



CN LAB-4.8.21

NAME: Ankit Raj

ROLL NO: 1906534

1. TCP SOCKET CLIENT WILL SEND AN INTEGER ARRAY, AND RECEIVE SORTED ARRAY SERVER.

Server.c

```
#include<stdio.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<arpa/inet.h>
#include<fcntl.h>
#include<string.h>
#include<unistd.h>
int main()
{
    int sockfd_534,fd1_534, length_534,i_534,buf_534[100],n_534;
    struct sockaddr_in sa_addr_534,cl_addr_534;
    sockfd_534=socket(AF_INET,SOCK_STREAM,0);
    sa_addr_534.sin_family=AF_INET;
    sa_addr_534.sin_addr.s_addr=INADDR_ANY;
    sa_addr_534.sin_port=htons(6000);
    i_534=bind(sockfd_534,(struct sockaddr *)&sa_addr_534,sizeof(sa_addr_534));
    printf("test %d%d\n",sockfd_534,i_534);
    listen(sockfd_534,5);
    length_534=sizeof(cl_addr_534);
    fd1_534=accept(sockfd_534, (struct sockaddr *) &cl_addr_534,&length_534);
    for(i_534=0; i_534 < 100; i_534++) buf_534[i_534] = 0;
    recv(fd1_534, buf_534, 100, 0);
    recv(fd1_534, &n_534, sizeof(n_534), 0);
    for(int i_534=0; i_534<n_534-1; i_534++)
    {
        for(int j_534=0; j_534<n_534-i_534-1; j_534++)
        {
            if(buf_534[j_534]>buf_534[j_534+1])
            {
                int temp_534=buf_534[j_534];
                buf_534[j_534]=buf_534[j_534+1];
                buf_534[j_534+1]=temp_534;
            }
        }
    }
    send(fd1_534, buf_534, 100, 0);
    close(fd1_534);
}
```

Client.c

```
#include<stdio.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<arpa/inet.h>
#include<fcntl.h>
#include<string.h>
#include<unistd.h>
int main()
{
    int i_534,sockfd_534,buf_534[100],n_534;
    struct sockaddr_in sa_addr_534;
    sockfd_534=socket(AF_INET,SOCK_STREAM,0);
    sa_addr_534.sin_family=AF_INET;
```

```

sa_addr_534.sin_addr.s_addr=inet_addr("127.0.0.1"); //Loop back IP address
sa_addr_534.sin_port=htons(6000);
memset(sa_addr_534.sin_zero, '\0', sizeof sa_addr_534.sin_zero);
i_534=connect(sockfd_534,(struct sockaddr *)&sa_addr_534,sizeof(sa_addr_534));
printf("utkarsh , Enter the size of the array\n");
scanf("%d",&n_534);
printf("utkarsh , Enter the input into the array\n");
for(int i_534=0; i_534<n_534; i_534++)
{
scanf("%d",&buf_534[i_534]);
}
send(sockfd_534, buf_534, 100, 0);
send(sockfd_534, &n_534, sizeof(n_534),0);
recv(sockfd_534, buf_534, 100, 0);
for(int i_534=0; i_534<n_534; i_534++)
{
printf("%d ",buf_534[i_534]);
}
printf("\n");
close(sockfd_534);
}

```

OUTPUT:

```

kiit@kiit-VirtualBox: ~/1906534/CN/lab3/q3
File Edit View Search Terminal Help
kiit@kiit-VirtualBox:~$ cd 1906534/CN/lab3
kiit@kiit-VirtualBox:~/1906534/CN/lab3$ cd q3
kiit@kiit-VirtualBox:~/1906534/CN/lab3/q3$ ./server
test 30
kiit@kiit-VirtualBox:~/1906534/CN/lab3/q3$ 

kiit@kiit-VirtualBox: ~/1906534/CN/lab3/q3
File Edit View Search Terminal Help
kiit@kiit-VirtualBox:~$ cd 1906534/CN/lab3
kiit@kiit-VirtualBox:~/1906534/CN/lab3$ cd q3
kiit@kiit-VirtualBox:~/1906534/CN/lab3/q3$ ./client
Enter the size of the array
5
Enter the input into the array
34 12 87 56 9
9 12 34 56 87
kiit@kiit-VirtualBox:~/1906534/CN/lab3/q3$ 

