

## Assignment 3

Develop a socket program to simulate Echo Server.  
Echo Server.

The client sends data to server. The server in turn sends the message back to the client.

### **server.c**

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <arpa/inet.h>

void main(int argc, char **argv)
{
    int fd, newfd;
    struct sockaddr_in saddr, caddr;
    int addrlen = sizeof(saddr);
    char buff[1024];
    char str[1024];
    fd = socket(AF_INET, SOCK_STREAM, 0);
    if (fd < 0)
    {
        perror("Socket failure");
    }
    saddr.sin_family = AF_INET;
    saddr.sin_addr.s_addr = INADDR_ANY;
    saddr.sin_port = htons(7228);
    if (bind(fd, (struct sockaddr *) &saddr, sizeof(saddr)) < 0)
    {
        perror("Bind error");
    }
    listen(fd, 2);
    int len = sizeof(caddr);
    newfd = accept(fd, (struct sockaddr *) &caddr, &len);
    int n = read(newfd, buff, sizeof(buff));
    printf("Message recieved from client : %s\n", buff);
    if (send(newfd, buff, n, 0) > 0)
    {
        printf("Message sent to client : %s\n", buff);
    }
}
```

```

    }
    close(fd);
    close(newfd);
}

```

### **client.c**

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <arpa/inet.h>

void main(int argc, char **argv)
{
    int fd;
    struct sockaddr_in saddr, caddr;
    int addrlen = sizeof(saddr);
    char buff[1024];
    char str[1024];
    fd = socket(AF_INET, SOCK_STREAM, 0);
    if (fd < 0)
    {
        perror("Socket failure");
    }
    bzero(&saddr, sizeof(saddr));
    saddr.sin_family = AF_INET;
    saddr.sin_addr.s_addr = inet_addr(argv[1]);
    saddr.sin_port = htons(7228);
    connect(fd, (struct sockaddr *)&saddr, sizeof(saddr));
    printf("Enter the message : ");
    scanf("%[^\n]s", buff);
    int n = write(fd, buff, sizeof(buff));
    n = read(fd, buff, 1024);
    buff[n] = '\0';
    printf("Echoed from server : %s\n", buff);
    close(fd);
}

```

## Output:

```
UGB2@ssn-22: ~/Downloads
UGB2@ssn-22:~/Downloads$ ./eserver
Message recieved from client : Hello, this is client.
Message sent to client : Hello, this is client.
UGB2@ssn-22:~/Downloads$
```

```
UGB2@ssn-22: ~/Downloads
UGB2@ssn-22:~/Downloads$ ./eclient 10.6.15.22
Enter the message : Hello, this is client.
Echoed from server : Hello, this is client.
UGB2@ssn-22:~/Downloads$
```