Computer Networks

Project-1

Nouran Mahmoud Ayman – 3679

Shahenda Elsayed – 3695

John Maged – 3690

Implementation of reliable data transfer using RDT 3.0, Selective repeat and GO-Back N algorithms

Stop and wait RDT 3.0:

At first Client request the file from the server. Server then starts by sending a packet and it waits for the client to send an acknowledgment. There are two alternating Sequence bits (0 and 1)

Pipelined Protocols: Sender allows multiple yet to be acknowledged packets.

Selective Repeat: Sender can have up to N unacknowledged packets in pipeline. Receiver sends individual acknowledgment for each packet. Sender maintains a timer for each unackd packet. Sender maintains a sliding window to keep record of packets that are already ackd, packets that are sent but not yet ackd and packets that are ready but not yet sent.

At first the client requests a file from the server. The server then starts sending the packets in the window, the client then upon receiving the packet sends acknowledgment individually. When the server sends a packet it starts a timer for each corresponding packet .

GO-Back-N: Sender can have up to N unacknowledged packets in pipeline. Receiver only sends cumulative ack, it doesn’t ack a packet if there is a gap .when timer expires , retransmit all unacked packets .

The performance of the three techniques :

the selective repeat is the fastest one then Go-Back-N then stop and wait .

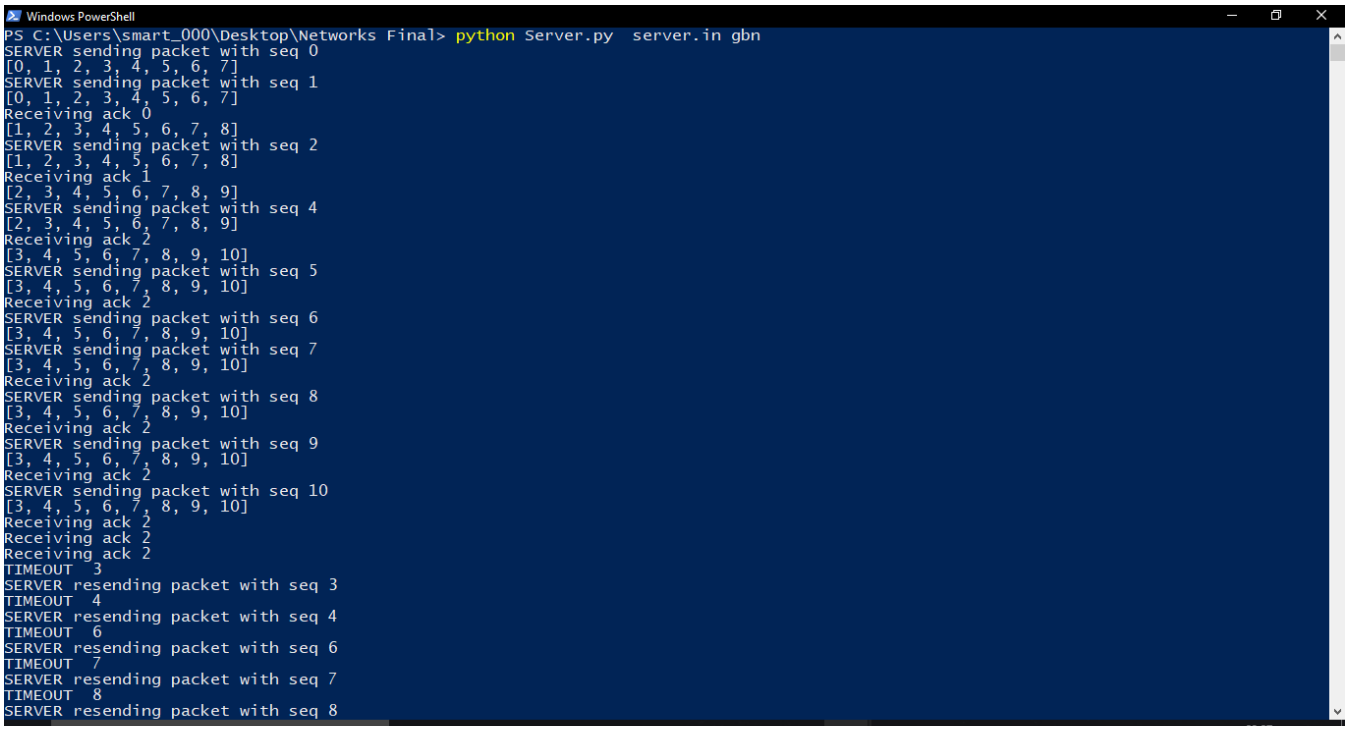
Run of stop and wait :

