicate how you could realise them.

Pipelining was not implemented which resulted in many advanced features of TCP. An obvious improvement would be to implement pipelining so that the maximum window size will become relevant through features such as fast transmission. This could be realised through the use of multithreading which I was unable to do because I am not familiar with how it worked and did not understand online tutorials.

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

An example of how to run the timer scheduler was taken from stack overflow:

```
Timer timer = new Timer();
To run the task once you would do:

timer.schedule(new TimerTask() {
    @Override
    public void run() {
        // Your database code here
    }
}, 2*60*1000);
```

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)

- i) 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). The file should be received correctly at the Receiver. Show the sequence of STP packets that are observed at the Receiver. It is sufficient to just indicate the sequence numbers of the STP packets that have arrived.
- ii) Run an additional experiment with pdrop = 0.3, transferring the same file (test0.pdf). In your report, discuss the resulting packet sequences of both experiments indicating where dropping occurred. Also, in the appendix section show the packet sequences for both the experiments.

The packet sequences differ in transmission time from 0.11seconds to 7.50 seconds with pDrop = 0.3. The reason for this delay is due to timeouts occurring. It can be noted that the delayed time after each drop decreases, going from 1.2 seconds to only 0.3 seconds due to the timeoutInterval being recalculated.

(b) The timeout for STP is given by:

$\text{TimeoutInterval} = \text{EstimatedRTT} + \text{gamma} * \text{DevRTT}$

where gamma will be supplied to the program as an input argument, see Section 4.5.

Set pdrop = 0.5, MWS = 500 bytes, MSS = 50 bytes, seed = 300, pdelay = 0.2, MaxDelay = 1000 and pDuplicate, pCorrupt, pOrder, MaxOrder all set to 0. Run three experiments with the following different gamma values:

i. gamma = 2

ii. gamma = 4

iii. gamma = 6

and transfer the file test1.pdf using STP. Show a table that indicates how many STP packets were transmitted in total and how long the overall transfer took. Discuss the results.

Gamma	STP Packets Transferred	Time of transfer (seconds)
2	9579	1456.06
4	9408	2559.77
6	6292	308205

As the value of gamma increases, the time of transfer also increases. By increasing the value of gamma, the timeoutIntervals ends up increasing. Because timeoutIntervals are increased and fast retransmit has not been implemented, more time is spent waiting when the packet has dropped which explains why the time of transfer increases.

(c) Use the following values and run STP to transfer test2.pdf.

MWS=500bytes MSS=50 gamma=4 pDrop=0.1 pDuplicate=0.1 pCorrupt=0.1 pOrder=0.1 maxOrder=4  
pDelay=0 maxDelay=0 seed=300

Has the file been successfully transferred? How long the overall transfer took? For this experiment, which of the factor (out of pDrop, pDuplicate, pCorrupt and pOrder) is the most critical contributing most in the overall transfer time? How have you determined this? Provide the screen shot for the initial transfer (connection establishment + first 20 entries) and the last 20 entries plus the summary statistics table for the sender\_log.txt and receiver\_log.txt files in appendix. Do not attach the complete log files due to their sizes.

The file could not transfer successfully.