```

2. TCP SOCKET SERVER DEAL WITH TWO CLIENT TO EXCHANGE NUMBER

Server.c

```

#include<stdio.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<arpa/inet.h>
#include<fcntl.h>
#include<string.h>
#include<unistd.h>
int main()
{
int sockfd_534,sockfd1_534,fd1_534,fd2_534, length_534,length1_534,i_534,j_534;
int buf_534,buf1_534;
struct sockaddr_in sa_addr1_534,cl_addr1_534;
struct sockaddr_in sa_addr_534,cl_addr_534;
sockfd_534=socket(AF_INET,SOCK_STREAM,0);
sockfd1_534=socket(AF_INET,SOCK_STREAM,0);
sa_addr_534.sin_family=AF_INET;
sa_addr1_534.sin_family=AF_INET;
sa_addr_534.sin_addr.s_addr=INADDR_ANY;
sa_addr1_534.sin_addr.s_addr=INADDR_ANY;
sa_addr_534.sin_port=htons(3000);
sa_addr1_534.sin_port=htons(2995);
memset(sa_addr_534.sin_zero, '\0', sizeof sa_addr_534.sin_zero);
memset(sa_addr1_534.sin_zero, '\0', sizeof sa_addr1_534.sin_zero);
i_534=bind(sockfd_534,(struct sockaddr *)&sa_addr_534,sizeof(sa_addr_534));
j_534=bind(sockfd1_534,(struct sockaddr *)&sa_addr1_534,sizeof(sa_addr1_534));
printf("test %d\n",sockfd_534,i_534);
printf("test %d\n",sockfd1_534,j_534);

```

```

listen(sockfd_534,5);
listen(sockfd1_534,5);

length_534=sizeof(cl_addr_534);
length_534=sizeof(cl_addr1_534);
fd1_534=accept(sockfd_534, (struct sockaddr *) &cl_addr_534,&length_534);
fd2_534=accept(sockfd1_534, (struct sockaddr *) &cl_addr1_534,&length1_534);
recv(fd1_534, &buf_534, sizeof(buf_534),0);
printf("%d",buf_534);
recv(fd2_534, &buf1_534,sizeof(buf1_534),0);
printf("%d",buf1_534);
send(fd1_534, &buf1_534, sizeof(buf_534),0);
send(fd2_534, &buf_534,sizeof(buf1_534),0);
close(fd1_534);
close(fd2_534);
}

```

Client1.c

```

#include<stdio.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<arpa/inet.h>
#include<fcntl.h>
#include<string.h>
#include<unistd.h>
int main()
{
    int i_534,sockfd_534;
    int buf_534;
    struct sockaddr_in sa_addr_534;
    sockfd_534=socket(AF_INET,SOCK_STREAM,0);
    sa_addr_534.sin_family=AF_INET;
    sa_addr_534.sin_addr.s_addr=inet_addr("127.0.0.1");
    sa_addr_534.sin_port=htons(2995);
    memset(sa_addr_534.sin_zero, '\0', sizeof sa_addr_534.sin_zero);
    i_534=connect(sockfd_534,(struct sockaddr *)&sa_addr_534,sizeof(sa_addr_534));
    printf("Enter the number\n");
    scanf("%d",&buf_534);
    send(sockfd_534, &buf_534, sizeof(buf_534), 0);
    recv(sockfd_534, &buf_534, sizeof(buf_534), 0);
    printf("%d\n", buf_534);
    close(sockfd_534);
}