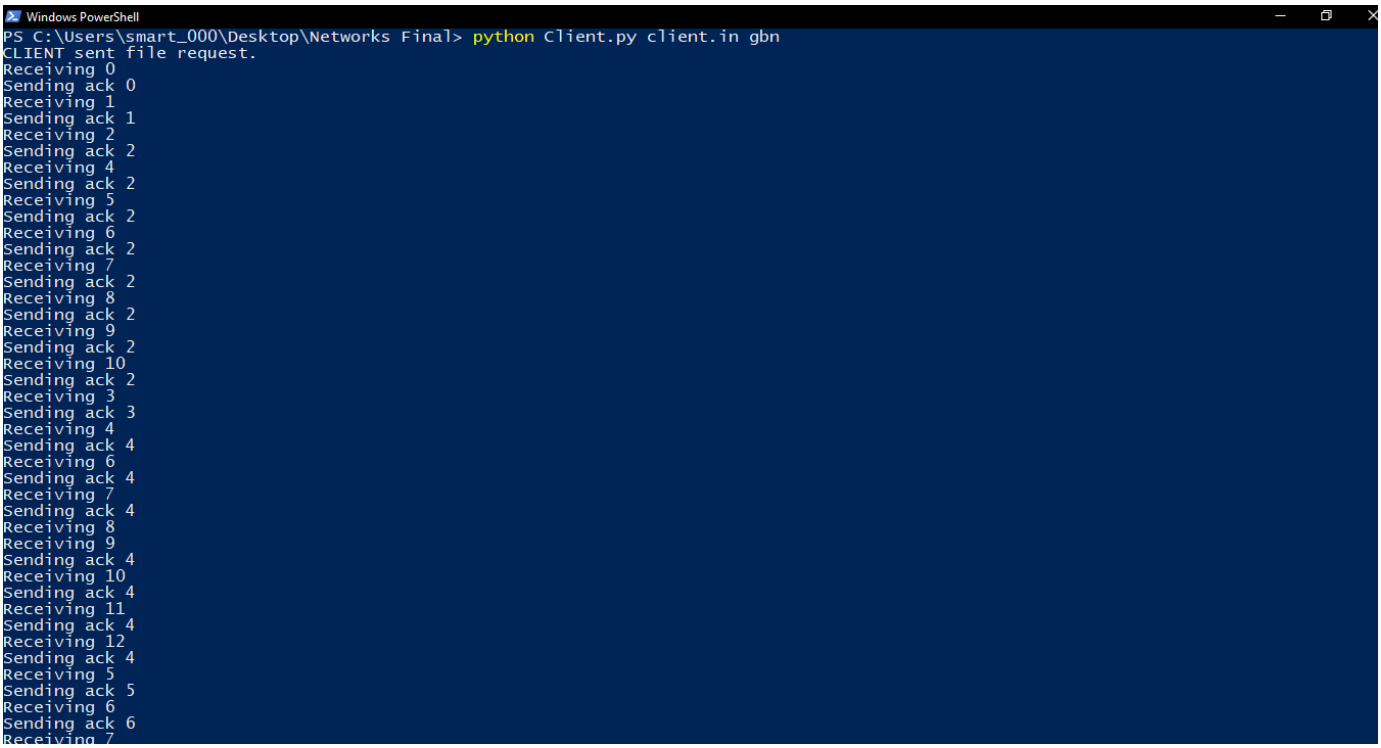
```
Windows PowerShell
PS C:\Users\smart_000\Desktop\Networks Final> python Server.py  server.in rdt
SERVER Sending packet with seq no  0
SERVER received ACK 0
SERVER Sending packet with seq no  1
SERVER received ACK 1
SERVER Sending packet with seq no  0
SERVER received ACK 0
SERVER Sending packet with seq no  1
SERVER received ACK 1
SERVER Sending packet with seq no  0
SERVER received ACK 0
SERVER Sending packet with seq no  1
SERVER received ACK 1
SERVER Sending packet with seq no  0
SERVER received ACK 0
SERVER Sending packet with seq no  1
SERVER received ACK 1
SERVER Sending packet with seq no  0
SERVER received ACK 0
SERVER Sending packet with seq no  1
SERVER received ACK 1
SERVER Sending packet with seq no  0
SERVER received ACK 0
SERVER Sending packet with seq no  1
SERVER received ACK 1
SERVER Sending packet with seq no  0
SERVER received ACK 0
SERVER Sending packet with seq no  1
SERVER received ACK 1
SERVER Sending packet with seq no  0
SERVER received ACK 0
SERVER Sending packet with seq no  1
SERVER received ACK 1
SERVER Sending packet with seq no  0
SERVER received ACK 0
SERVER Sending packet with seq no  1
SERVER received ACK 1
SERVER Sending packet with seq no  0
SERVER received ACK 0
SERVER Sending packet with seq no  1
SERVER received ACK 1
SERVER Sending packet with seq no  0
SERVER received ACK 0
SERVER Sending packet with seq no  1
```

```
Windows PowerShell
PS C:\Users\smart_000\Desktop\Networks Final> python Client.py client.in rdt
CLIENT sent file request.
CLIENT received packet with sequence 0
CLIENT resending ACK 0
CLIENT sent ACK 0
CLIENT expecting packet 1
CLIENT received packet with sequence 1
CLIENT resending ACK 1
CLIENT sent ACK 1
CLIENT expecting packet 0
CLIENT received packet with sequence 0
CLIENT resending ACK 0
CLIENT sent ACK 0
CLIENT expecting packet 1
CLIENT received packet with sequence 1
CLIENT resending ACK 1
CLIENT sent ACK 1
CLIENT expecting packet 0
CLIENT received packet with sequence 0
CLIENT resending ACK 0
CLIENT sent ACK 0
CLIENT expecting packet 1
CLIENT received packet with sequence 1
CLIENT resending ACK 1
CLIENT sent ACK 1
CLIENT expecting packet 0
CLIENT received packet with sequence 0
CLIENT resending ACK 0
CLIENT sent ACK 0
CLIENT expecting packet 1
CLIENT received packet with sequence 1
CLIENT resending ACK 1
CLIENT sent ACK 1
CLIENT expecting packet 0
CLIENT received packet with sequence 0
CLIENT resending ACK 0
CLIENT sent ACK 0
CLIENT expecting packet 1
CLIENT received packet with sequence 1
CLIENT resending ACK 1
CLIENT sent ACK 1
CLIENT expecting packet 0
CLIENT received packet with sequence 0
CLIENT resending ACK 0
```

