Introduction:
In this laboratory programming assignment, we will be writing the sending and receiving transport-level code in order to implement a simple reliable data transfer protocol using Go-Back-N and a packet version of TCP. We will be testing the network against different possible scenarios, packet loss, packet corruption, acknowledgement loss, acknowledgment corruption and different window sizes. We will compare and contrast how Go-Back-N performs against TCP when dealing with such scenarios.

Protocol descriptions and explanations:

***Go-Back-N*:** A transport layer protocol in which a sender sends segments to a server according to a specified window size N. The window represents the total amount of segments that can be sent at a time. After a segment is sent, the sender will expect an acknowledgment (ACK) from the server for each segment sent. The window will only move on to new segment(s) when an ACK for the "base"[1] segment is received. If no ACKs are received after a certain amount of time the sender will simply resend all the packets in its current window. Once all ACKs for the current segments in the window are received, the window will move forward to obtain new segments. Go-Back-N also implements something called "cumulative ACKs" a process in which the receiver will send the ACK for a group of segments in contrast to ACKs for a single segment, this lets the sender know that all segments before that ACK number have been received.

***TCP*:** A transport layer protocol in which a sender sends segments to a server according to a specified window size N. The window represents the total amount of segments that can be sent at a time. After a segment is sent, the sender will expect an acknowledgment (ACK) from the server for each segment sent. The receiver will buffer out of order segments and return ACKs accordingly. The window will only move on to new segment(s) when an ACK for the "base"[2] segment is received. If no ACKs are received after a certain amount of time the sender will only resend the first segment that have not yet been acknowledged. In addition, there is a fast retransmit function that automatically resends the segment if 3 ACKs for the previous packet has been received. TCP also implements something called "cumulative ACKs" a process in which the receiver will send the ACK for a group of segments in contrast to ACKs for a single segment, this lets the sender know that all segments before that ACK number have been received.

---

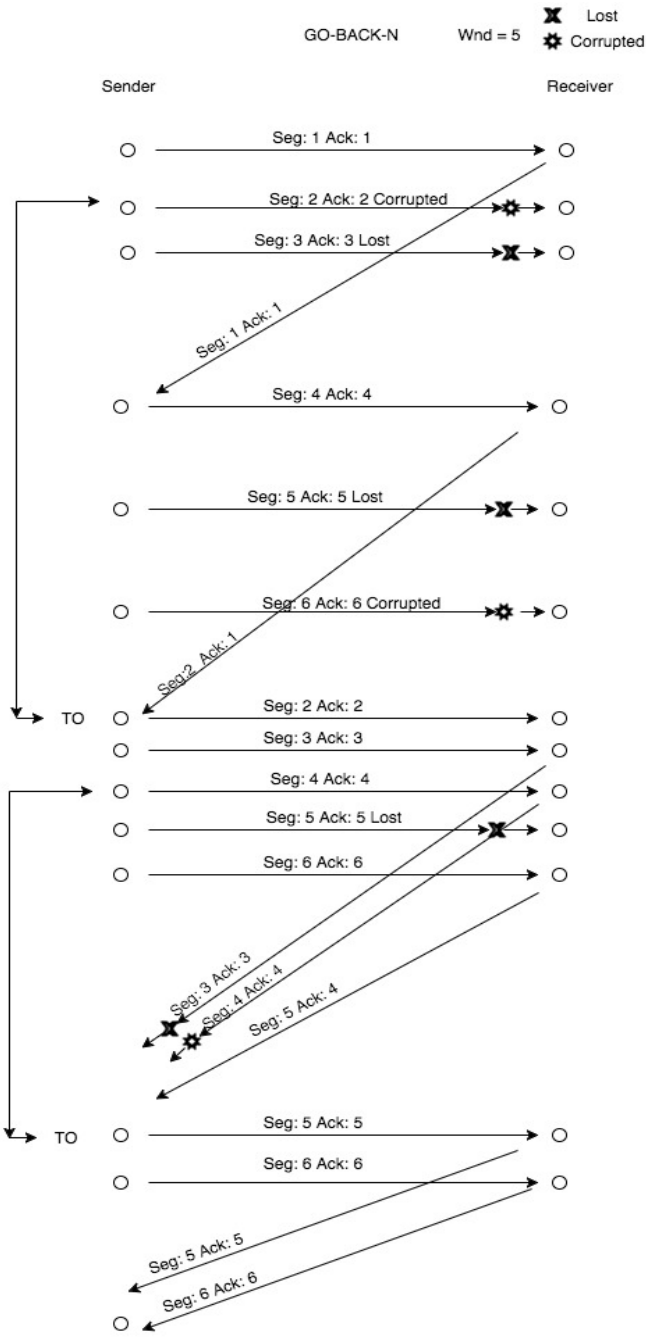
1 The first segment in the window to be sent

