Explanation
Socket Programming: Question 1
server.c

1. The server socket is set up on port 5000, along with an array of length 30 to store the client socket FDs.
2. The socket is bound and the server starts listening
3. When the client connects to the server, the server adds the client's FD to the array if it's a new connection and sends a welcome message
4. The server receives the string sent by the client and sends the reverse of it to the corresponding client FD, and displays the received and sent strings on output
5. When the client disconnects by sending an exit signal, the FD is removed from the array and the server continues to listen on that port

client.c

1. The client socket is set up, and sends a connection request to the server at port 5000
2. When the connection is successfully made, it receives a welcome message from the server
3. The client sends a string and receives the reverse of it, which is displayed on the output, and the connection remains open
4. The client disconnects from the server when an exit signal is sent

NS2: Question 3
TCL Code

1. The properties of the network are set as elements of the variable val, where the channel type is set as wireless and the routing protocol is set as AODV or DSR
2. The simulator is created and files for trace and nam are opened and are are linked to trace-all and namtrace-all-wireless respectively
3. The GOD object is created to store global information about the state of the wireless network
4. The configuration of the simulator is added through node-config from the values defined earlier in the variable val
5. 10 nodes are initialised with X and Y positions according to the topology and the random motion is set to 0
6. To generate the movements of node 6, the corresponding destination and speed to move is defined for times 10 and 20 seconds
7. To generate UDP connections, a UDP agent is attached with a CBR traffic source, and attached to node 1. A corresponding NULL agent is attached to node 9, and the connection between the UDP source and NULL destination is made. The CBR source is instructed to start transmission at 2 seconds.
8. To generate TCP connections, a TCP agent is attached with a FTP traffic source, and attached to node 1. A corresponding TCPSink agent is attached to node 8, and the connection between the TCP source and TCPSink is made. The FTP source is instructed to start transmission at 30 seconds.
9. A stop procedure is defined to handle the end of the simulation, where the trace is flushed to the recording files and the files are close. AWK scripts are executed to generate the XG files needed for plotting the graphs and the NAM file is executed.

Graphs

Throughput - The throughput initially is low, but increases once the routes from the source and destination has been identified. There is a change in the graph at 30 seconds, which is due to the introduction of the TCP traffic to the network

PDR - The packet drop rate is initially high, but decreases once the routes from the source and destination has been identified. The drop rate decreases substantially at around 30 seconds, which indicates that the TCP packets had a lower drop rate after starting transmission

Delay - The delay increases as more traffic is introduced to the network, but eventually evens out as the routes from the source to the destination have been identified.

DSR and AODV Graph Comparison

Throughput - The trends are similar, although the steady throughput of DSR is higher than AODV

Delay - In DSR there is a downward trend in Delay as time goes, while in AODV it remains mostly constant, however the delay in AODV is less than DSR

PDR - The trends are similar, although the steady PDR of AODV is slightly less than DSR

Difference between AODV and DSR

1. AODV works on hop by hop routing with routing table entry lifetimes, while DSR is source routing without lifetimes
2. AODV stores only one route per destination, while DSR uses caching and stores multiple routes for the same destination
3. As AODV entries have a lifetime, route discovery cycle is much more frequent in AODV than DSR, which also leads to better bandwidth