## Appendix

Q5a) part i) pDrop = 0

rcv	0.00	S	0	0	0
snd	0.03	S	1	0	1
rcv	0.03	A	1	0	1
rcv	0.04	D	1	100	1
snd	0.04	A	1	0	101
rcv	0.04	D	101	100	1
snd	0.04	A	1	0	201
rcv	0.04	D	201	100	1
snd	0.04	A	1	0	301
rcv	0.04	D	301	100	1
snd	0.05	A	1	0	401
rcv	0.05	D	401	100	1
snd	0.05	A	1	0	501
rcv	0.05	D	501	100	1
snd	0.05	A	1	0	601
rcv	0.05	D	601	100	1
snd	0.05	A	1	0	701
rcv	0.06	D	701	100	1
snd	0.06	A	1	0	801
rcv	0.06	D	801	100	1
snd	0.06	A	1	0	901
rcv	0.06	D	901	100	1
snd	0.06	A	1	0	1001
rcv	0.06	D	1001	100	1
snd	0.06	A	1	0	1101
rcv	0.06	D	1101	100	1
snd	0.07	A	1	0	1201
rcv	0.07	D	1201	100	1
snd	0.07	A	1	0	1301
rcv	0.07	D	1301	100	1
snd	0.07	A	1	0	1401
rcv	0.07	D	1401	100	1
snd	0.07	A	1	0	1501
rcv	0.07	D	1501	100	1
snd	0.07	A	1	0	1601
rcv	0.08	D	1601	100	1
snd	0.08	A	1	0	1701
rcv	0.08	D	1701	100	1
snd	0.08	A	1	0	1801
rcv	0.08	D	1801	100	1
snd	0.08	A	1	0	1901
rcv	0.08	D	1901	100	1
snd	0.08	A	1	0	2001
rcv	0.09	D	2001	100	1
snd	0.09	A	1	0	2101

rcv	0.09	D	2101	100	1
snd	0.09	A	1	0	2201
rcv	0.09	D	2201	100	1
snd	0.09	A	1	0	2301
rcv	0.09	D	2301	100	1
snd	0.10	A	1	0	2401
rcv	0.10	D	2401	100	1
snd	0.10	A	1	0	2501
rcv	0.10	D	2501	100	1
snd	0.10	A	1	0	2601
rcv	0.10	D	2601	100	1
snd	0.10	A	1	0	2701
rcv	0.10	D	2701	100	1
snd	0.10	A	1	0	2801
rcv	0.10	D	2801	100	1
snd	0.10	A	1	0	2901
rcv	0.11	D	2901	100	1
snd	0.11	A	1	0	3001
rcv	0.11	D	3001	28	1
snd	0.11	A	1	0	3029
rcv	0.11	F	3029	0	2
snd	0.11	F	1	0	3030
snd	0.11	A	1	0	3030
rcv	0.11	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 0

5a) part ii) pDrop = 0.3

rcv	0.00	S	0	0	0
snd	0.03	S	1	0	1
rcv	0.03	A	1	0	1
rcv	0.03	D	1	100	1
snd	0.03	A	1	0	101
rcv	0.03	D	101	100	1
snd	0.03	A	1	0	201
rcv	0.04	D	201	100	1
snd	0.04	A	1	0	301
rcv	1.78	D	301	100	1
snd	1.78	A	1	0	401
rcv	3.53	D	401	100	1
snd	3.53	A	1	0	501
rcv	3.53	D	501	100	1

snd	3.53	A	1	0	601
rcv	3.54	D	601	100	1
snd	3.54	A	1	0	701
rcv	3.54	D	701	100	1
snd	3.54	A	1	0	801
rcv	3.54	D	801	100	1
snd	3.54	A	1	0	901
rcv	3.54	D	901	100	1
snd	3.54	A	1	0	1001
rcv	3.55	D	1001	100	1
snd	3.55	A	1	0	1101
rcv	4.67	D	1101	100	1
snd	4.68	A	1	0	1201
rcv	5.80	D	1201	100	1
snd	5.80	A	1	0	1301
rcv	5.80	D	1301	100	1
snd	5.81	A	1	0	1401
rcv	5.81	D	1401	100	1
snd	5.81	A	1	0	1501
rcv	5.81	D	1501	100	1
snd	5.81	A	1	0	1601
rcv	6.62	D	1601	100	1
snd	6.62	A	1	0	1701
rcv	6.62	D	1701	100	1
snd	6.62	A	1	0	1801
rcv	6.62	D	1801	100	1
snd	6.62	A	1	0	1901
rcv	6.63	D	1901	100	1
snd	6.63	A	1	0	2001
rcv	6.63	D	2001	100	1
snd	6.63	A	1	0	2101
rcv	6.63	D	2101	100	1
snd	6.63	A	1	0	2201
rcv	6.63	D	2201	100	1
snd	6.63	A	1	0	2301
rcv	6.64	D	2301	100	1
snd	6.64	A	1	0	2401
rcv	6.98	D	2401	100	1
snd	6.98	A	1	0	2501
rcv	6.98	D	2501	100	1
snd	6.98	A	1	0	2601
rcv	7.29	D	2601	100	1
snd	7.29	A	1	0	2701
rcv	7.29	D	2701	100	1
snd	7.29	A	1	0	2801
rcv	7.29	D	2801	100	1
snd	7.29	A	1	0	2901
rcv	7.29	D	2901	100	1