2 The first segment in the window to be sent

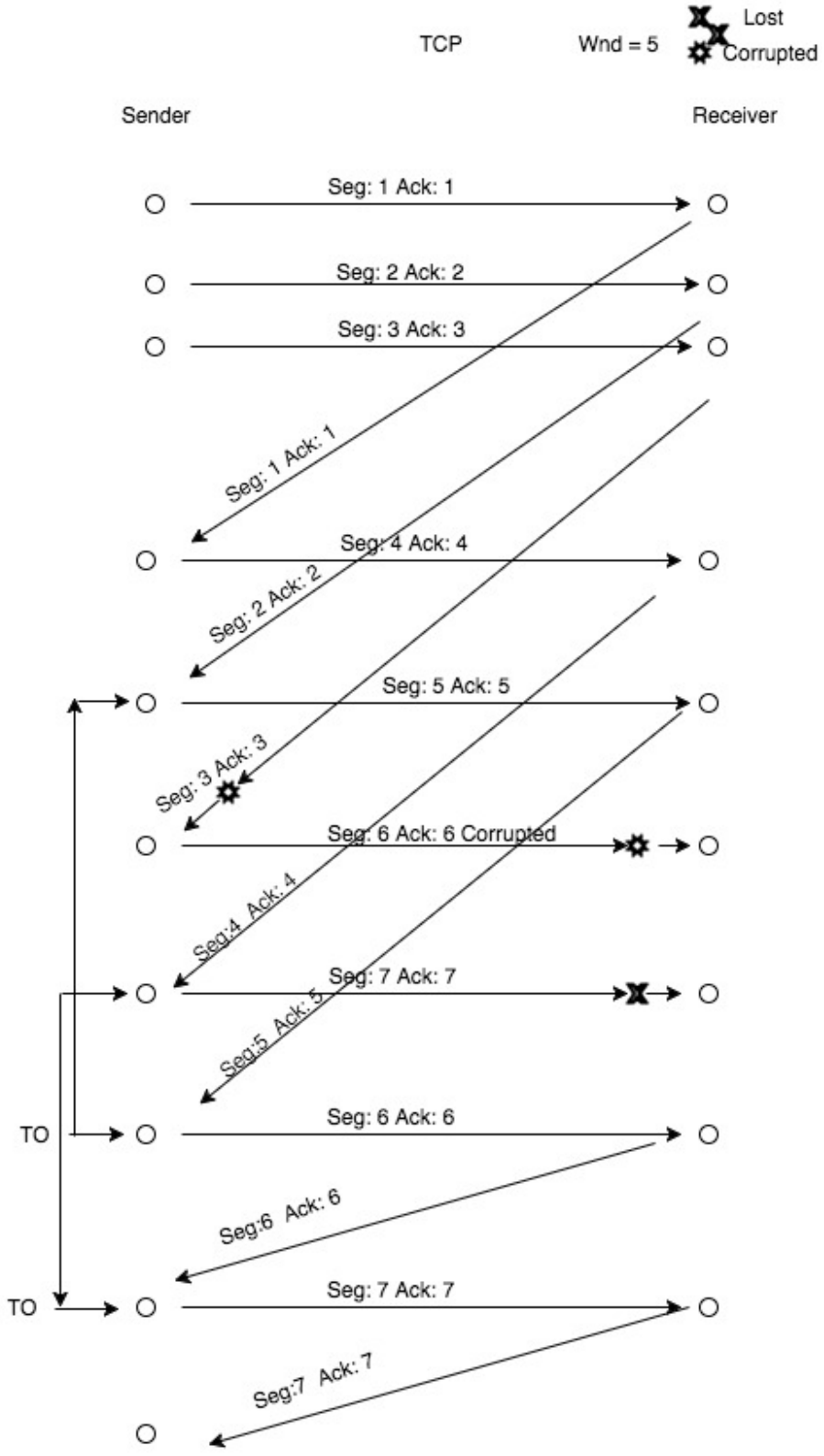
Code Design and Data Structures:
A Hash map was used to simulate the window because of its fast look up times and organizational efficiency. A Queue was used to buffer waiting segments in the sender. An array list was used to buffer out of order segments in the receiver.