```

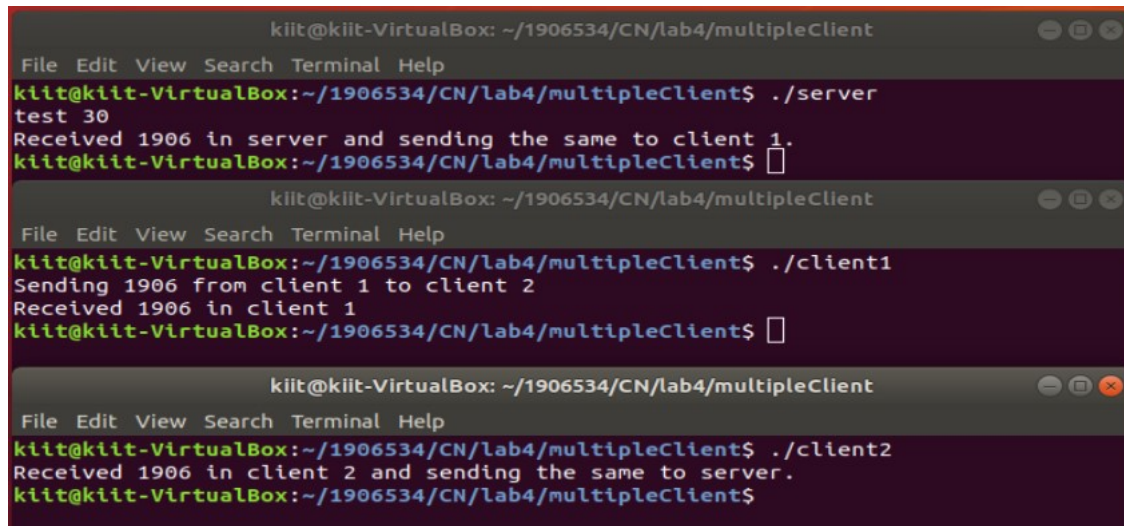
Client2.c

```

#include<stdio.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<arpa/inet.h>
#include<fcntl.h>
#include<string.h>
#include<unistd.h>
int main()
{
    int i_534,sockfd_534;
    int buf_534;
    struct sockaddr_in sa_addr_534;
    sockfd_534=socket(AF_INET,SOCK_STREAM,0);
    sa_addr_534.sin_family=AF_INET;
    sa_addr_534.sin_addr.s_addr=inet_addr("127.0.0.1");
    sa_addr_534.sin_port=htons(3000);
    memset(sa_addr_534.sin_zero, '\0', sizeof sa_addr_534.sin_zero);
    i_534=connect(sockfd_534,(struct sockaddr *)&sa_addr_534,sizeof(sa_addr_534));
    printf("Enter the number\n");
    scanf("%d",&buf_534);
    send(sockfd_534, &buf_534, sizeof(buf_534), 0);
    recv(sockfd_534, &buf_534, sizeof(buf_534), 0);
    printf("%d\n", buf_534);
    close(sockfd_534);
}

```

OUTPUT:



The image shows three terminal windows from a VirtualBox environment. The first window shows the server program being executed with the command `./server`, which receives a value of 1906 and sends it to client 1. The second window shows client 1 being executed with `./client1`, which sends 1906 to client 2 and receives 1906 back. The third window shows client 2 being executed with `./client2`, which receives 1906 from client 2 and sends it back to the server.

```
kiit@kiit-VirtualBox: ~/1906534/CN/lab4/multipleClient
File Edit View Search Terminal Help
kiit@kiit-VirtualBox:~/1906534/CN/lab4/multipleClient$ ./server
test 30
Received 1906 in server and sending the same to client 1.
kiit@kiit-VirtualBox:~/1906534/CN/lab4/multipleClient$

kiit@kiit-VirtualBox: ~/1906534/CN/lab4/multipleClient
File Edit View Search Terminal Help
kiit@kiit-VirtualBox:~/1906534/CN/lab4/multipleClient$ ./client1
Sending 1906 from client 1 to client 2
Received 1906 in client 1
kiit@kiit-VirtualBox:~/1906534/CN/lab4/multipleClient$

kiit@kiit-VirtualBox: ~/1906534/CN/lab4/multipleClient
File Edit View Search Terminal Help
kiit@kiit-VirtualBox:~/1906534/CN/lab4/multipleClient$ ./client2
Received 1906 in client 2 and sending the same to server.
kiit@kiit-VirtualBox:~/1906534/CN/lab4/multipleClient$
```

