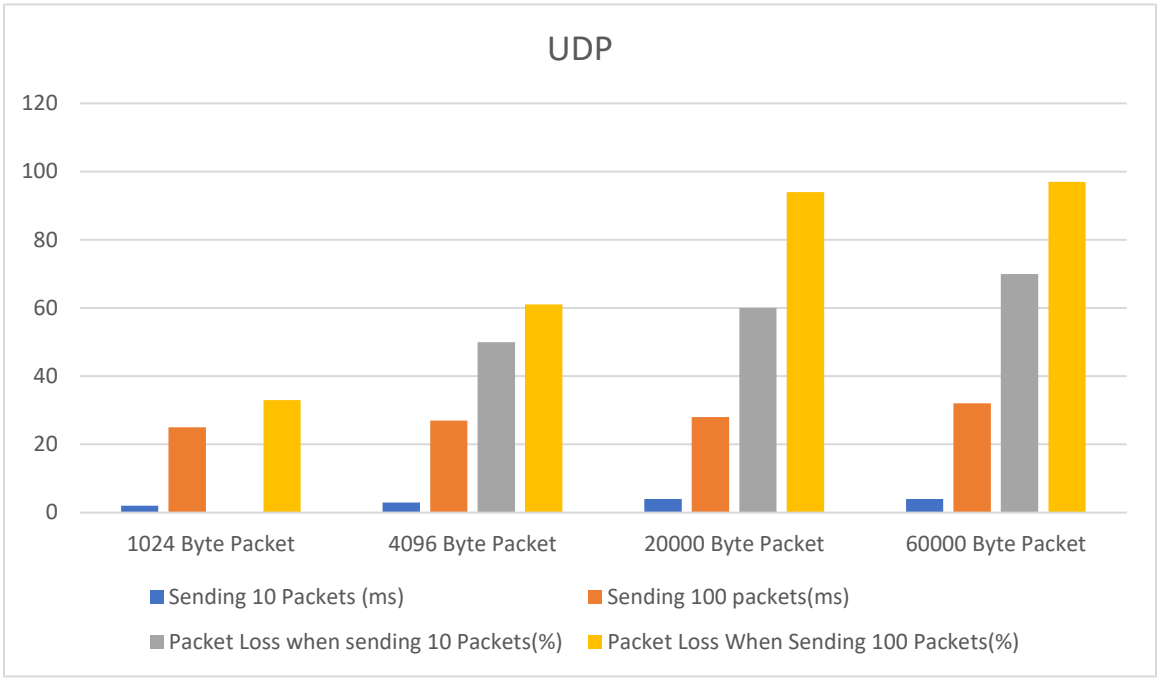
Analysis and Report

This section will compare transmission times obtained from sending packets of various sizes using UDP and TCP.