snd	7.29	A	1	0	3001
rcv	7.50	D	3001	28	1
snd	7.50	A	1	0	3029
rcv	7.50	F	3029	0	2
snd	7.50	F	1	0	3030
snd	7.50	A	1	0	3030
rcv	7.50	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 0

=====

b) i)  $\gamma = 0.2$

Receiver\_log

rcv	0.00	S	0	0	0
snd	0.03	S	1	0	1
rcv	0.03	A	1	0	1
rcv	0.03	D	1	50	1
snd	0.03	A	1	0	51
rcv	0.04	D	51	50	1
snd	0.04	A	1	0	101
rcv	1.11	D	101	50	1
snd	1.11	A	1	0	151
rcv	1.11	D	151	50	1
snd	1.11	A	1	0	201
rcv	1.11	D	201	50	1
snd	1.11	A	1	0	251
rcv	1.12	D	251	50	1
snd	1.12	A	1	0	301
rcv	1.12	D	301	50	1
snd	1.12	A	1	0	351
rcv	1.12	D	351	50	1
snd	1.12	A	1	0	401
rcv	1.12	D	401	50	1
...					
rcv	1455.59	D	307901	50	1
snd	1455.59	A	1	0	307951
rcv	1455.59	D	307951	50	1
snd	1455.59	A	1	0	308001
rcv	1455.75	D	308001	50	1
snd	1455.75	A	1	0	308051
rcv	1455.75	D	308051	50	1
snd	1455.75	A	1	0	308101
rcv	1455.90	D	308101	50	1
snd	1455.90	A	1	0	308151

rcv	1456.05	D	308151	50	1
snd	1456.05	A	1	0	308201
rcv	1456.05	D	308201	3	1
snd	1456.05	A	1	0	308204
rcv	1456.05	F	308204	0	2
snd	1456.05	F	1	0	308205
snd	1456.05	A	1	0	308205
rcv	1456.05	A	308205	0	2

=====

Amount of data received (bytes) 327503

Total Segments Received 6555

Data segments received 6551

Data segments with Bit Errors 0

Duplicate data segments received 386

Duplicate ACKs sent 386

=====

#### Sender\_log

Snd	0.01	S	0	0	0
rcv	0.04	SA	0	0	1
snd	0.04	A	1	0	1
snd	0.04	D	1	50	1
rcv	0.04	A	1	0	51
snd	0.04	D	51	50	1
rcv	0.04	A	1	0	101
drop	0.04	D	101	50	1
snd/RXT	1.11	D	101	50	1
rcv	1.12	A	1	0	151
snd	1.12	D	151	50	1
rcv	1.12	A	1	0	201
snd	1.12	D	201	50	1
rcv	1.12	A	1	0	251
snd	1.12	D	251	50	1
rcv	1.12	A	1	0	301
snd	1.12	D	301	50	1
rcv	1.13	A	1	0	351
snd	1.13	D	351	50	1
...					
drop	1455.37	D	307851	50	1
snd/RXT	1455.60	D	307851	50	1
rcv	1455.60	A	1	0	307901
snd	1455.60	D	307901	50	1
rcv	1455.60	A	1	0	307951
snd	1455.60	D	307951	50	1
rcv	1455.60	A	1	0	308001
drop	1455.60	D	308001	50	1
snd/RXT	1455.76	D	308001	50	1
rcv	1455.76	A	1	0	308051



snd	1455.76	D	308051 50	1
rcv	1455.76	A	1 0	308101
drop	1455.76	D	308101 50	1
snd/RXT	1455.91	D	308101 50	1
rcv	1455.91	A	1 0	308151
drop	1455.91	D	308151 50	1
snd/RXT	1456.06	D	308151 50	1
rcv	1456.06	A	1 0	308201
snd	1456.06	D	308201 3	1
rcv	1456.06	A	1 0	308204
snd	1456.06	F	308204 0	2
rcv	1456.06	A	1 0	308205
rcv	1456.06	F	308204 0	2
snd	1456.06	A	308205 0	2

