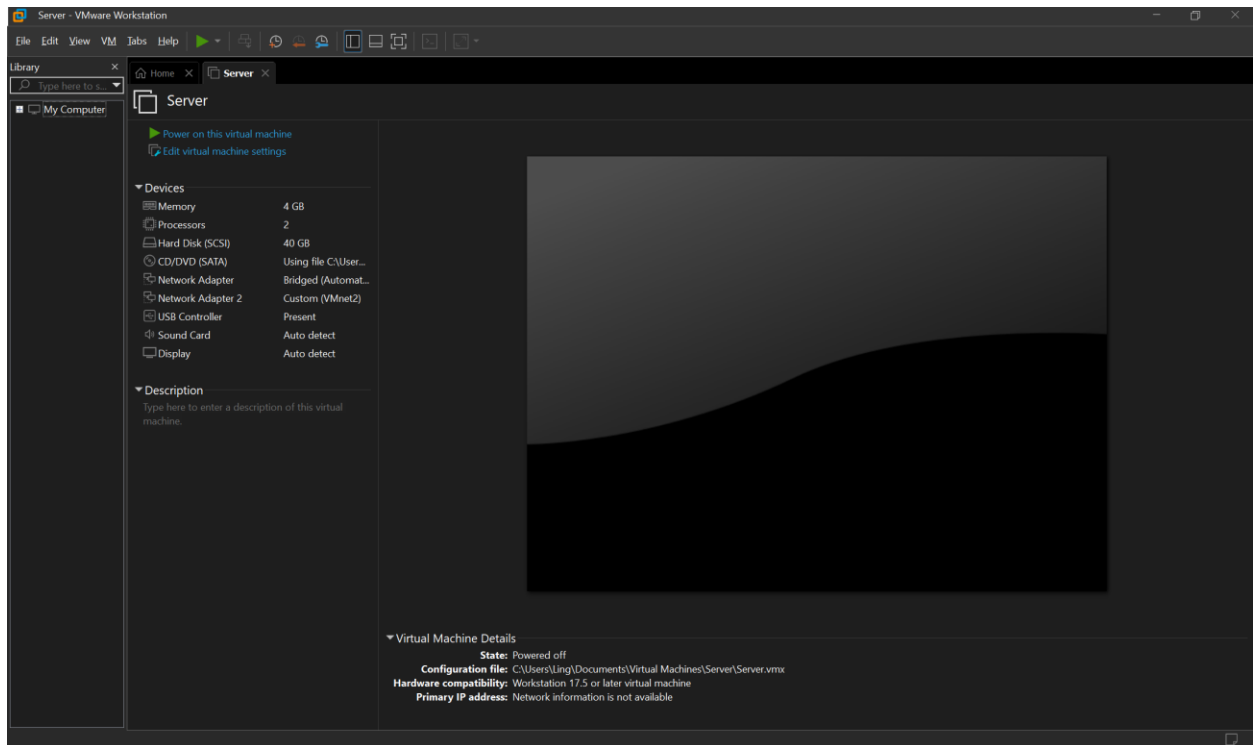
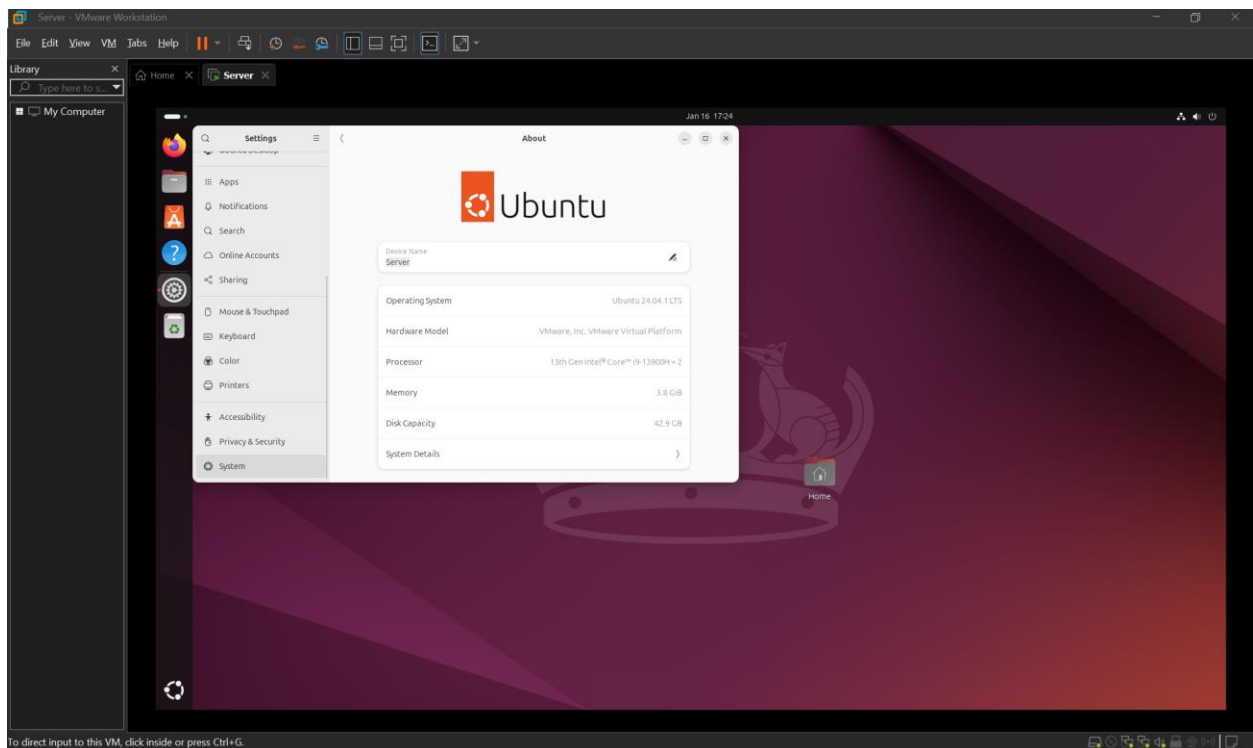