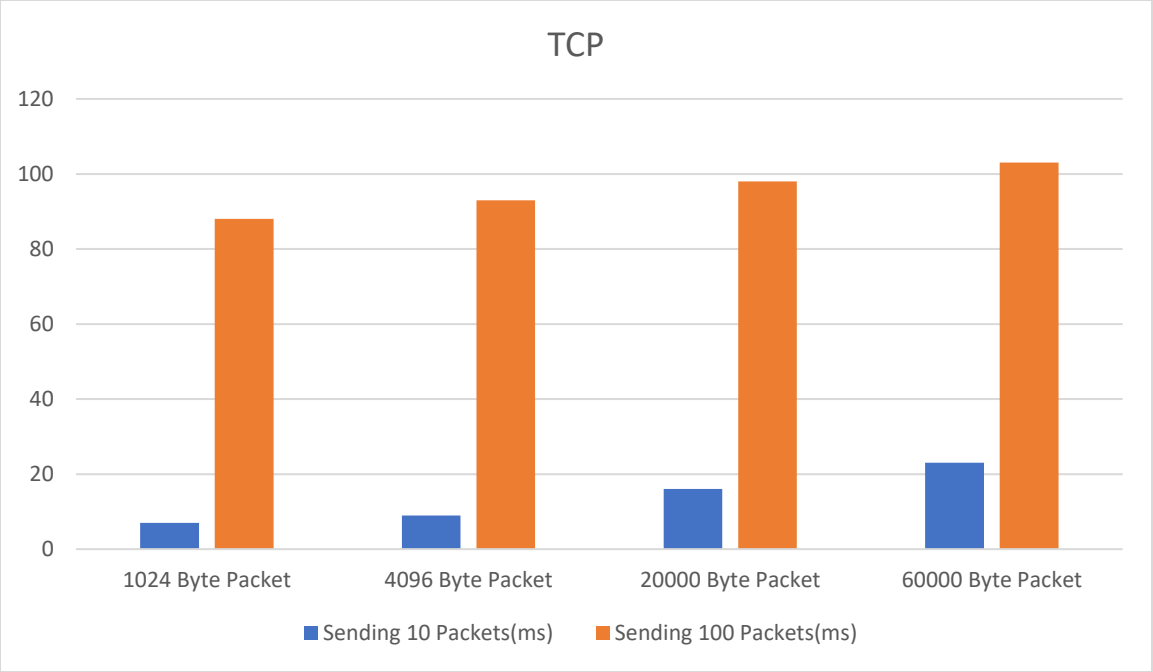


The diagram above shows the results obtained after sending 10 and 100 packets of various packet sizes in UDP.

We can see that as packet size increases:

- Transmission time of data increases
- Packet Loss percentage increases

The transmission times range from 2ms up to 5ms when sending 10 packets, and 23ms to 35ms when sending 100 packets.



The diagram above shows the results obtained after sending 10 and 100 packets of various packet sizes in TCP.

Similarly to UDP, as packet size increases, transmission time of data increases too. TCP, however, takes much longer to transmit data. The range for TCP's transmission times is 7ms to 23ms for sending 10 packets and 85ms to 105ms when sending 100 packets, almost four times as long as UDP.