Size of the file (in Bytes) 308203

Segments transmitted (including drop & RXT) 9579

Number of Segments handled by PLD 9575

Number of Segments dropped 3024

Number of Segments Corrupted 0

Number of Segments Re-ordered 0

Number of Segments Duplicated 0

Number of Segments Delayed 623

Number of Retransmissions due to TIMEOUT 3410

Number of DUP ACKS received 386

ii) gamma = 4

Receiver\_log

rcv	0.00	S	0	0	0
snd	0.03	S	1	0	1
rcv	0.03	A	1	0	1
rcv	0.03	D	1	50	1
snd	0.03	A	1	0	51
rcv	0.04	D	51	50	1
snd	0.04	A	1	0	101
rcv	1.79	D	101	50	1
snd	1.79	A	1	0	151
rcv	1.79	D	151	50	1
snd	1.80	A	1	0	201
rcv	1.80	D	201	50	1
snd	1.80	A	1	0	251
rcv	1.80	D	251	50	1
snd	1.80	A	1	0	301
rcv	1.80	D	301	50	1
snd	1.80	A	1	0	351
rcv	1.81	D	351	50	1
snd	1.81	A	1	0	401

rcv	1.81	D	401	50	1	
...						
snd	2558.05	A	1	0		307751
rcv	2558.51	D	307751	50	1	
snd	2558.51	A	1	0		307801
rcv	2558.51	D	307801	50	1	
snd	2558.51	A	1	0		307851
rcv	2558.91	D	307851	50	1	
snd	2558.91	A	1	0		307901
rcv	2558.91	D	307901	50	1	
snd	2558.91	A	1	0		307951
rcv	2558.91	D	307951	50	1	
snd	2558.91	A	1	0		308001
rcv	2559.21	D	308001	50	1	
snd	2559.21	A	1	0		308051
rcv	2559.21	D	308051	50	1	
snd	2559.21	A	1	0		308101
rcv	2559.48	D	308101	50	1	
snd	2559.48	A	1	0		308151
rcv	2559.76	D	308151	50	1	
snd	2559.76	A	1	0		308201
rcv	2559.76	D	308201	3	1	
snd	2559.76	A	1	0		308204
rcv	2559.76	F	308204	0	2	
snd	2559.76	F	1	0		308205
snd	2559.76	A	1	0		308205
rcv	2559.76	A	308205	0	2	

=====

Amount of data received (bytes) 318953

Total Segments Received 6384

Data segments received 6380

Data segments with Bit Errors 0

Duplicate data segments received 215

Duplicate ACKs sent 215

=====

Sender\_log

snd	0.01	S	0	0	0	
rcv	0.03	SA	0	0	1	
snd	0.04	A	1	0	1	
snd	0.04	D	1	50	1	
rcv	0.04	A	1	0	51	
snd	0.04	D	51	50	1	
rcv	0.04	A	1	0	101	
drop	0.04	D	101	50	1	
snd/RXT	1.80	D	101	50	1	
rcv	1.80	A	1	0	151	
snd	1.80	D	151	50	1	