Correctness Results:* (See end of pdf for terminal code print outs)

***Go-Back-N*:**

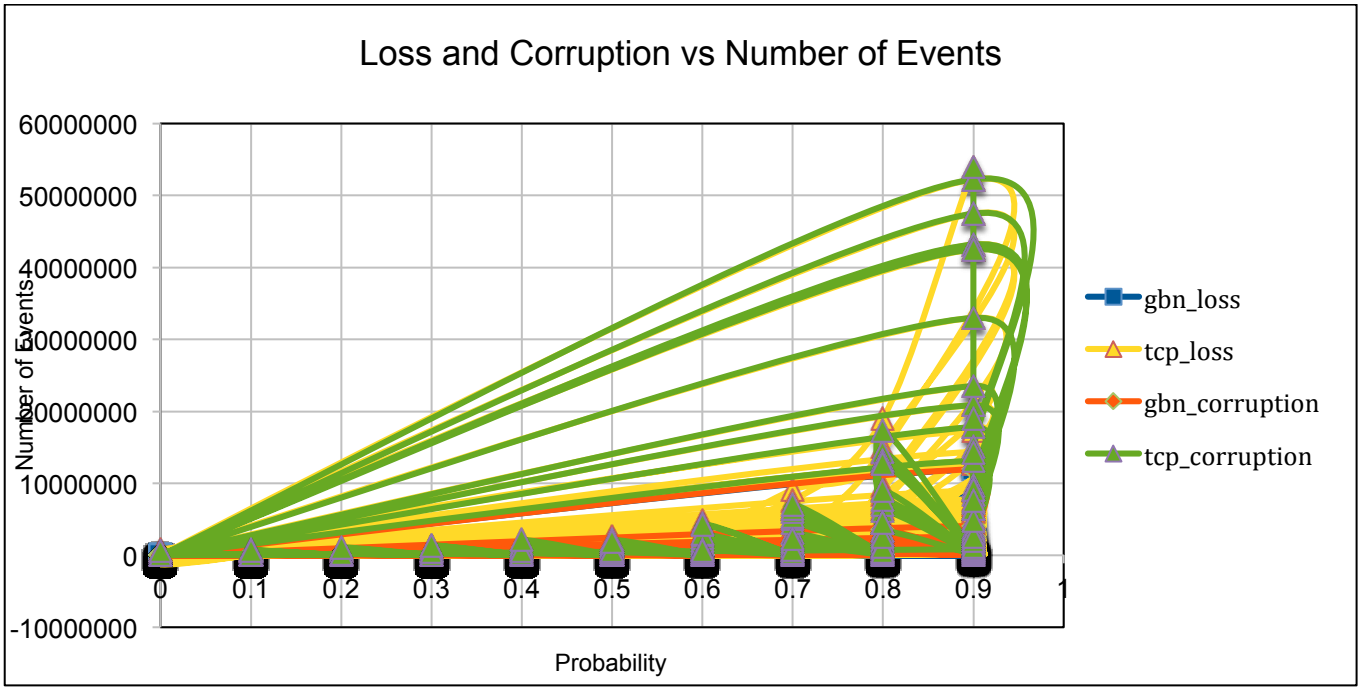


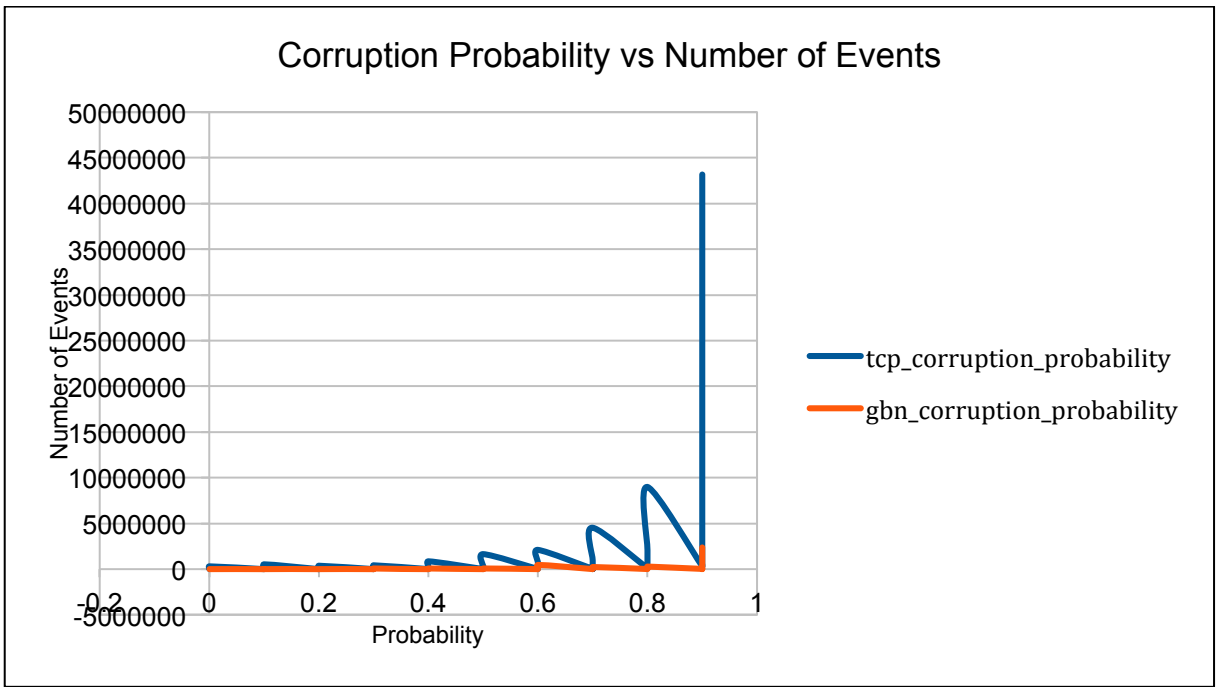
***TCP*:**

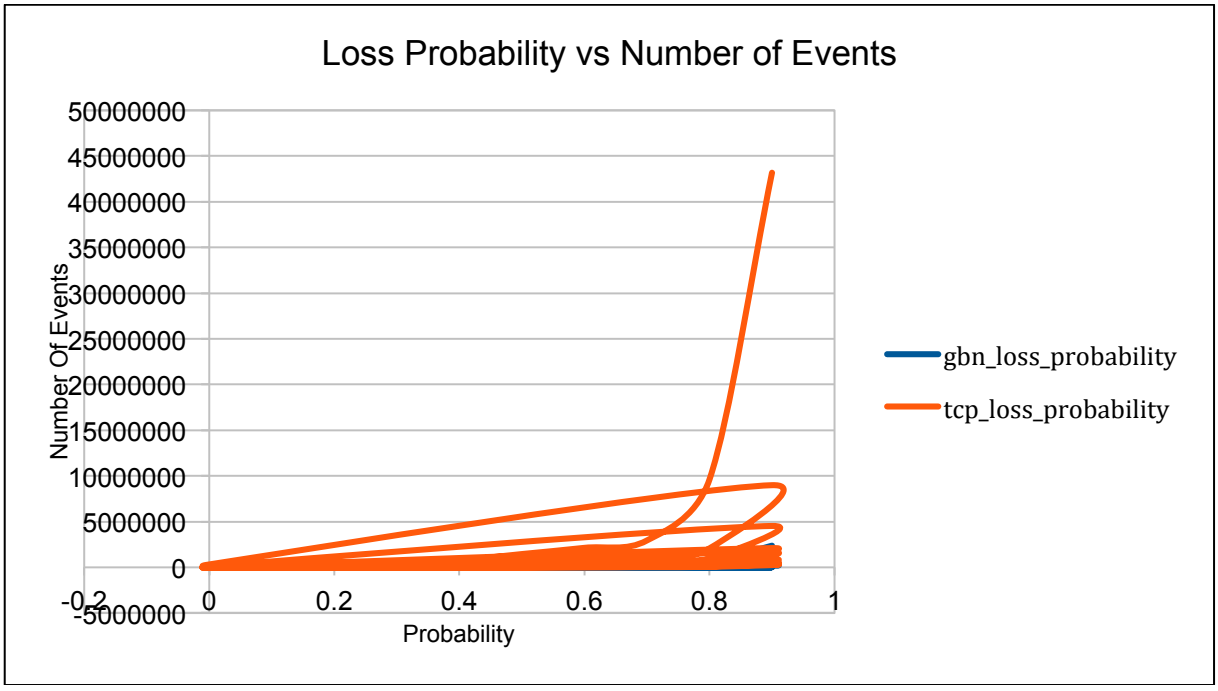

TCP
Wnd = 5
Lost
Corrupted
Sender
Receiver
Seg: 1 Ack: 1
Seg: 2 Ack: 2
Seg: 3 Ack: 3
Seg: 1 Ack: 1
Seg: 4 Ack: 4
Seg: 2 Ack: 2
Seg: 5 Ack: 5
Seg: 3 Ack: 3
Seg: 6 Ack: 6 Corrupted
Seg:4 Ack: 4
Seg: 7 Ack: 7
Seg:5 Ack: 5
TO
Seg: 6 Ack: 6
Seg:6 Ack: 6
TO
Seg: 7 Ack: 7
Seg:7 Ack: 7