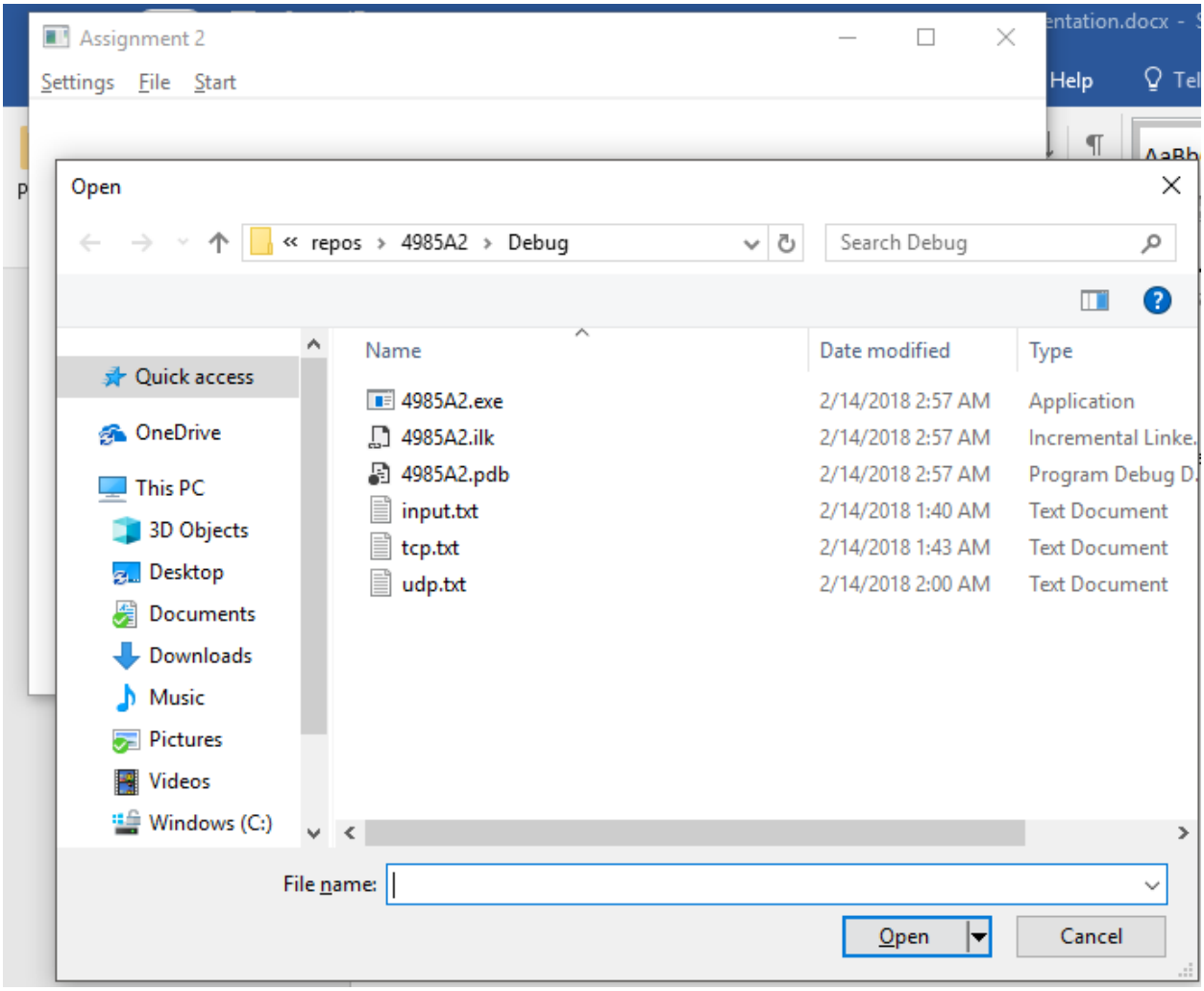
Conclusion

UDP is faster than TCP, but TCP is more reliable. As packet size increases, it is more likely for the packet to contain errors and be discarded in UDP.

TCP is better suited for applications that require high reliability; UDP is suited for applications that need fast transmission. An example for TCP would be file transfers and for UDP, gaming.