Part1: I went through the vmware New virtual Machine Wizard to create the first virtual machine using an ubuntu image I downloaded from the internet, and I call it server:

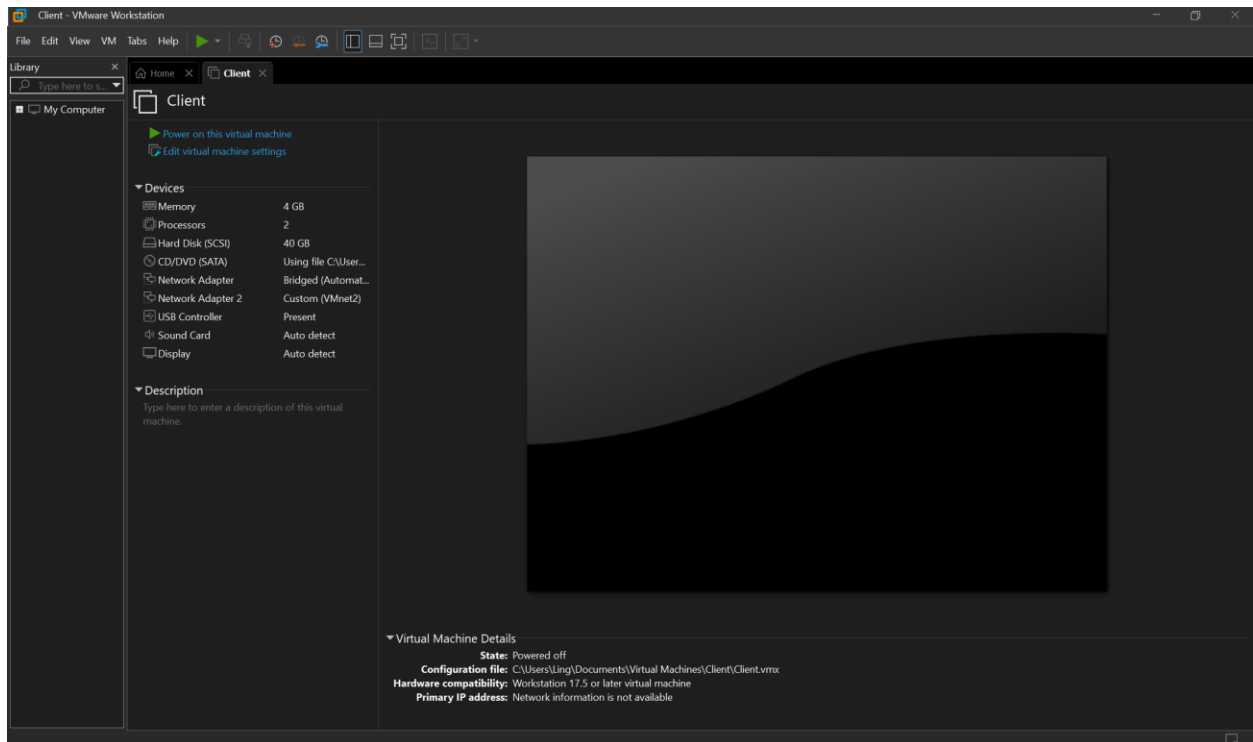




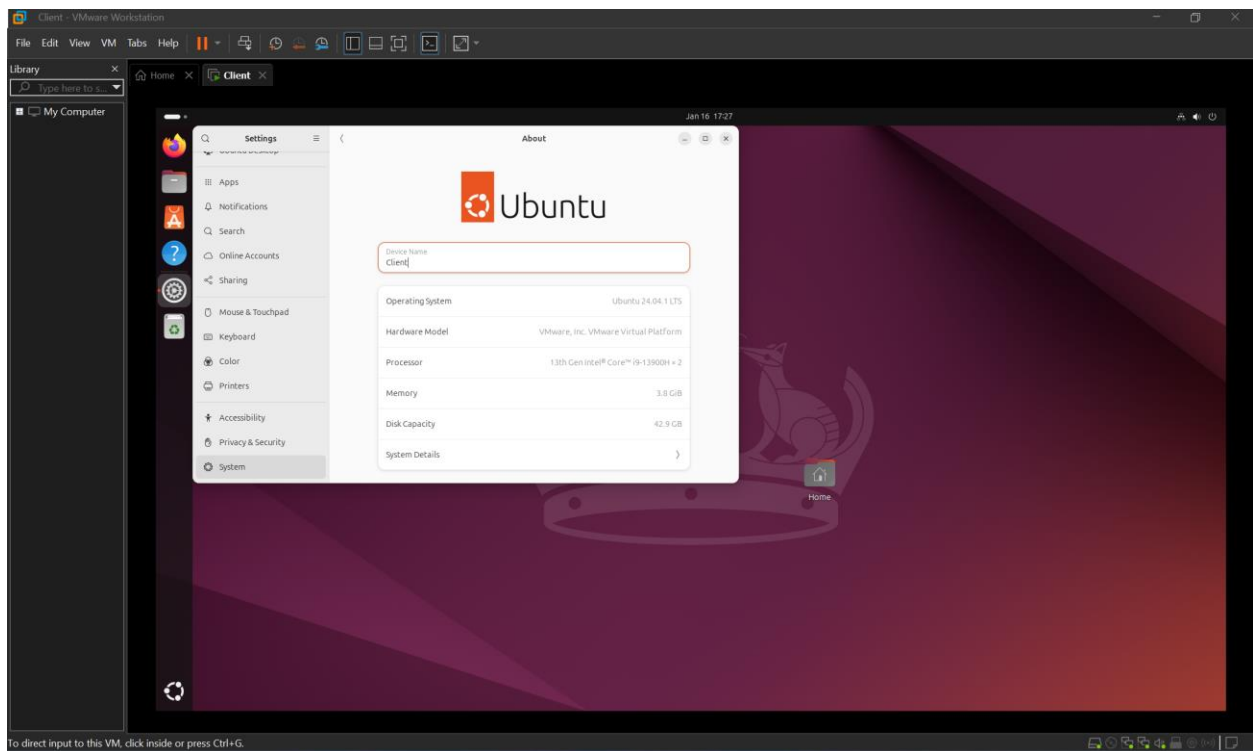
Here is a screenshot of it running:



I then use the clone feature vmware provides to clone the server into a completely different vm and I named the copy client:

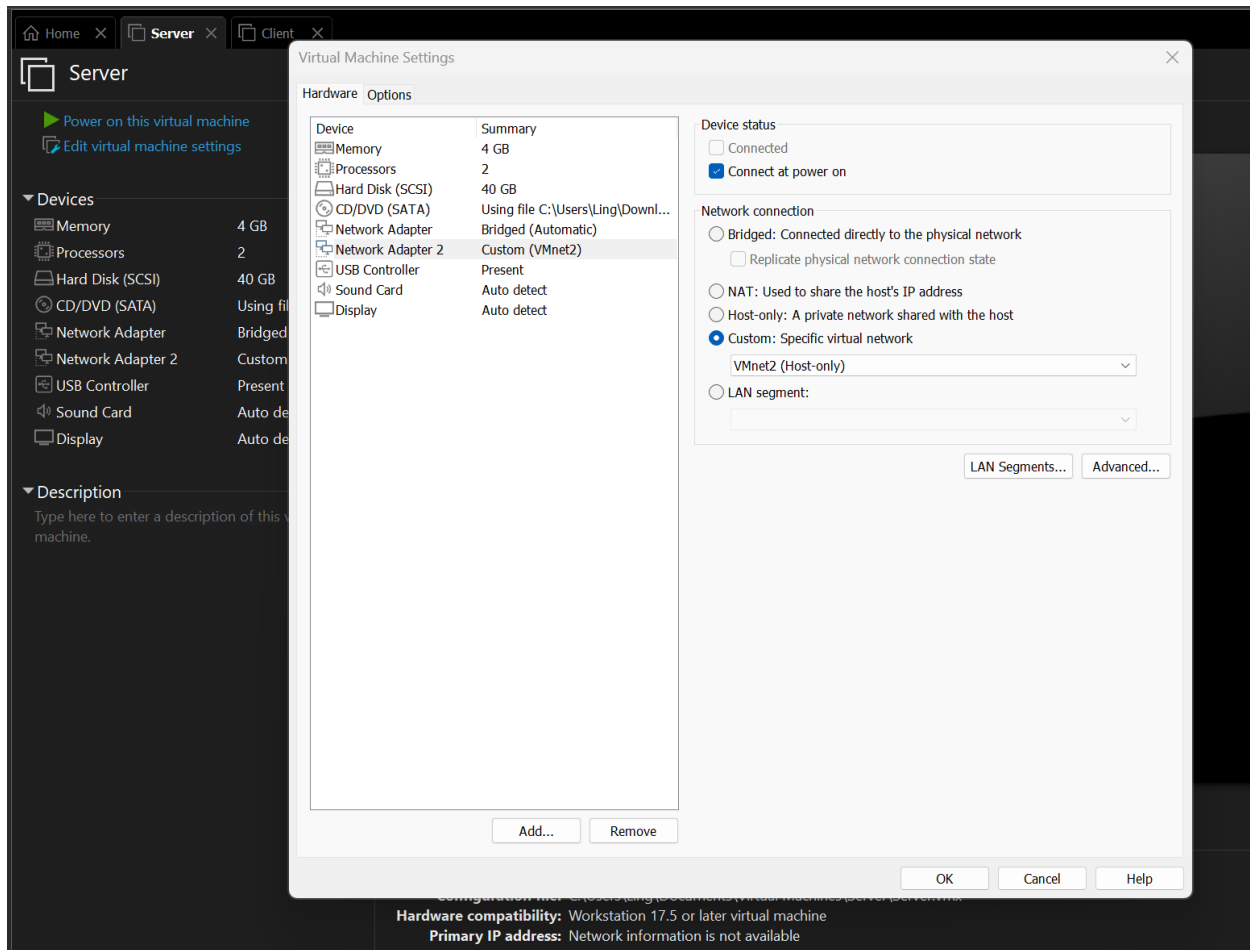


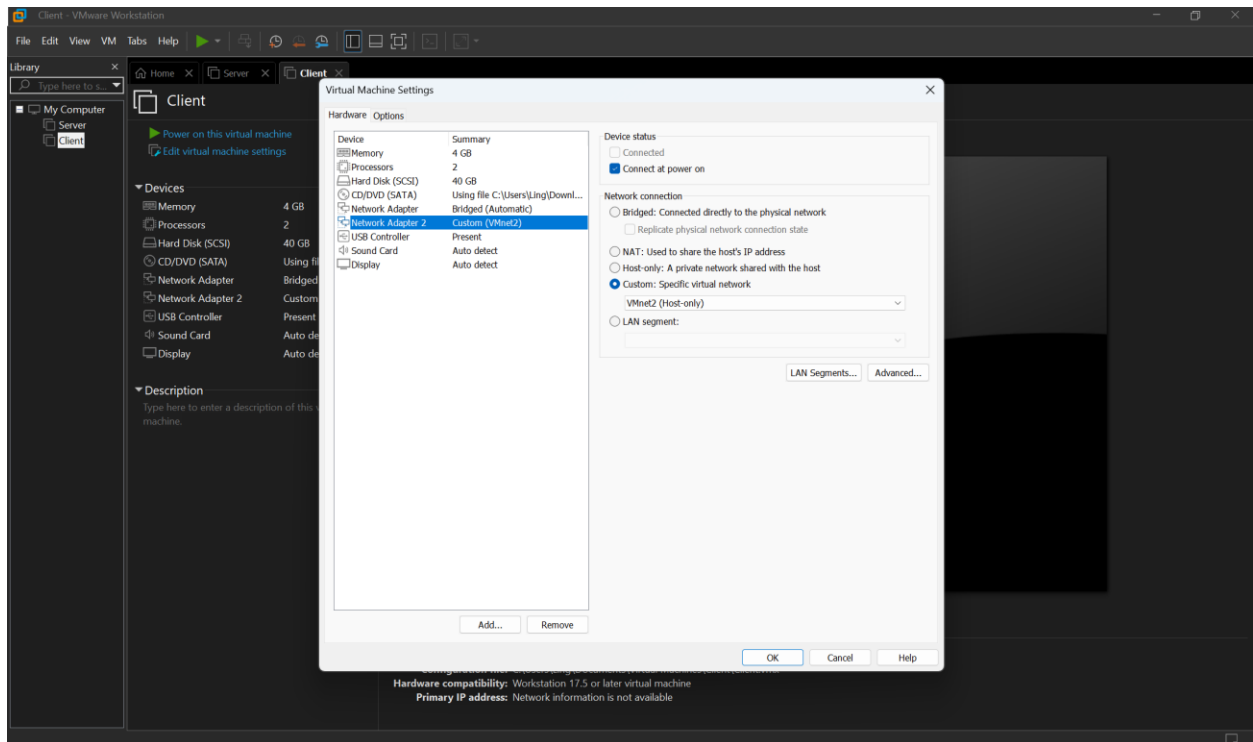
Here is a screenshot of it running:



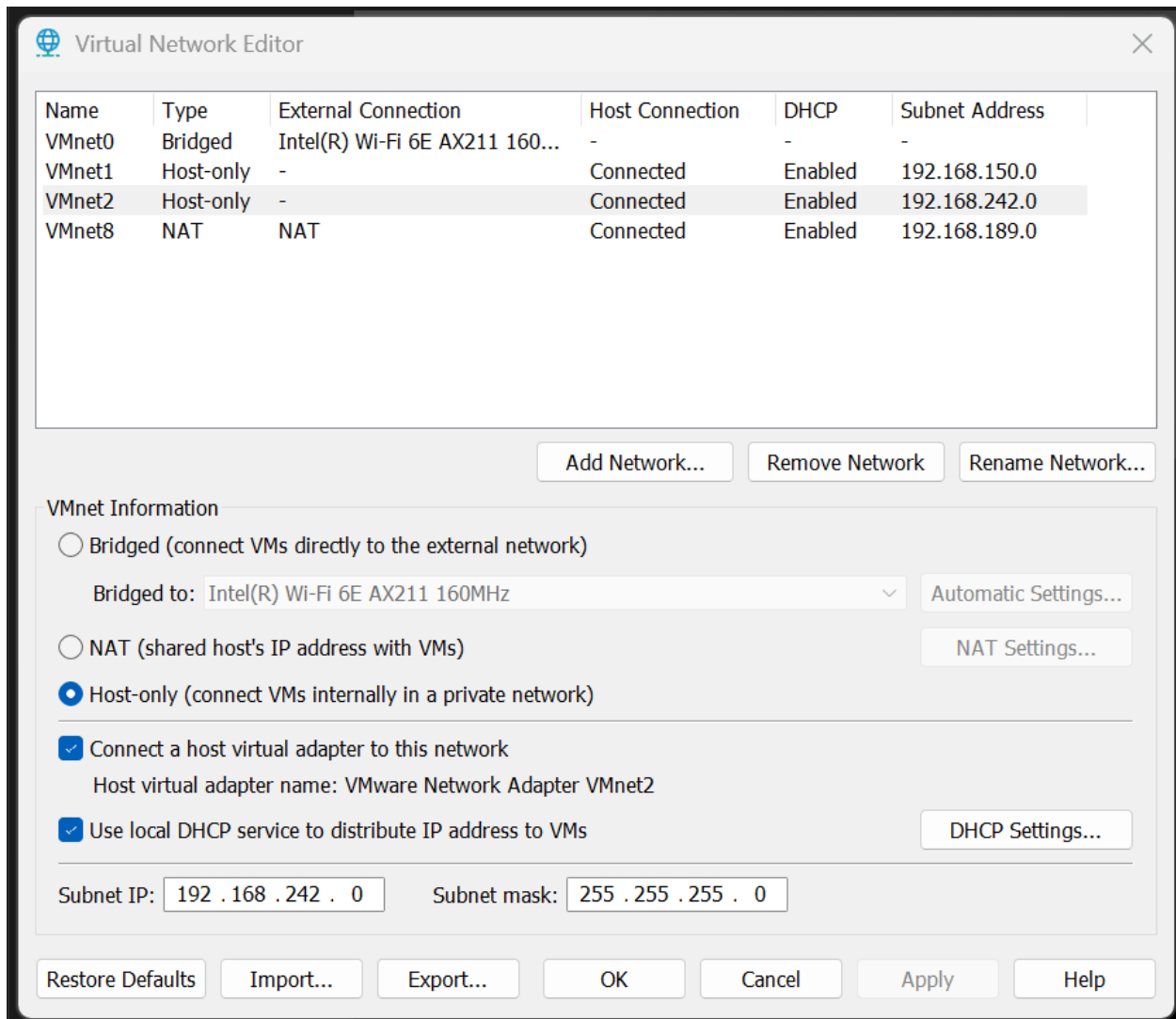
Part 2: Setting up the network between the VMs that I created:

I then read through the article provided in the pdf and change the first network adapter to bridged network and then adds a second network adapter to both Server and Client, and specify the virtual network that the second network adapter going to connect to VMnet2:





I then opens up the Virtual network editor vmware provides:



And then I realized that since local DHCP service is on, I don't have to manually assign IP addresses for server and client's second network adapter; so I continues.

