



CN LAB-3

NAME: Ankit Raj

ROLL NO: 1906534

1. Write a client server program where the client sends and the server accepts it and prints it till exit is not sent.

Server.c

```
#include <arpa/inet.h>
#include <netinet/in.h>
#include <stdio.h>
#include <string.h>
#include <sys/socket.h>
#include <sys/stat.h>
#include <unistd.h>
#include <sys/types.h>
int main()
{
    struct sockaddr_in client, server;
    int s, n, sock, flag_534=0;
    char b1_534[20];
    s = socket(AF_INET, SOCK_STREAM, 0);
    server.sin_family = AF_INET;
    server.sin_port = 2000;
    server.sin_addr.s_addr = inet_addr("127.0.0.1");
    bind(s, (struct sockaddr*)&server, sizeof server);
    listen(s, 1);
    n = sizeof client;
    sock = accept(s, (struct sockaddr*)&client, &n);
    while(flag_534==0)
    {
        recv(sock, b1_534, sizeof(b1_534), 0);
        if(strcmp(b1_534,"exit")==0)
            flag_534=1;
        printf("\nThe string received is:%s\n", b1_534);
    }
    close(sock);
    close(s);
}
```

Client.c

```
#include <arpa/inet.h>
#include <netinet/in.h>
#include <stdio.h>
#include <string.h>
#include <sys/socket.h>
#include <unistd.h>
#include <sys/stat.h>
#include <sys/types.h>
int main()
{
    struct sockaddr_in client;
    int s, flag_534;
    char buffer_534[20];
    s = socket(AF_INET, SOCK_STREAM, 0);
    client.sin_family = AF_INET;
    client.sin_port = 2000;
    client.sin_addr.s_addr = inet_addr("127.0.0.1");
    connect(s, (struct sockaddr*)&client, sizeof client);
    for (;;) {
        printf("\nEnter a string: ");
        scanf("%s", buffer_534);
```

```

send(s, buffer_534, sizeof(buffer_534), 0);
if(strcmp(buffer_534,"exit")==0)
break;
}
close(s);
}

```

OUTPUT:

The first screenshot shows the compilation of the server program and its execution. The user runs `gcc server.c -o server` and then `./server`. The server outputs "test 30", "hi", "hello", and "exit".

The second screenshot shows the compilation of the client program and its execution. The user runs `gcc client.c -o client` and then `./client`. The client outputs "Response from server" and then prompts the user to enter a message to send. The user enters "hi", "hello", and "exit".

2. Write a client server program to reverse a number which client sends to the server

Server.c

```

#include <arpa/inet.h>
#include <netinet/in.h>
#include<unistd.h>
#include <stdio.h>
#include <string.h>
#include<stdlib.h>
#include <sys/socket.h>
#include <sys/stat.h>
#include <sys/types.h>
int main()
{
struct sockaddr_in client, server;
int s, n, sock;
char a_534[100],*b;
s = socket(AF_INET, SOCK_STREAM, 0);
server.sin_family = AF_INET;
server.sin_port = 2000;
server.sin_addr.s_addr = inet_addr("127.0.0.1");
bind(s, (struct sockaddr*)&server, sizeof server);
listen(s, 1);
n = sizeof client;
sock = accept(s, (struct sockaddr*)&client, &n);
recv(s, a_534, sizeof(a_534), 0);
send(s,a_534,sizeof(a_534),0);
close(sock);
close(s);
}