Testing

Selecting a file



TCP Connection Established

Select Command Prompt

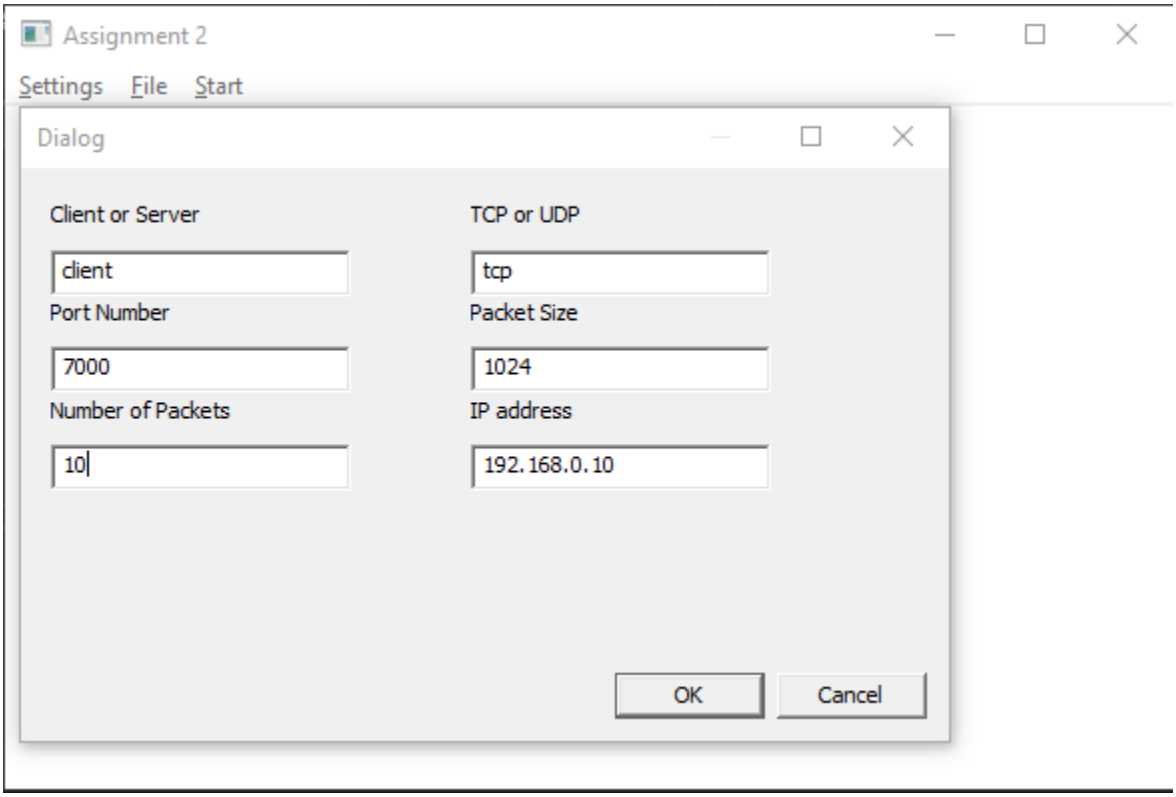
```
  TCP    127.0.0.1:53545        0.0.0.0:0              LISTENING
  TCP    127.0.0.1:53545        127.0.0.1:64439        ESTABLISHED
  TCP    127.0.0.1:56333        0.0.0.0:0              LISTENING
  TCP    127.0.0.1:56334        0.0.0.0:0              LISTENING
  TCP    127.0.0.1:56335        0.0.0.0:0              LISTENING
  TCP    127.0.0.1:56336        0.0.0.0:0              LISTENING
  TCP    127.0.0.1:64437        127.0.0.1:27015        ESTABLISHED
  TCP    127.0.0.1:64439        127.0.0.1:53545        ESTABLISHED
  TCP    192.168.0.10:139       0.0.0.0:0              LISTENING
  TCP    192.168.0.10:7000      192.168.0.10:53810     ESTABLISHED
  TCP    192.168.0.10:49441     13.89.184.238:443      ESTABLISHED
  TCP    192.168.0.10:49487     142.232.230.30:443     FIN_WAIT_2
  TCP    192.168.0.10:51909     52.43.202.10:443       ESTABLISHED
  TCP    192.168.0.10:52464     157.55.170.113:5671    ESTABLISHED
  TCP    192.168.0.10:53365     34.210.102.41:443      ESTABLISHED
  TCP    192.168.0.10:53604     13.89.184.96:443       ESTABLISHED
  TCP    192.168.0.10:53768     13.33.151.143:443      CLOSE_WAIT
  TCP    192.168.0.10:53810     192.168.0.10:7000      ESTABLISHED
  TCP    192.168.0.10:53812     40.121.213.159:443     TIME_WAIT
```