rcv	1.80	A	1	0	201
snd	1.80	D	201	50	1
rcv	1.80	A	1	0	251
snd	1.81	D	251	50	1
rcv	1.81	A	1	0	301
snd	1.81	D	301	50	1
rcv	1.81	A	1	0	351
snd	1.81	D	351	50	1
rcv	1.81	A	1	0	401
...					
drop	2558.51		D	307851	50 1
snd/RXT	2558.91		D	307851	50 1
rcv	2558.91		A	1	0 307901
snd	2558.91		D	307901	50 1
rcv	2558.91		A	1	0 307951
snd	2558.91		D	307951	50 1
rcv	2558.91		A	1	0 308001
drop	2558.91		D	308001	50 1
snd/RXT	2559.22		D	308001	50 1
rcv	2559.22		A	1	0 308051
snd	2559.22		D	308051	50 1
rcv	2559.22		A	1	0 308101
drop	2559.22		D	308101	50 1
snd/RXT	2559.49		D	308101	50 1
rcv	2559.49		A	1	0 308151
drop	2559.49		D	308151	50 1
snd/RXT	2559.77		D	308151	50 1
rcv	2559.77		A	1	0 308201
snd	2559.77		D	308201	3 1
rcv	2559.77		A	1	0 308204
snd	2559.77		F	308204	0 2
rcv	2559.77		A	1	0 308205
rcv	2559.77		F	308204	0 2
snd	2559.77		A	308205	0 2

=====

Size of the file (in Bytes) 308203  
 Segments transmitted (including drop & RXT) 9408  
 Number of Segments handled by PLD 9404  
 Number of Segments dropped 3024  
 Number of Segments Corrupted 0  
 Number of Segments Re-ordered 0  
 Number of Segments Duplicated 0  
 Number of Segments Delayed 623  
 Number of Retransmissions due to TIMEOUT 3239  
 Number of DUP ACKS received 215

=====

iii)

rcv	0.00	S	0	0	0
snd	0.03	S	1	0	1
rcv	0.03	A	1	0	1
rcv	0.03	D	1	50	1
snd	0.03	A	1	0	51
rcv	0.04	D	51	50	1
snd	0.04	A	1	0	101
rcv	2.48	D	101	50	1
snd	2.48	A	1	0	151
rcv	2.48	D	151	50	1
snd	2.48	A	1	0	201
rcv	2.48	D	201	50	1
snd	2.49	A	1	0	251
rcv	2.49	D	251	50	1
snd	2.49	A	1	0	301
rcv	2.49	D	301	50	1
snd	2.49	A	1	0	351
rcv	2.49	D	351	50	1
snd	2.49	A	1	0	401
rcv	2.50	D	401	50	1
...					
rcv	3921.30	D	307801	50	1
snd	3921.30	A	1	0	307851
rcv	3922.65	D	307851	50	1
snd	3922.65	A	1	0	307901
rcv	3922.65	D	307901	50	1
snd	3922.65	A	1	0	307951
rcv	3922.65	D	307951	50	1
snd	3922.65	A	1	0	308001
rcv	3923.70	D	308001	50	1
snd	3923.70	A	1	0	308051
rcv	3923.70	D	308051	50	1
snd	3923.70	A	1	0	308101
rcv	3924.62	D	308101	50	1
snd	3924.62	A	1	0	308151
rcv	3925.56	D	308151	50	1
snd	3925.56	A	1	0	308201
rcv	3925.56	D	308201	3	1
snd	3925.56	A	1	0	308204
rcv	3925.56	F	308204	0	2
snd	3925.56	F	1	0	308205
snd	3925.56	A	1	0	308205
rcv	3925.56	A	308205	0	2