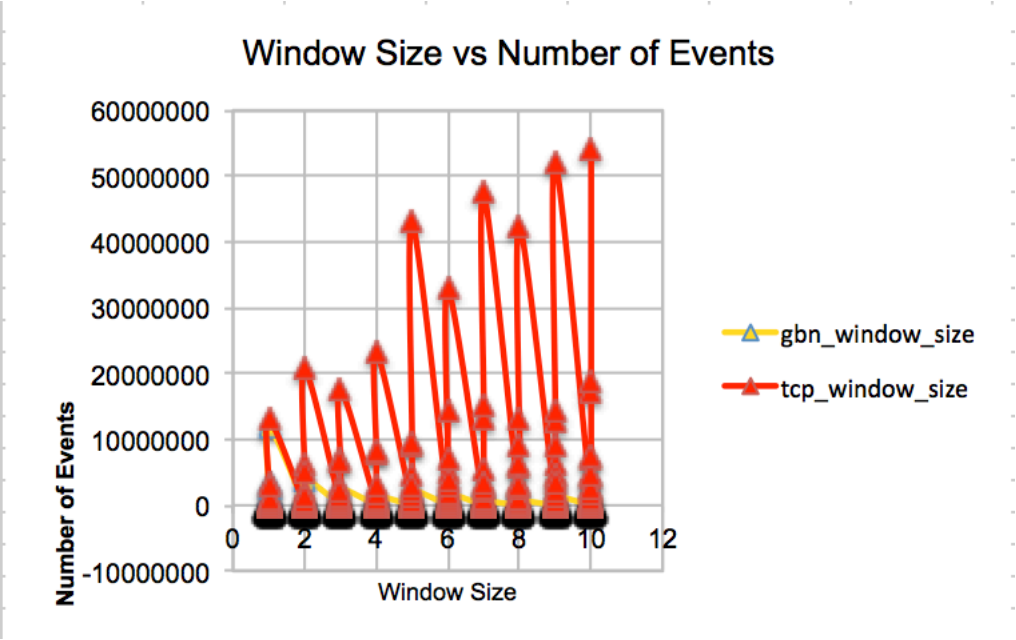
Analysis: * (See looper.sh and data.csv for further details)

For the analysis a shell script was written to run the NetworkSimulator.java program multiple times and store the data it generated in a .csv file that could easily opened by Microsoft Excel. The script looped through 4 variables: loss, corruption, window size, and protocol type. For both loss and corruption it went from 0.0 to 0.9 incrementing by 0.1 respectively, and the window size went from 1 to 10 incrementing by 1. In total the program was run 2000 times to capture every combination of variables for both TCP and GBN. For each of the conditions the number of events required for completion as well as the total time the program took to complete was recorded. In the charts below the number of events is shown on the Y-Axis and when using time instead of events the trends will be the same. `




Corruption Probability vs Number of Events
50000000
45000000
40000000
35000000
30000000
25000000
20000000
15000000
10000000
5000000
0
-5000000
Number of Events
-0.2
0
0.2
0.4
0.6
0.8
1
Probability
tcp_corruption_probability
gbn_corruption_probability


Loss Probability vs Number of Events
50000000
45000000
40000000
35000000
30000000
25000000
20000000
15000000
10000000
5000000
0
-5000000
Number Of Events
-0.2
0
0.2
0.4
0.6
0.8
1
Probability
gbn_loss_probability
tcp_loss_probability



Conclusion:
After carefully looking through our data and analysis we can safely say that GBN is far more efficient than TCP. This is likely because GBN will do better when there is more corruption and loss with a window size that is not overly large. We predict that with window sizes that mirror the internet more closely TCP would likely surpass GBN at some point, but for our simulation as corruption and loss increase GBN performs the best.

At corruption and loss levels lower than 0.5, the two protocols are essentially equivalent in terms of the time they take to complete.

Delegation:
Manny:
Sender's transport:

- sendMessage(Message msg
- receiveMessage(Packet pkt)
- timerExpired()
- initialize()

Other:

- Report
- Correctness
- GBN protocols

Conrad:
Receivers transport:

- receiveMessage(Packet pkt)
- initialize()

Packet class:

- setChecksum()
- isCorrupt()