UDP Connection Established

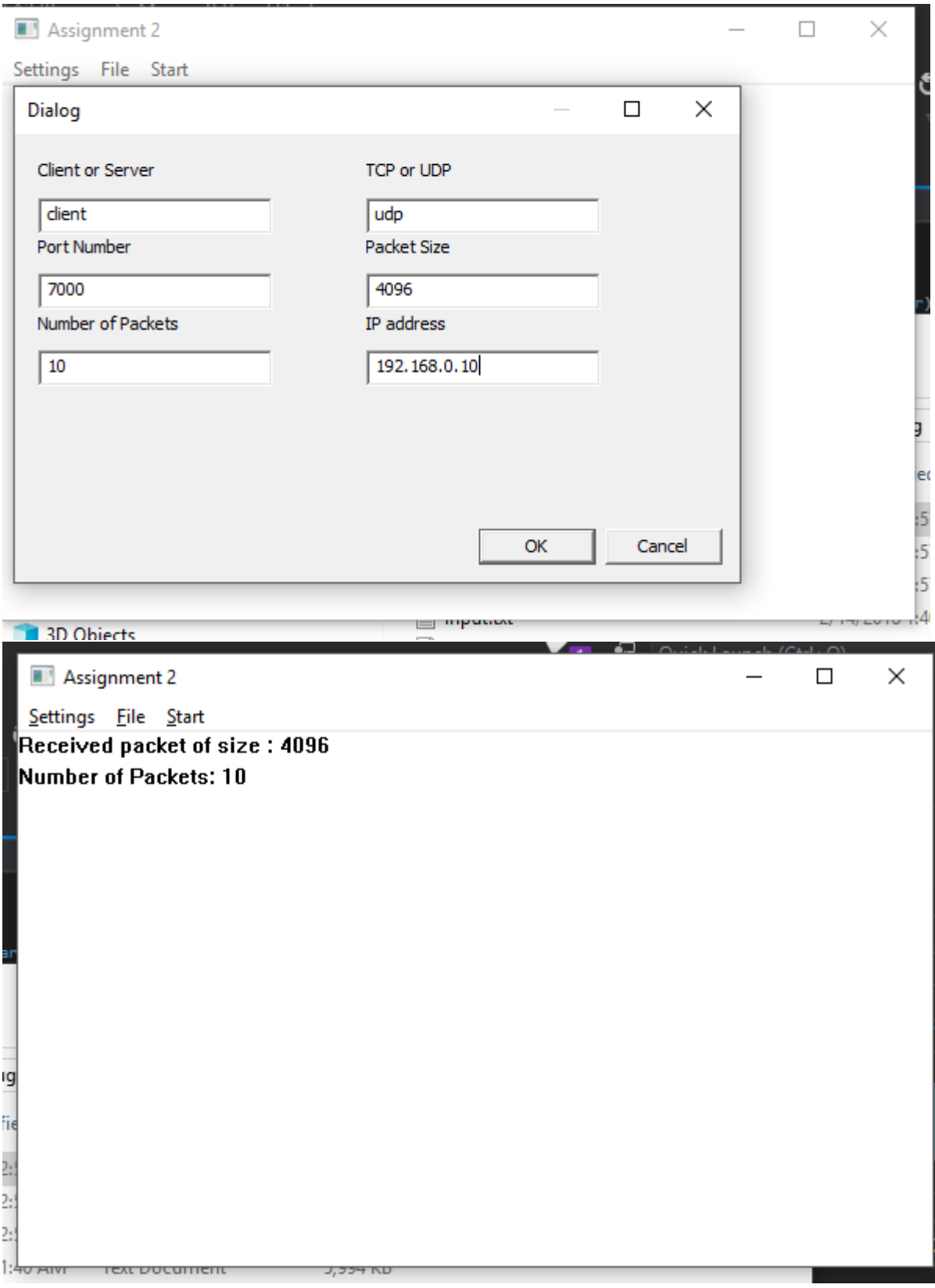
Select Command Prompt

```
  UDP    0.0.0.0:3544           *:*
  UDP    0.0.0.0:3702           *:*
  UDP    0.0.0.0:3702           *:*
  UDP    0.0.0.0:3702           *:*
  UDP    0.0.0.0:3702           *:*
  UDP    0.0.0.0:3702           *:*
  UDP    0.0.0.0:3702           *:*
  UDP    0.0.0.0:5050           *:*
  UDP    0.0.0.0:5353           *:*
  UDP    0.0.0.0:5353           *:*
  UDP    0.0.0.0:5353           *:*
  UDP    0.0.0.0:5353           *:*
  UDP    0.0.0.0:5353           *:*
  UDP    0.0.0.0:5353           *:*
  UDP    0.0.0.0:5355           *:*
  UDP    0.0.0.0:7000           *:*
  UDP    0.0.0.0:50532          *:*
  UDP    0.0.0.0:50955          *:*
  UDP    0.0.0.0:52484          *:*
  UDP    0.0.0.0:52692          *:*
  UDP    0.0.0.0:52693          *:*
  UDP    0.0.0.0:53698          *:*
  UDP    0.0.0.0:55233          *:*
```