3. TCP SOCKET USING STRUCTURE SEND THE DETAILS TO THE CLIENT, AFTER RECEIVING ROLL NUMBER

Server.c

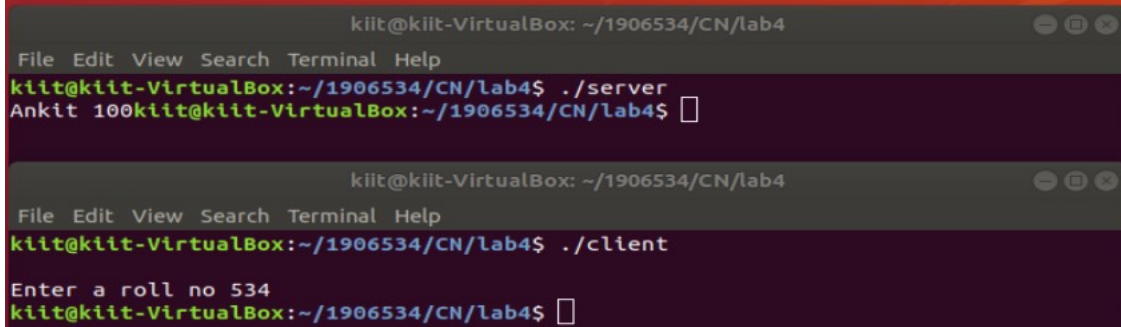
```
#include <arpa/inet.h>
#include <netinet/in.h>
#include <stdio.h>
#include <string.h>
#include <sys/socket.h>
#include <sys/stat.h>
#include <unistd.h>
#include <sys/types.h>
struct student
{
    int roll;
    char name[50];
    int marks;
};
int main()
{
    struct sockaddr_in client, server;
    int s, n, sock, flag_534=0;
    int b1_534;
    struct student a[2];
    a[0].roll=534;
    strcpy(a[0].name,"Ankit");
    a[0].marks=100;
    a[1].roll=535;
    strcpy(a[1].name,"Raj");
    a[1].marks=95;
    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(sock, &b1_534, sizeof(b1_534), 0);
    int k=0,i=0;
    for(;i<2;i++)
    {
        if(a[i].roll==b1_534)
        {
            k=1;
            break;
        }
    }
    if(k==1)
        printf("%s %d",a[i].name,a[i].marks);
    else
        printf("\nNot found");
}
```

```
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;  
    int buffer_534;  
    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);  
    printf("\nEnter a roll no\t");  
    scanf("%d", &buffer_534);  
    send(s, &buffer_534, sizeof(buffer_534), 0);  
    close(s);  
}
```

OUTPUT:



The image shows two terminal windows from a VirtualBox environment. The first window shows the execution of a server program. The user runs `./server` and the prompt changes to `Ankit 100`. The second window shows the execution of a client program. The user runs `./client` and is prompted with `Enter a roll no 534`. The prompt then changes to `kiit@kiit-VirtualBox:~/1906534/CN/lab4$`.

```
kiit@kiit-VirtualBox: ~/1906534/CN/lab4  
File Edit View Search Terminal Help  
kiit@kiit-VirtualBox:~/1906534/CN/lab4$ ./server  
Ankit 100kiit@kiit-VirtualBox:~/1906534/CN/lab4$  
  
kiit@kiit-VirtualBox: ~/1906534/CN/lab4  
File Edit View Search Terminal Help  
kiit@kiit-VirtualBox:~/1906534/CN/lab4$ ./client  
Enter a roll no 534  
kiit@kiit-VirtualBox:~/1906534/CN/lab4$
```