Part 3: Try to get the simple client and server code to compile and run:

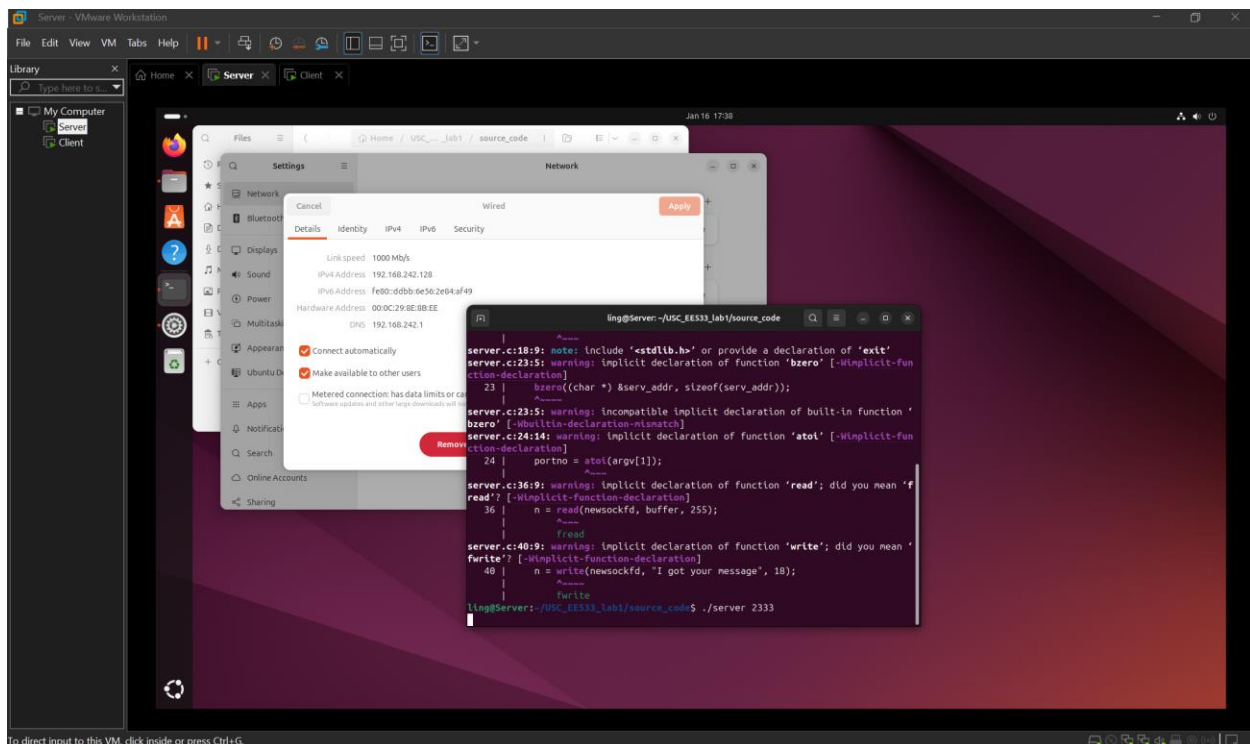
Here is a screenshot of the server code being compiled:

```
ling@Server:~/USC_EE533_lab1/source_code$ gcc -o server server.c
server.c: In function 'error':
server.c:8:5: warning: implicit declaration of function 'exit' [-Wimplicit-function-declaration]
    8 |     exit(1);
      |     ^~~~~
server.c:18:5: note: include '<stdlib.h>' or provide a declaration of 'exit'
    18 |     exit(1);
      |     ^~~~~
server.c:23:5: warning: incompatible implicit declaration of built-in function 'bzero' [-Wbuiltin-declaration-mismatch]
    23 |     bzero(char *) &serv_addr, sizeof(serv_addr));
      |     ^~~~~
server.c:23:5: note: include '<stdlib.h>' or provide a declaration of 'exit'
server.c: In function 'main':
server.c:18:9: warning: incompatible implicit declaration of built-in function 'exit' [-Wbuiltin-declaration-mismatch]
    18 |     exit(1);
      |     ^~~~~
server.c:23:5: note: include '<stdlib.h>' or provide a declaration of 'exit'
server.c:23:5: warning: implicit declaration of function 'bzero' [-Wimplicit-function-declaration]
server.c:24:14: warning: implicit declaration of function 'atoi' [-Wimplicit-function-declaration]
    24 |     portno = atoi(argv[1]);
      |              ^~~~~
server.c:36:9: warning: implicit declaration of function 'read'; did you mean 'fread'? [-Wimplicit-function-declaration]
    36 |     n = read(newsockfd, buffer, 255);
      |         ^~~~
server.c:40:9: warning: implicit declaration of function 'write'; did you mean 'fwrite'? [-Wimplicit-function-declaration]
    40 |     n = write(newsockfd, "I got your message", 18);
      |         ^~~~~
ling@Server:~/USC_EE533_lab1/source_code$
```

