

A separate (typed) document (in PDF) of a page or so describing the overall program design, a verbal description of how it works, and design tradeoffs considered and made. Also describe possible improvements and extensions to your program (and sketch how they might be made). This should include instructions on how to install and run the program. Please include any needed file (make, etc. if any).

UDP server:

When the server receives a message, it is parsed and then passed to a math function, the math function will return a string "300,-1" if it cannot do the math, or "200,x oper y" on success. If anything fails during the message the server will send "300,-1". Occasionally the server will not respond to simulate an unreliable protocol, this is done with a pseudorandom math function.

UDP client:

The client takes input from the user, if the input was stop it raises an exception and ends the program, otherwise it sends it to the server. The server is composed mainly of two while loops, the first one which repeatedly asks and sends queries to the server, the inner is for retry. The timeout is set using python's built in socket receive timeout, this timeout helps set the wait time between sending and receiving. If the server does not resend the packet in time and the timeout is greater than 2 then the client will quit.

TCP client:

The client attempts to establish a connection with the server, then takes user input and sends to the server, then it receives a reply and prints it out. The process loops the input so it always takes input until told to stop, then the server closes. The client prints out debugging messages at every stage.

TCP server:

The server opens a socket and listens for clients, when data is received it parses, and evaluates, if that fails it returns 300,-1. The server then takes that reply and sends it over the open connection to the client. The server prints out debugging messages at every stage.

Instructions:

To run python just do "python3 python.py", to run node js do "npm install" then "node file.js", remember to start the servers first.