Assignment 2 Document

CP372

Kosta Kalenteridis

150383370

Due Date: March 7, 2018

HandShake:

My file transfer begins with a handshake between the sender and the receiver while both are already running. Once the information is filled in on the receiver’s gui, the user selects Reliable or Unreliable Mode and clicks the button labelled “TRANSFER”. At this point, a datagram is sent from the receiver to the already running sender notifying the sender for the request of the transfer. This datagram includes the filename inputted in the GUI, with a datagram length of 100 bytes. The sender receives this information and checks the receiver’s address/port along with a check that makes sure the requested file name matches the one in the sender’s arguments. If the information matches, a datagram is sent back with the number of packets it will take to send the file. The receiver receives the datagram, derives the file size from the number of packets, and prompts the user with the file size to insure they want to download the file / have the space to download it. I included this handshake because I feel a file transfer program should include a confirmation for the file size so the receiver knows the exact file size they are receiving. If yes is clicked on the confirmation, an ACK is sent back to the sender notifying to begin transferring the data.

Data Transfer:

Once the sender confirms the receiver is waiting for transfer, the file transfer begins. Each datagram is 124 bytes long, with the last byte being a sequence number (0 and 1). Each datagram is sent with a timeout for an ACK from the receiver. The ACK matches the sequence number, and the receiver makes sure the corresponding packets have a sequence number that matches what it expects. The receiver reconstructs the file byte array by byte array as each packet is received in order. Once the last packet is sent from the sender side, it proceeds to send the EOT packet, notifying the receiver that there is no more data. The receiver confirms this, and finishes closes the file along with the socket.

Timing Tests:

Test 1: File size: 2.95kb , Mode Reliable, Timeout = 500

Total Time (MilliSeconds)= 46

Test 2: File Size 2.1 MB, Mode Reliable, Timeout 500

Total Time (MilliSeconds) = 1345

Test 3: File size: 2.1MB , Mode UnReliable, Timeout = 100

Packets Dropped: 1952

Total Time (Seconds)= 209

Total Time (MilliSeconds)= 209788

Test 4: File size: 2.95kb , Mode Reliable, Timeout = 10

Total Time (MilliSeconds)= 53