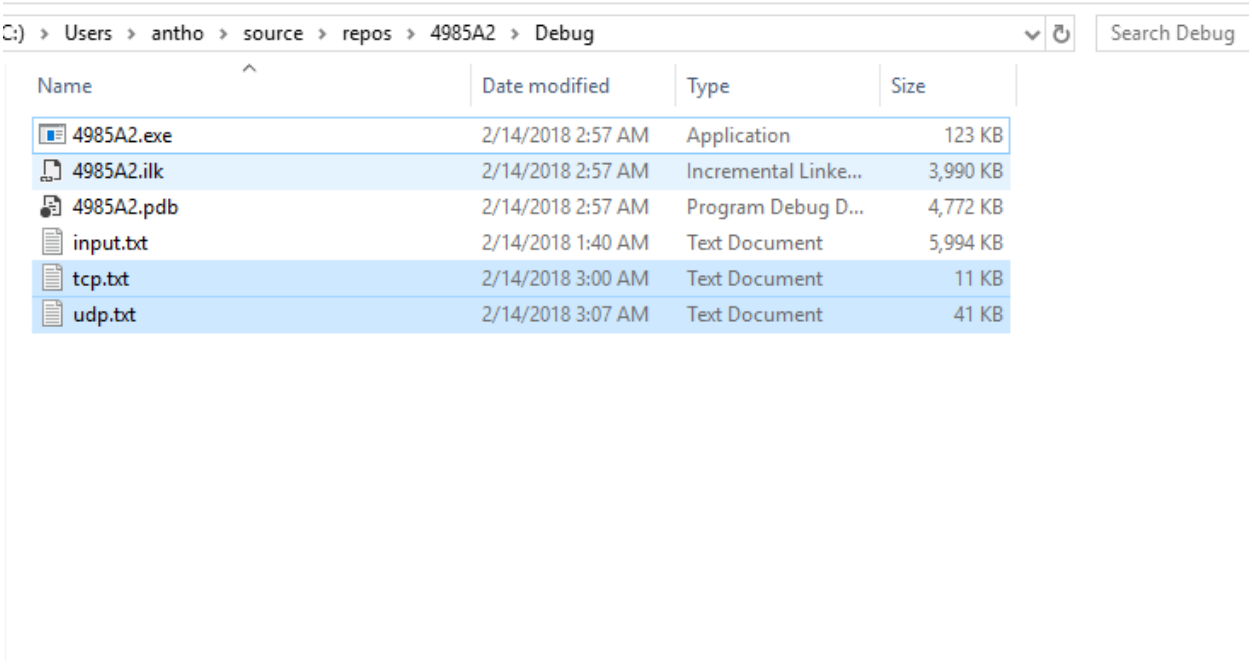
Sending Data Using TCP



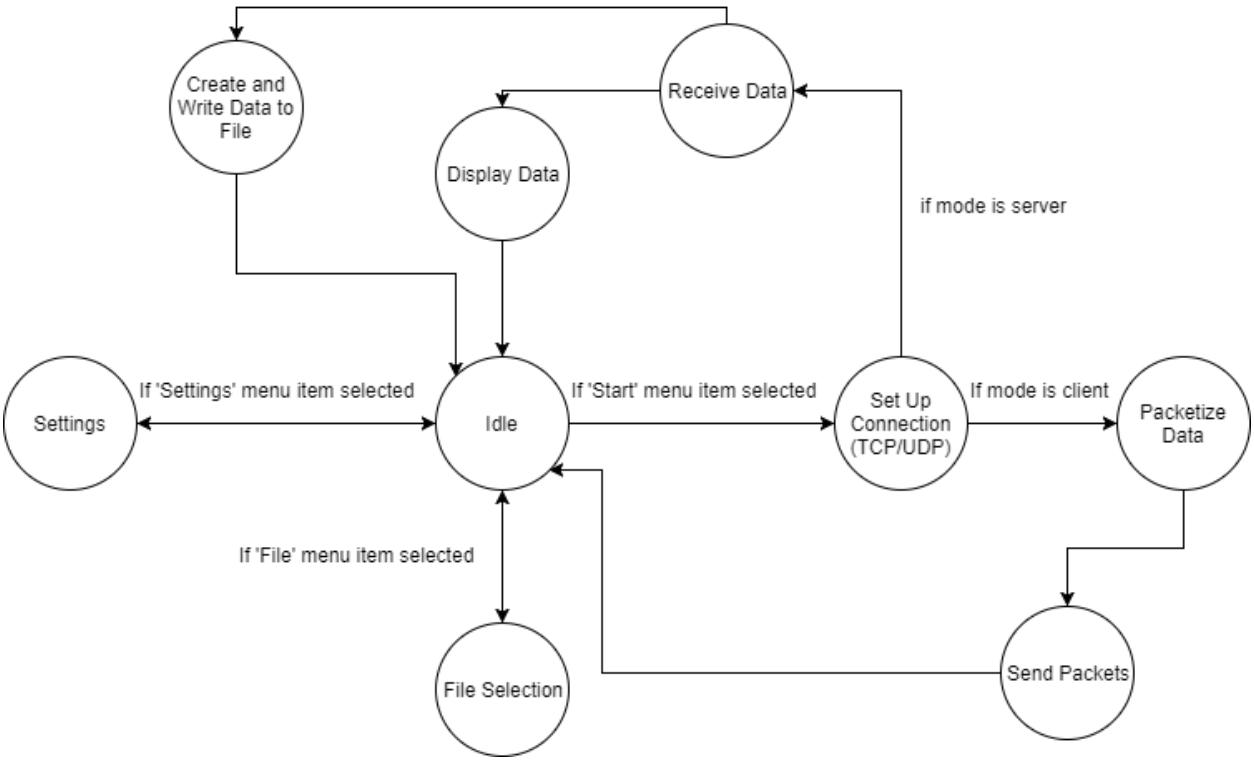
Sending Data Using UDP



Data Saved to File



State Chart



Idle

Starting state of the program. Displays a simple GUI to the user.

- If user decides to upload a file, go to **File Selection** state
- If user decides to change Settings, go to **Settings** state
- If user wants to begin sending or receiving data, go to **Set Up Connection** state

File Selection

Opens another GUI to allow the user to select a file to upload to a server. Selecting 'OK' will send the user back to the **Idle** state.

Settings

Opens a dialog box for the user to enter in values for:

- Mode: Client or Server
- Protocol: TCP or UDP
- Port number
- Packet Size
- Number of packets to send
- IP address

Selecting 'OK' or 'Cancel' will send the user back to the **Idle** state.

Set Up Connection

For TCP:

- Create a socket
- Client requests a connection
- Server Accepts Request

For UDP:

- Create a socket
- Both Client and Server bind to a local address

If set up is successful, go to either **Packetize Data** state or **Receive Data** state, depending on the mode

Packetize Data

Breaks up the file specified by the user into packets of n Bytes, where n is a value specified by the user in the **Settings** state. Once the data has been broken up, go to **Send Packets** state.

Send Packets

Send packets to a socket. When all packets have been sent or max number of packets has been sent, return to **Idle** state.

Receive Data

If there is data in the socket, go to **Display Data** state and **Create and Write Data to File** state.

Display Data

Display number of packets Received, as well as the size of the most recent packet received. Return to **Idle** state.

Create and Write Data to File