**** Go-Back-N*:**

```
Manny Orders > java NetworkSimulator test2.txt 10 .1 .1 5 0 3
'createSendEvent'-Inserting future arrive event at 0 with avg RTT
time: 21
'createSendEvent'-Inserting future arrive event at 21 with avg
RTT time: 23
YAY IN 1
ADDING to map: 1




Packet seq: 1 ack: 1 sent:)
'createArriveEvent'-Inserting future arrive event at 21 with
calculated avg RTT: 26 to: Receiver
startTimer'-Inserting future timer event at time: 21 plus
increment = 100
Message sent from sender to receiver at time 21
'createSendEvent'-Inserting future arrive event at 23 with avg
RTT time: 24
YAY IN 1
ADDING to map: 2
Packet seq: 2 ack: 2 CORRUPTED
Packet seq: 2 ack: 2 sent:)
'createArriveEvent'-Inserting future arrive event at 23 with
calculated avg RTT: 30 to: Receiver
Message sent from sender to receiver at time 23
'createSendEvent'-Inserting future arrive event at 24 with avg
RTT time: 26
YAY IN 1
ADDING to map: 3
Packet seq: 3 ack: 3 LOST
Message sent from sender to receiver at time 24
YAY IN 3 AT RECEIVER


Message arriving from sender to receiver at time 26
from receiver:Article 1  Right to Equality
Packet seq: 1 ack: 1 sent:)
```

```
'createArriveEvent'-Inserting future arrive event at 26 with
calculated avg RTT: 36 to: Sender
'createSendEvent'-Inserting future arrive event at 26 with avg
RTT time: 41
YAY IN 1
ADDING to map: 4
Packet seq: 4 ack: 4 sent:)
'createArriveEvent'-Inserting future arrive event at 26 with
calculated avg RTT: 42 to: Receiver
Message sent from sender to receiver at time 26
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 30
Sorry msg is from sender is corrupted
YAY IN 2 AT Sender
Message arriving from receiver to sender at time 36
GOT: 1 true 4
REmoving map: 1
startTimer'-Inserting future timer event at time: 36 plus
increment = 228
'createSendEvent'-Inserting future arrive event at 41 with avg
RTT time: 47
YAY IN 1
ADDING to map: 5
Packet seq: 5 ack: 5 LOST
Message sent from sender to receiver at time 41
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 42
from receiver:Article 4   Freedom from Slavery
Packet seq: 2 ack: 1 sent:)
'createArriveEvent'-Inserting future arrive event at 42 with
calculated avg RTT: 49 to: Sender
'createSendEvent'-Inserting future arrive event at 47 with avg
RTT time: 59
YAY IN 1
ADDING to map: 6
Packet seq: 6 ack: 6 CORRUPTED
Packet seq: 6 ack: 6 sent:)
'createArriveEvent'-Inserting future arrive event at 47 with
calculated avg RTT: 58 to: Receiver
Message sent from sender to receiver at time 47
YAY IN 2 AT Sender
Message arriving from receiver to sender at time 49
GOT: 1 false 5
seqnum != to base base: 2next seq: 7
wait for timeout
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 58
Sorry msg is from sender is corrupted
YAY IN 1
Buffering msg, window is FULL
Message sent from sender to receiver at time 59
YAY IN 5 IM A TIMER
```

```
YAY IN 4 IM A TIMER
Timer expired at time 264
base: 2
sequence pkt: 2
Resending pkt: 2
Packet seq: 2 ack: 2 sent:)
'createArriveEvent'-Inserting future arrive event at 264 with
calculated avg RTT: 272 to: Receiver
base: 2
sequence pkt: 2
Resending pkt: 3
Packet seq: 3 ack: 3 sent:)
'createArriveEvent'-Inserting future arrive event at 264 with
calculated avg RTT: 273 to: Receiver
base: 2
sequence pkt: 2
Resending pkt: 4
Packet seq: 4 ack: 4 sent:)
'createArriveEvent'-Inserting future arrive event at 264 with
calculated avg RTT: 277 to: Receiver
base: 2
sequence pkt: 2
Resending pkt: 5
Packet seq: 5 ack: 5 LOST
base: 2
sequence pkt: 2
Resending pkt: 6
Packet seq: 6 ack: 6 sent:)
