

Alison Lee  
CS428 Project 2  
2021/07/08

## DESIGN

I initially wrote the code to be similar to that of the first project, but I quickly discovered the differences between code for UDP and TCP connections. The very first change was changing “SOCK\_DGRAM” to “SOCK\_STREAM” in a parameter for the initialization of the socket. I also found that the server would have to listen for incoming connections, and then accept each one.

I also found that the send and receive functions for TCP were similar to that for UDP, but didn’t have to “to” and “from” “suffixes” (sendto, recvfrom).

As for the client code, I found that the client would have to initiate the connection with connect().

At the bottom of the document, I have attached the execution for a Linux environment.

## ERROR CASES

Because I wrote my code meant to run in a Linux environment, significant changes had to be made for execution in Windows (Visual Studio). Because of missing POSIX headers in Windows, for example, I had to use Windows equivalents. WSStartup() had to be used to initialize winsock. In the end, I could not get my Windows code to work properly although my Linux code was fully functional.

Several errors can arise:

- During socket creation, where socket() will return a negative value.
- Binding can fail, and bind() returns a negative value. I found that this happened several times by chance, but also didn’t happen other times.
- Listening can fail, and listen() returns a negative value.
- Accepting the client connection can fail, and accept() returns a negative value.

I didn’t check for the last two errors since they never failed when I executed my files, but I included a check for bind() since I found it to fail many times. It may take several “./server &” to get it to work.

```
salee889@remote01:~/cs428/project2-linux$ ./server &
[21] 658479
alee889@remote01:~/cs428/project2-linux$ Waiting for connections...
./clientX &
[22] 658494
alee889@remote01:~/cs428/project2-linux$ ./clientY &
[23] 658505
alee889@remote01:~/cs428/project2-linux$ Received messages
Sent: Client Y: Bob
Sent: Client X: Alice
Response: X: Alice received before Y: Bob
Sent acknowledgements
Response: X: Alice received before Y: Bob
```

```
alee889@remote-n31:~/cs428/project2-linux$ ./server &
[11] 917124
alee889@remote-n31:~/cs428/project2-linux$ Waiting for connections...
./clientY &
[12] 917134
alee889@remote-n31:~/cs428/project2-linux$ ./clientX
Received messages
Sent: Client X: Alice
Sent: Client Y: Bob
Sent acknowledgements
Response: Y: Bob received before X: Alice
Response: Y: Bob received before X: Alice
```