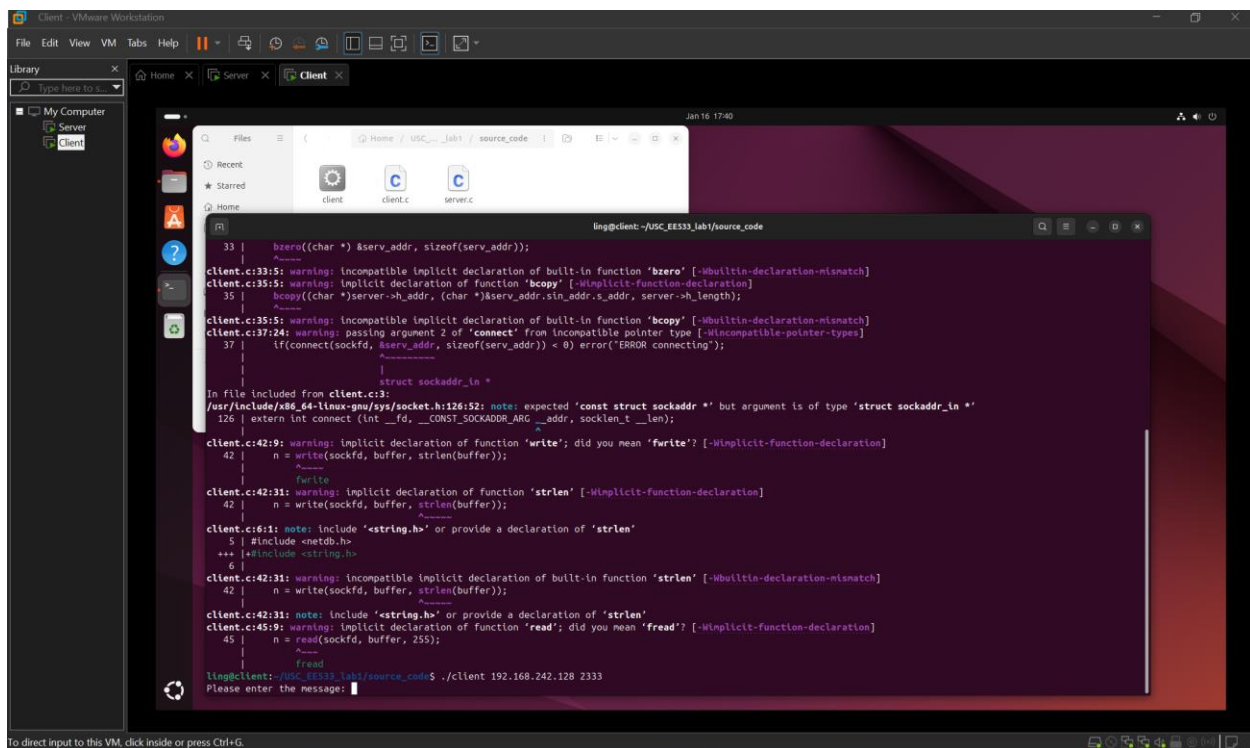
Here is a screenshot of the client code being compiled:

```
ling@Client:~/USC_EE533_lab1/source_code$ gcc -o client client.c
client.c:33:5: warning: implicit declaration of function 'bzero' [-Wimplicit-function-declaration]
    33 |     bzero((char *) &serv_addr, sizeof(serv_addr));
      |     ^~~~~
client.c:33:5: warning: incompatible implicit declaration of built-in function 'bzero' [-Wbuiltin-declaration-mismatch]
client.c:35:5: warning: implicit declaration of function 'bcopy' [-Wimplicit-function-declaration]
    35 |     bcopy((char *)server->h_addr, (char *)&serv_addr.sin_addr.s_addr, server->h_length);
      |     ^~~~~
client.c:35:5: warning: incompatible implicit declaration of built-in function 'bcopy' [-Wbuiltin-declaration-mismatch]
client.c:37:24: warning: passing argument 2 of 'connect' from incompatible pointer type [-Wincompatible-pointer-types]
    37 |     if(connect(sockfd, &serv_addr, sizeof(serv_addr)) < 0) error("ERROR connecting");
      |                    ^~~~~~
      |                    |
      |                    struct sockaddr_in *
In file included from client.c:3:
/usr/include/x86_64-linux-gnu/sys/socket.h:126:52: note: expected 'const struct sockaddr *' but argument is of type 'struct sockaddr_in *'
   126 | extern int connect(int __fd, __CONST_SOCKADDR_ARG __addr, socklen_t __len);
      |
client.c:42:9: warning: implicit declaration of function 'write'; did you mean 'fwrite'? [-Wimplicit-function-declaration]
    42 |     n = write(sockfd, buffer, strlen(buffer));
      |         ^~~~~
      |         |
      |         fwrite
client.c:42:31: warning: implicit declaration of function 'strlen' [-Wimplicit-function-declaration]
    42 |     n = write(sockfd, buffer, strlen(buffer));
      |                               ^~~~~~
client.c:6:1: note: include '<string.h>' or provide a declaration of 'strlen'
    6 | #include <string.h>
      | ^~~~~
client.c:42:31: warning: incompatible implicit declaration of built-in function 'strlen' [-Wbuiltin-declaration-mismatch]
    42 |     n = write(sockfd, buffer, strlen(buffer));
      |                               ^~~~~~
client.c:42:31: note: include '<string.h>' or provide a declaration of 'strlen'
client.c:45:9: warning: implicit declaration of function 'read'; did you mean 'fread'? [-Wimplicit-function-declaration]
    45 |     n = read(sockfd, buffer, 255);
      |         ^~~~
      |         |
      |         fread
ling@Client:~/USC_EE533_lab1/source_code$
```