'createArriveEvent'-Inserting future arrive event at 264 with
calculated avg RTT: 278 to: Receiver
startTimer'-Inserting future timer event at time: 264 plus
increment = 238
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 272
from receiver:Article 2  Freedom from Discrimination
Packet seq: 2 ack: 2 sent:)
'createArriveEvent'-Inserting future arrive event at 272 with
calculated avg RTT: 280 to: Sender
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 273
from receiver:Article 3  Right to Life, Liberty, Personal
Security
Packet seq: 3 ack: 3 LOST
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 277
from receiver:Article 4  Freedom from Slavery
Packet seq: 4 ack: 4 CORRUPTED
Packet seq: 4 ack: 4 sent:)
'createArriveEvent'-Inserting future arrive event at 277 with
calculated avg RTT: 283 to: Sender
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 278
```

```
from receiver:Article 6  Right to Recognition as a Person before
the Law
Packet seq: 5 ack: 4 sent:)
'createArriveEvent'-Inserting future arrive event at 278 with
calculated avg RTT: 291 to: Sender
YAY IN 2 AT Sender
Message arriving from receiver to sender at time 280
GOT: 2 true 5
REmoving map: 2
ADDING to map: 2
Packet seq: 2 ack: 2 sent:)
'createArriveEvent'-Inserting future arrive event at 280 with
calculated avg RTT: 292 to: Receiver
startTimer'-Inserting future timer event at time: 280 plus
increment = 235
YAY IN 2 AT Sender
Message arriving from receiver to sender at time 283
Sorry pkt from receiver is corrupted 4
YAY IN 2 AT Sender
Message arriving from receiver to sender at time 291
GOT: 4 true 5
REmoving map: 2
REmoving map: 3
REmoving map: 4
startTimer'-Inserting future timer event at time: 291 plus
increment = 235
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 292
from receiver:Article 7  Right to Equality before the Law
Packet seq: 5 ack: 4 sent:)
'createArriveEvent'-Inserting future arrive event at 292 with
calculated avg RTT: 297 to: Sender
YAY IN 2 AT Sender
Message arriving from receiver to sender at time 297
GOT: 4 false 2
seqnum != to base base: 5next seq: 3
wait for timeout
YAY IN 5 IM A TIMER
YAY IN 5 IM A TIMER
YAY IN 4 IM A TIMER
Timer expired at time 526
base: 5
sequence pkt: 5
Resending pkt: 5
Packet seq: 5 ack: 5 sent:)
'createArriveEvent'-Inserting future arrive event at 526 with
calculated avg RTT: 534 to: Receiver
base: 5
sequence pkt: 5
Resending pkt: 6
Packet seq: 6 ack: 6 sent:)
'createArriveEvent'-Inserting future arrive event at 526 with
```

```
calculated avg RTT: 537 to: Receiver
startTimer'-Inserting future timer event at time: 526 plus
increment = 235
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 534
from receiver:Article 5   Freedom from Torture and Degrading
Treatment
Packet seq: 5 ack: 5 sent:)
'createArriveEvent'-Inserting future arrive event at 534 with
calculated avg RTT: 540 to: Sender
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 537
from receiver:Article 6   Right to Recognition as a Person before
the Law
Packet seq: 6 ack: 6 sent:)
'createArriveEvent'-Inserting future arrive event at 537 with
calculated avg RTT: 547 to: Sender
YAY IN 2 AT Sender
Message arriving from receiver to sender at time 540
GOT: 5 true 2
REmoving map: 5
startTimer'-Inserting future timer event at time: 540 plus
increment = 235
YAY IN 2 AT Sender
Message arriving from receiver to sender at time 547
GOT: 6 true 1
REmoving map: 6
startTimer'-Inserting future timer event at time: 547 plus
increment = 235
YAY IN 5 IM A TIMER
YAY IN 5 IM A TIMER
YAY IN 4 IM A TIMER
Timer expired at time 782
```