Run of Go-Back-N :

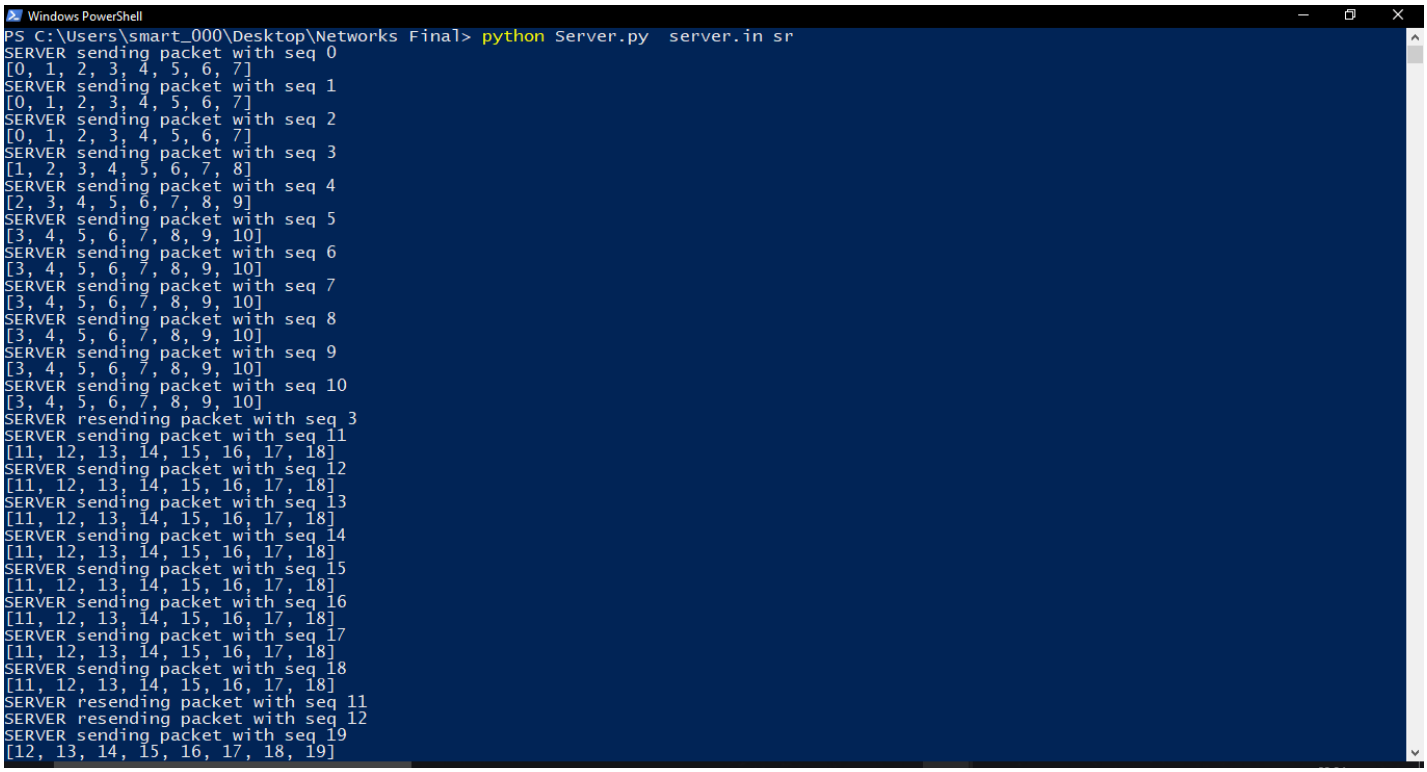
note that in this test case the window size = 8