=====

Amount of data received (bytes) 314353

Total Segments Received 6292

Data segments received 6288

Data segments with Bit Errors 0

Duplicate data segments received 123

Duplicate ACKs sent 123

=====

Sender\_log

snd	0.01	S	0	0	0	
rcv	0.04	SA	0	0	1	
snd	0.04	A	1	0	1	
snd	0.04	D	1	50	1	
rcv	0.04	A	1	0	51	
snd	0.04	D	51	50	1	
rcv	0.04	A	1	0	101	
drop	0.04	D	101	50	1	
snd/RXT	2.49	D	101	50	1	
rcv	2.49	A	1	0	151	
snd	2.49	D	151	50	1	
rcv	2.49	A	1	0	201	
snd	2.49	D	201	50	1	
rcv	2.49	A	1	0	251	
snd	2.49	D	251	50	1	
rcv	2.49	A	1	0	301	
snd	2.50	D	301	50	1	
rcv	2.50	A	1	0	351	
snd	2.50	D	351	50	1	
rcv	2.50	A	1	0	401	
...						
rcv	3918.26	A	1	0	307701	
drop	3918.26	D	307701	50	1	
snd/RXT	3919.78	D	307701	50	1	
rcv	3919.78	A	1	0	307751	
drop	3919.78	D	307751	50	1	
snd/RXT	3921.31	D	307751	50	1	
rcv	3921.31	A	1	0	307801	
snd	3921.31	D	307801	50	1	
rcv	3921.31	A	1	0	307851	
drop	3921.31	D	307851	50	1	
snd/RXT	3922.66	D	307851	50	1	
rcv	3922.66	A	1	0	307901	
snd	3922.66	D	307901	50	1	
rcv	3922.66	A	1	0	307951	
snd	3922.66	D	307951	50	1	
rcv	3922.66	A	1	0	308001	
drop	3922.66	D	308001	50	1	
snd/RXT	3923.71	D	308001	50	1	
rcv	3923.71	A	1	0	308051	
snd	3923.71	D	308051	50	1	
rcv	3923.71	A	1	0	308101	
drop	3923.71	D	308101	50	1	

snd/RXT	3924.62	D	308101	50	1
rcv	3924.62	A	1	0	308151
drop	3924.63	D	308151	50	1
snd/RXT	3925.56	D	308151	50	1
rcv	3925.56	A	1	0	308201
snd	3925.56	D	308201	3	1
rcv	3925.56	A	1	0	308204
snd	3925.56	F	308204	0	2
rcv	3925.56	A	1	0	308205
rcv	3925.56	F	308204	0	2
snd	3925.56	A	308205	0	2

=====

Size of the file (in Bytes) 308203

Segments transmitted (including drop & RXT) 9316

Number of Segments handled by PLD 9312

Number of Segments dropped 3024

Number of Segments Corrupted 0

Number of Segments Re-ordered 0

Number of Segments Duplicated 0

Number of Segments Delayed 623

Number of Retransmissions due to TIMEOUT 3147

Number of DUP ACKS received 123

=====

c) Sender\_log.txt

First 20 lines

Sender_log.txt ✕						
1	snd	0.01	S	0	0	0
2	rcv	0.02	SA	0	0	1
3	snd	0.02	A	1	0	1
4	snd	0.03	D	1	50	1
5	rcv	0.03	A	1	0	51
6	snd	0.03	D	51	50	1
7	rcv	0.03	A	1	0	101
8	snd	0.03	D	101	50	1
9	rcv	0.03	A	1	0	151
10	snd	0.03	D	151	50	1
11	rcv	0.03	A	1	0	201
12	snd	0.04	D	201	50	1
13	rcv	0.04	A	1	0	251
14	snd	0.04	D	251	50	1
15	rcv	0.04	A	1	0	301
16	snd	0.04	D	301	50	1
17	rcv	0.04	A	1	0	351
18	snd/corr	0.04	D	351	50	1
19	snd/RXT	1.41	D	351	50	1

Last lines when program was forcefully terminated

684983	snd/RXT	171.54	D	769201	50	1
684984	rcv/DA	171.54	A	1	0	769251
684985	snd/RXT	171.54	D	769201	50	1
684986	rcv/DA	171.54	A	1	0	769251
684987	snd/RXT	171.54	D	769201	50	1
684988	rcv/DA	171.54	A	1	0	769251
684989	snd/RXT	171.54	D	769201	50	1
684990	rcv/DA	171.54	A	1	0	769251
684991	snd/RXT	171.54	D	769201	50	1
684992	rcv/DA	171.54	A	1	0	769251
684993	snd/RXT	171.54	D	769201	50	1
684994	rcv/DA	171.54	A	1	0	769251
684995	snd/RXT	171.54	D	769201	50	1
684996	rcv/DA	171.54	A	1	0	769251
684997	snd/RXT	171.54	D	769201	50	1
684998	rcv/DA	171.54	A	1	0	769251
684999	snd/RXT	171.54	D	769201	50	1
685000	rcv/DA	171.54	A	1	0	769251
685001	snd/RXT	171.54	D	769201	50	1
685002	rcv/DA	171.54	A	1	0	769251

Receiver.txt

First 20 lines