Client program:

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

int main(int argc, char* argv[])
{
    int sfd, newsfd, s, len, n, i,j=0;
    char buf[21];
    struct sockaddr_in servaddr, serveracc;

    sfd = socket(AF_INET, SOCK_STREAM, 0);
    if(sfd<0)
    {
        perror("socket() error!");
    }
    servaddr.sin_family = AF_INET;
    servaddr.sin_port = htons(atoi(argv[1]));
    servaddr.sin_addr.s_addr = inet_addr("172.20.1.181");

    s = connect(sfd, (struct sockaddr*)&servaddr, sizeof(servaddr));

    if(s<0)
    {
        perror("connect() error!");

    }

    for(i = 1; i<=10; i++)
    {

        write(1, "\nYOUR MESSAGE: ",14);
        n = read(0, &buf, 100);
        send(sfd, &buf, n, 0);
        n = recv(sfd, &buf, 20,0);
        write(1, "RECEIVED FROM SERVER: ", 22);
        write(1, &buf, n);
    }
    close(newsfd);

    return 0;
}
```

Server program:

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

int main(int argc, char* argv[])
{
    int sfd, newsfd, s, len, n, i,j;
    char buf[100];
    struct sockaddr_in servaddr, clientacc, client;

    sfd = socket(AF_INET, SOCK_STREAM, 0);
    if(sfd<0)
    {
        perror("socket() error!");
        exit(-1);
    }
    servaddr.sin_family = AF_INET;
    servaddr.sin_port = htons(atoi(argv[1]));
    servaddr.sin_addr.s_addr = htonl(INADDR_ANY);

    s = bind(sfd, (struct sockaddr*)&servaddr, sizeof(servaddr));

    if(s<0)
    {
        perror("bind() error!");
        exit(-1);
    }

    s = listen(sfd, 5);

    if(s<0)
    {
        perror("listen() error!");
        exit(-1);
    }

    printf("\n---SERVER WAITING FOR CLIENT REQUEST---");
    newsfd = accept(sfd, (struct sockaddr*)&clientacc, &len);
    if(newsfd<0)
    {
        perror("accept() error!");
        exit(-1);
    }
```

```c
        printf("\n---CONNECTED TO CLIENT---");
        for(j = 1; j<=10; j++)
        {

                n = recv(newsfd, &buf, 100, 0);
                buf[n] = '\0';
                printf("\nMESSAGE FROM CLIENT: %s", buf);
                for(i = 0; buf[i]!='\0' ; i++)
                        buf[i] = toupper(buf[i]);
                send(newsfd, &buf, n, 0);
        }
        close(sfd);
        close(newsfd);

        return 0;
}
```
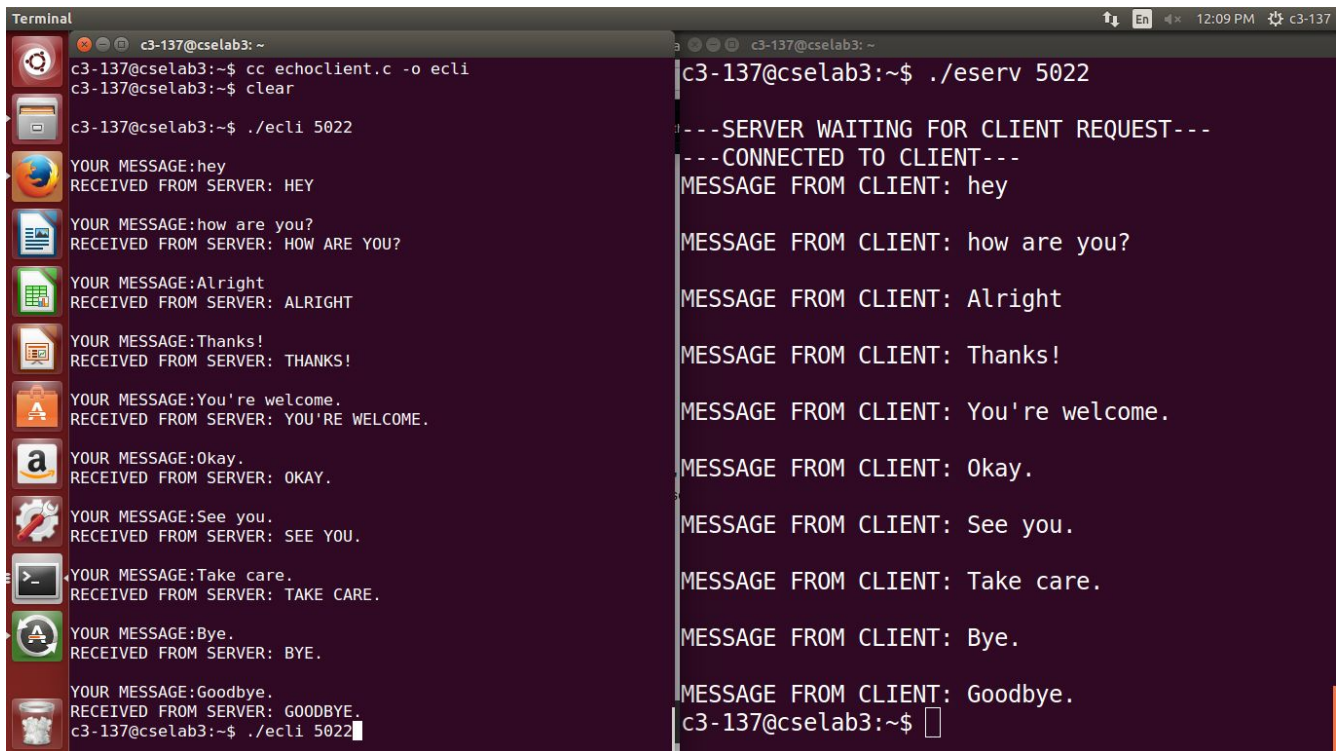
```
c3-138@cselab3-HP-ProDesk-400-G2-MT-TPM-DP:~$ cc socketsample.c -o ecli
c3-138@cselab3-HP-ProDesk-400-G2-MT-TPM-DP:~$ ./ecli 5007
enter a message:hey
received from server:HEY
enter a message:richa!!
received from server:RICHA!!
enter a message:how are you
received from server:HOW ARE YOU
enter a message:happy holidays
received from server:HAPPY HOLIDAYS
enter a message:okay byee
received from server:OKAY BYEE
c3-138@cselab3-HP-ProDesk-400-G2-MT-TPM-DP:~$
```

```
c3-137@cselab3:~$ ./eserv 5007

---SERVER WAITING FOR CLIENT REQUEST---
---CONNECTED TO CLIENT---
MESSAGE FROM CLIENT: hey

MESSAGE FROM CLIENT: richa!!

MESSAGE FROM CLIENT: how are you

MESSAGE FROM CLIENT: happy holidays

MESSAGE FROM CLIENT: okay byee

MESSAGE FROM CLIENT:
MESSAGE FROM CLIENT:
MESSAGE FROM CLIENT:
MESSAGE FROM CLIENT:
MESSAGE FROM CLIENT: c3-137@cselab3:~$
```