Creates a file to save data. Return to **Idle** state.

Pseudocode

WinMain

Initialize variables for creating window and opening files
Create window

WndProc

Create WM_SOCKET event handler
    If WM_SOCKET event triggered
        Call ServerProc() or ClientProc()
Create File Selector event handler
    If File Selector event triggered
        show window for file selection
Create Settings event handler
    If Settings button clicked
        Show dialog box for settings
Create Start event handler
    If Start button clicked
        Set up Connection for TCP/UDP
Create WM_PAINT event handler
    Repaint the window every time a packet is received and display that value

ClientProc

Create Socket Information for socket if connected
Listen for ready-for-write events
    Packetize data received from file
    Send data to socket
    Calculate delay for sending

ServerProc

Accept a request for connection if requested
Create Socket Information for accepted socket
Listen for read events
    If there is data in the socket
        Retrieve data and send to file
        Output data to the GUI
    Clear Buffer

Lowercase

Loop through each character in the array
    Change character to lowercase

setDefaultSettings

set default values for mode, protocol, port number, packet size, max packets, and IP address

GetSocketInformation

Look for socket in SocketInfoList
    If SocketInfoList contains socket

Return socket

CreateSocketInformation

Set Values for LPSOCKET_INFORMATION struct of a socket

FreeSocketInformation

Look for Socket in SocketInfoList

If SocketInfoList contains socket

Free Socket

ToolDlgProc

Handle dialog box events for changing settings

Delay

Subtract end time from original time and multiple by 1000
Subtract end milliseconds from original milliseconds
Add values together and return that value