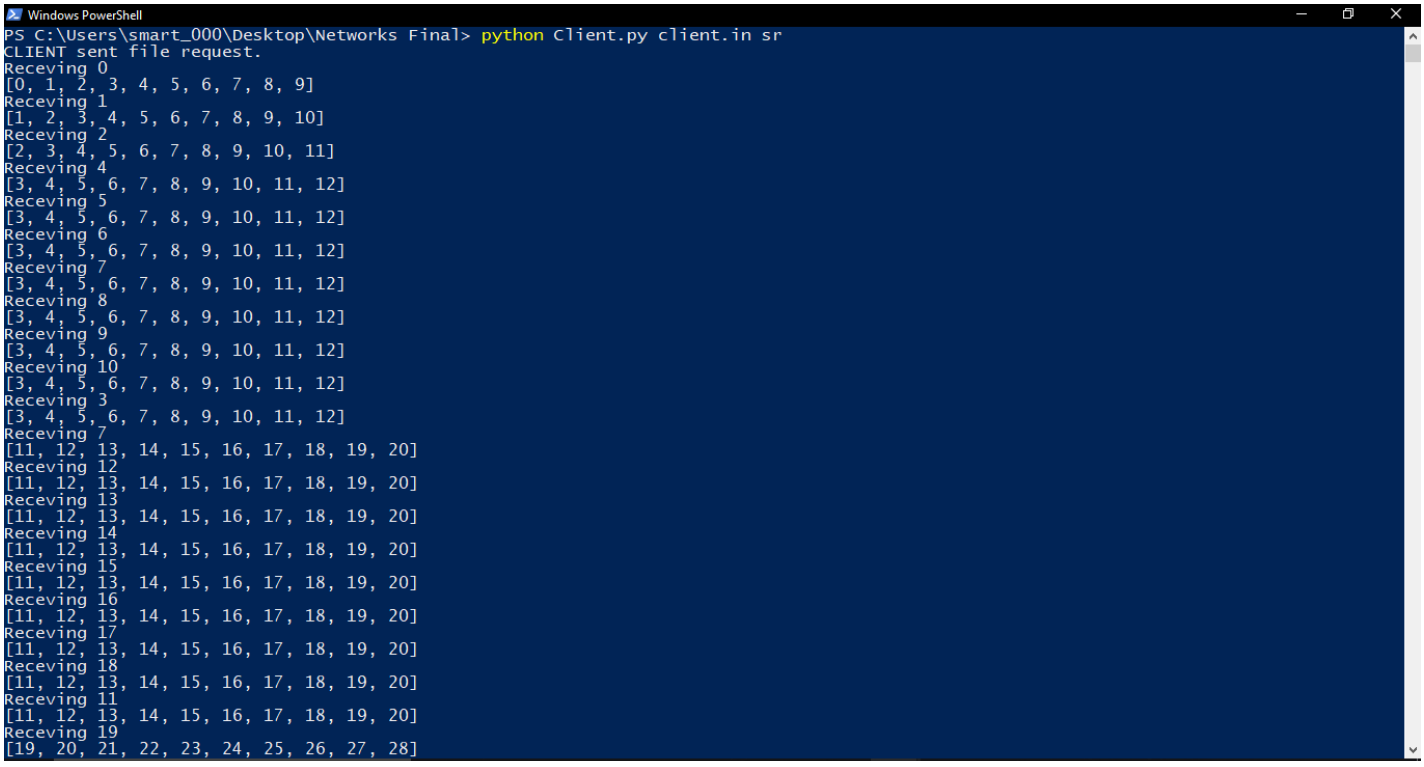




Run of Selective repeat :

note that in this test case the window size = 8





How to run code :

to run stop and wait

in cmd prompt > python Server.py server.in rdt

another cmd prompt > python Client.py client.in rdt

to run Go-Back-N

in cmd prompt > python Server.py server.in gbn

another cmd prompt > python Client.py client.in gbn

to run selective repeat

in cmd prompt > python Server.py server.in sr

another cmd prompt > python Client.py client.in sr