```

Client.c

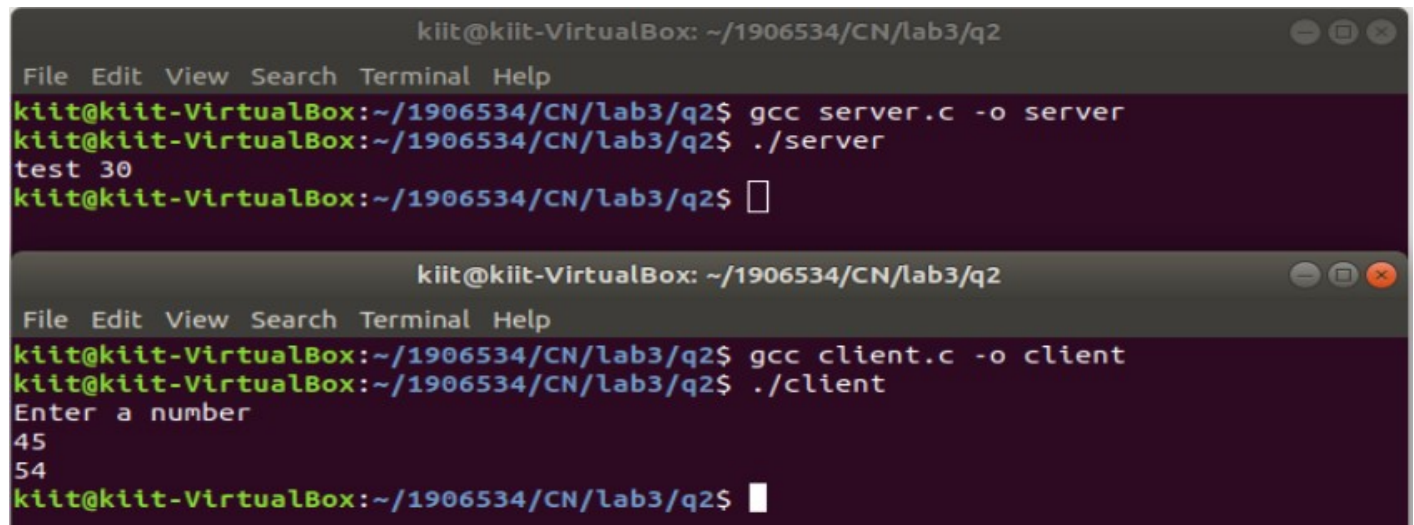
```

#include <arpa/inet.h>
#include <netinet/in.h>
#include <stdio.h>
#include <string.h>
#include <sys/socket.h>
#include<stdlib.h>
#include<unistd.h>
#include <sys/stat.h>
#include <sys/types.h>

```

```
int main()
{
    struct sockaddr_in client;
    int l_534,s_534, flag_534;
    char buffer_534[20];
    s_534 = socket(AF_INET, SOCK_STREAM, 0);
    client.sin_family = AF_INET;client.sin_port = 2000;
    client.sin_addr.s_addr = inet_addr("127.0.0.1");
    connect(s_534, (struct sockaddr*)&client, sizeof client);
    scanf("%s_534",buffer_534);
    send(s_534,buffer_534,sizeof(buffer_534),0);
    recv(s_534,buffer_534,sizeof(buffer_534),0);
    l_534=strlen(buffer_534);
    for(int i=l_534-1;i>=0;i--)
    printf("%c",buffer_534[i]);
    close(s_534);
}
```

OUTPUT:



```
kiit@kiit-VirtualBox: ~/1906534/CN/lab3/q2
File Edit View Search Terminal Help
kiit@kiit-VirtualBox:~/1906534/CN/lab3/q2$ gcc server.c -o server
kiit@kiit-VirtualBox:~/1906534/CN/lab3/q2$ ./server
test 30
kiit@kiit-VirtualBox:~/1906534/CN/lab3/q2$ □

kiit@kiit-VirtualBox: ~/1906534/CN/lab3/q2
File Edit View Search Terminal Help
kiit@kiit-VirtualBox:~/1906534/CN/lab3/q2$ gcc client.c -o client
kiit@kiit-VirtualBox:~/1906534/CN/lab3/q2$ ./client
Enter a number
45
54
kiit@kiit-VirtualBox:~/1906534/CN/lab3/q2$ █
```