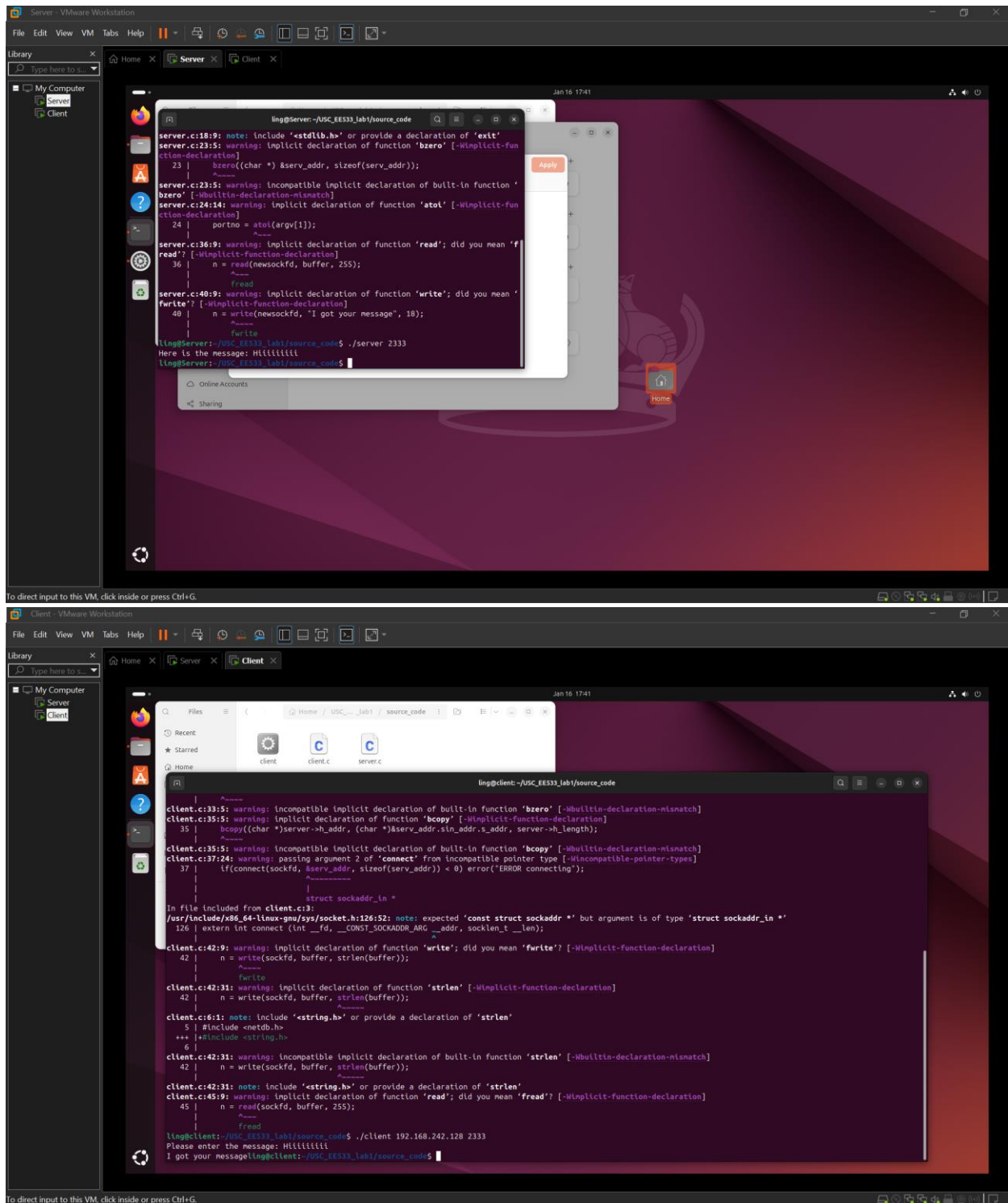
Here is a screenshot of the server program running:



We can see in the background that the address that the client will connect to is 192.168.242.128



Here is the screenshot of after we enter a message in client and that the server receives the messages:



Part4: Improvement to the server code:

Here is the server code after I made the suggestion in the pdf of adding fork and signal to prevents zombies:

```
#include <stdio.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <stdlib.h>
#include <signal.h>

void error(char *msg){
    perror(msg);
    exit(1);
}

void dostuff(int sockfd){
    char buffer[256];
    bzero(buffer, 256);
    int n = read(sockfd, buffer, 255);

    if (n < 0) error("ERROR reading from socket");

    printf("Here is the message: %s", buffer);

    n = write(sockfd, "I got your message", 18);
    if(n < 0) error("ERROR writing to socket");

    close(sockfd);
}

int main(int argc, char *argv[]){
    int sockfd, newsockfd, portno, clilen, n;
    char buffer[256];
    struct sockaddr_in serv_addr, cli_addr;

    if(argc < 2){
        fprintf(stderr, "ERROR, no port provided");
        exit(1);
    }

    signal(SIGCHLD, SIG_IGN); // zombie prevention
```

```

sockfd = socket(AF_INET, SOCK_STREAM, 0);
if(sockfd<0) error("ERROR opening socket");
bzero((char *) &serv_addr, sizeof(serv_addr));
portno = atoi(argv[1]);
serv_addr.sin_family = AF_INET;
serv_addr.sin_port = htons(portno);
serv_addr.sin_addr.s_addr = INADDR_ANY;
if(bind(sockfd, (struct sockaddr *) & serv_addr, sizeof(serv_addr)) < 0)
error("ERROR on binding");

listen(sockfd, 5);
clilen = sizeof(cli_addr);

// newsockfd = accept(sockfd, (struct sockaddr *) &cli_addr, &clilen);
// if(newsockfd < 0) error("ERROR on accept");

while (1){
    newsockfd = accept(sockfd, (struct sockaddr *) &cli_addr, &clilen);
    if (newsockfd < 0)
        error("ERROR on accept");

    pid_t pid = fork();
    if (pid < 0) error("ERROR on fork");

    if (pid == 0){
        close(sockfd);
        dostuff(newsockfd);
        exit(0);
    }else
        close(newsockfd);
}

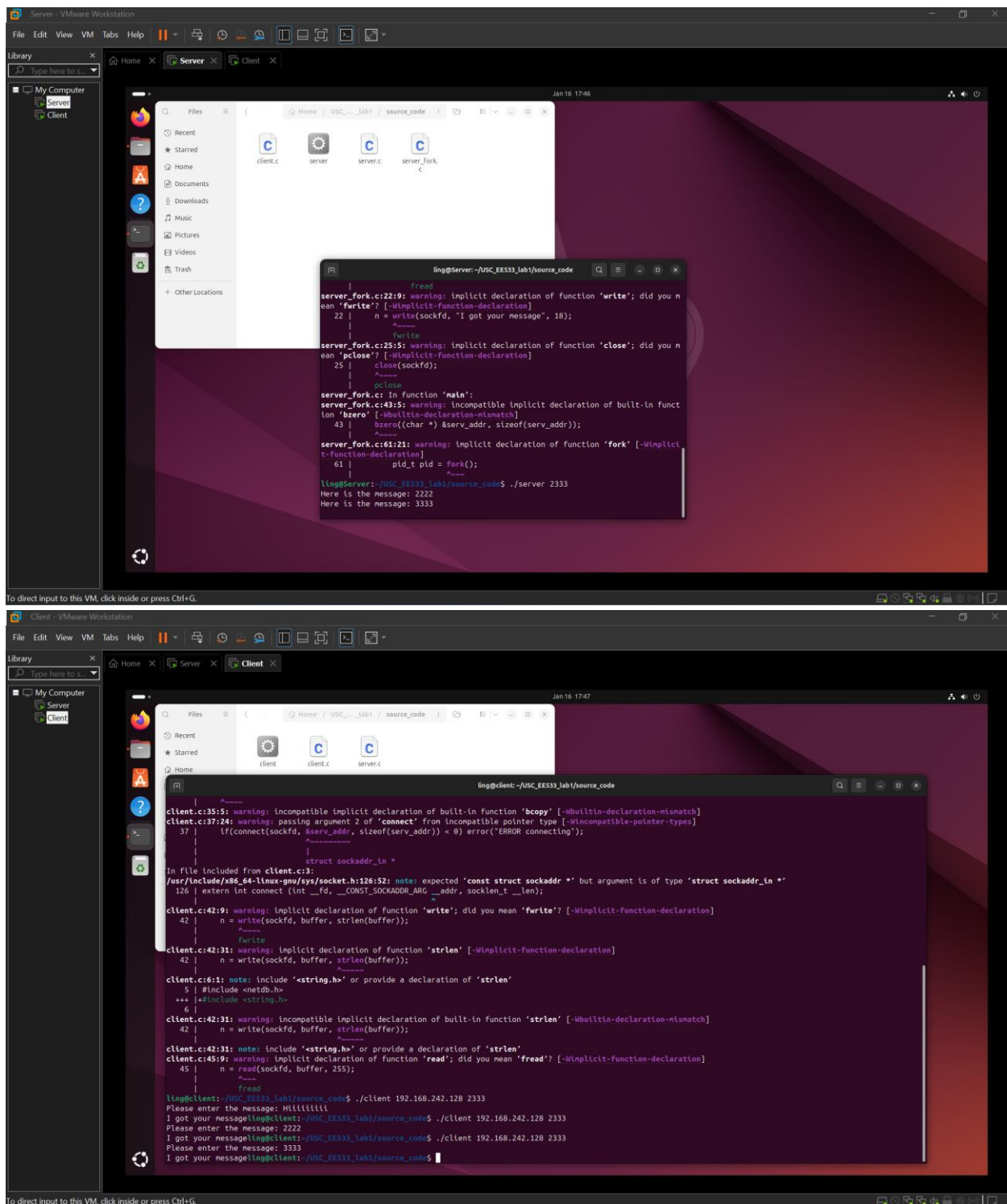
return 0;
}

```





















It is also available at:

https://github.com/yuezhenglingluan/USC_EE533_lab1/blob/main/source_code/server_fork.c

Here is a screenshot of the new server running and gets multiple responses to the client's messages:



Lastly, here is a screenshot of my commit history on github:

Commits on Jan 16, 2025		
reduce spacing between lines	yuezhenglingluan committed 1 hour ago	ef072ad  
tires to address zombie issues in the enhanced server code	yuezhenglingluan committed 1 hour ago	f038964  
fix the server code enhancement	yuezhenglingluan committed 1 hour ago	e99e077  
fix the server code enhancement	yuezhenglingluan committed 1 hour ago	c81d69f  
try to add enhancement to the server code	yuezhenglingluan committed 1 hour ago	87a8be8  
Commits on Jan 15, 2025		
fix typo in client.c	yuezhenglingluan committed yesterday	08f8289  
fix another typo in client.c	yuezhenglingluan committed yesterday	7cd8be0  
fix typo in client.c	yuezhenglingluan committed yesterday	d1d85dc  
upload the source code for a simple client	yuezhenglingluan committed yesterday	a12d7c9  
adds the source code for a simple server	yuezhenglingluan committed yesterday	7391ae8  

And here is the link to the project on github:

https://github.com/yuezhenglingluan/USC_EE533_lab1