***TCP*:**

```
Manny Orders > java NetworkSimulator test2.txt 10 .1 .1 5 1 3
'createSendEvent'-Inserting future arrive event at 0 with avg RTT
time: 1
'createSendEvent'-Inserting future arrive event at 1 with avg RTT
time: 9
YAY IN 1
ADDING to map: 1
Packet seq: 1 ack: 1 sent:)
'createArriveEvent'-Inserting future arrive event at 1 with
calculated avg RTT: 4 to: Receiver
startTimer'-Inserting future timer event at time: 1 plus
increment = 100
Message sent from sender to receiver at time 1
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 4
from receiver:Article 1   Right to Equality
Packet seq: 1 ack: 1 sent:)
```

```
'createArriveEvent'-Inserting future arrive event at 4 with
calculated avg RTT: 10 to: Sender
'createSendEvent'-Inserting future arrive event at 9 with avg RTT
time: 14
YAY IN 1
ADDING to map: 2
Packet seq: 2 ack: 2 sent:)
'createArriveEvent'-Inserting future arrive event at 9 with
calculated avg RTT: 14 to: Receiver
Message sent from sender to receiver at time 9
YAY IN 2 AT Sender
Message arriving from receiver to sender at time 10
GOT: 1 true 2
Removing map: 1
startTimer'-Inserting future timer event at time: 10 plus
increment = 163
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 14
from receiver:Article 2  Freedom from Discrimination
Packet seq: 2 ack: 2 sent:)
'createArriveEvent'-Inserting future arrive event at 14 with
calculated avg RTT: 22 to: Sender
'createSendEvent'-Inserting future arrive event at 14 with avg
RTT time: 53
YAY IN 1
ADDING to map: 3
Packet seq: 3 ack: 3 sent:)
'createArriveEvent'-Inserting future arrive event at 14 with
calculated avg RTT: 28 to: Receiver
Message sent from sender to receiver at time 14
YAY IN 2 AT Sender
Message arriving from receiver to sender at time 22
GOT: 2 true 2
Removing map: 2
startTimer'-Inserting future timer event at time: 22 plus
increment = 204
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 28
from receiver:Article 3  Right to Life, Liberty, Personal
Security
Packet seq: 3 ack: 3 CORRUPTED
Packet seq: 3 ack: 3 sent:)
'createArriveEvent'-Inserting future arrive event at 28 with
calculated avg RTT: 36 to: Sender
YAY IN 2 AT Sender
Message arriving from receiver to sender at time 36
Sorry pkt from receiver is corrupted 3
'createSendEvent'-Inserting future arrive event at 53 with avg
RTT time: 67
YAY IN 1
ADDING to map: 4
Packet seq: 4 ack: 4 sent:)
```

```
'createArriveEvent'-Inserting future arrive event at 53 with
calculated avg RTT: 62 to: Receiver
Message sent from sender to receiver at time 53
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 62
from receiver:Article 4  Freedom from Slavery
Packet seq: 4 ack: 4 sent:)
'createArriveEvent'-Inserting future arrive event at 62 with
calculated avg RTT: 71 to: Sender
'createSendEvent'-Inserting future arrive event at 67 with avg
RTT time: 68
YAY IN 1
ADDING to map: 5
Packet seq: 5 ack: 5 sent:)
'createArriveEvent'-Inserting future arrive event at 67 with
calculated avg RTT: 76 to: Receiver
Message sent from sender to receiver at time 67
'createSendEvent'-Inserting future arrive event at 68 with avg
RTT time: 74
YAY IN 1
ADDING to map: 6
Packet seq: 6 ack: 6 CORRUPTED
Packet seq: 7 ack: 6 sent:)
'createArriveEvent'-Inserting future arrive event at 68 with
calculated avg RTT: 78 to: Receiver
Message sent from sender to receiver at time 68
YAY IN 2 AT Sender
Message arriving from receiver to sender at time 71
GOT: 4 true 4
Removing map: 3
Removing map: 4
startTimer'-Inserting future timer event at time: 71 plus
increment = 241
YAY IN 1
ADDING to map: 7
Packet seq: 7 ack: 7 LOST
Message sent from sender to receiver at time 74
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 76
from receiver:Article 5  Freedom from Torture and Degrading
Treatment
Packet seq: 5 ack: 5 sent:)
'createArriveEvent'-Inserting future arrive event at 76 with
calculated avg RTT: 87 to: Sender
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 78
Sorry msg is from sender is corrupted
YAY IN 2 AT Sender
Message arriving from receiver to sender at time 87
GOT: 5 true 3
Removing map: 5
startTimer'-Inserting future timer event at time: 87 plus
```

```
increment = 238
YAY IN 5 IM A TIMER
YAY IN 5 IM A TIMER
YAY IN 5 IM A TIMER
YAY IN 5 IM A TIMER
YAY IN 4 IM A TIMER
Timer expired at time 325
Resending pkt: 6
base: 6
Packet seq: 6 ack: 6 sent:)
'createArriveEvent'-Inserting future arrive event at 325 with
calculated avg RTT: 328 to: Receiver
startTimer'-Inserting future timer event at time: 325 plus
increment = 238
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 328
from receiver:Article 6  Right to Recognition as a Person before
the Law
Packet seq: 6 ack: 6 sent:)
'createArriveEvent'-Inserting future arrive event at 328 with
calculated avg RTT: 331 to: Sender
YAY IN 2 AT Sender
Message arriving from receiver to sender at time 331
GOT: 6 true 2
Removing map: 6
startTimer'-Inserting future timer event at time: 331 plus
increment = 238
YAY IN 5 IM A TIMER
YAY IN 4 IM A TIMER
Timer expired at time 569
Resending pkt: 7
base: 7
Packet seq: 7 ack: 7 sent:)
'createArriveEvent'-Inserting future arrive event at 569 with
calculated avg RTT: 575 to: Receiver
startTimer'-Inserting future timer event at time: 569 plus
increment = 238
YAY IN 3 AT RECEIVER
Message arriving from sender to receiver at time 575
from receiver:Article 7  Right to Equality before the Law
Packet seq: 7 ack: 7 sent:)
'createArriveEvent'-Inserting future arrive event at 575 with
calculated avg RTT: 576 to: Sender
YAY IN 2 AT Sender
Message arriving from receiver to sender at time 576
GOT: 7 true 1
Removing map: 7
startTimer'-Inserting future timer event at time: 576 plus
increment = 238
YAY IN 5 IM A TIMER
YAY IN 4 IM A TIMER
Timer expired at time 814
```