client1.c

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>


void main()
{
	int client_sock=socket(AF_INET,SOCK_DGRAM,0);
	if(client_sock<0)
	{
		printf("Socket Creation Failed..!!!\n");
		return;
	}
	struct sockaddr_in server_addr;
	server_addr.sin_family=AF_INET;
	server_addr.sin_port=htons(8000);
	server_addr.sin_addr.s_addr=inet_addr("127.0.0.1");

	char buffer[50];
	printf("Enter the String:");
	scanf("%s",buffer);

	 if(sendto(client_sock,buffer,sizeof(buffer),0,(struct
sockaddr*)&server_addr,sizeof(server_addr))<0)
	{
		printf("Message Sending Failed..!!\n");
		close(client_sock);
		return;
	}

	printf("Message Sent to server...\n");
	close(client_sock);
	return;
}
```

client2.c

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>
```

```
void main()
{
	int client_sock=socket(AF_INET,SOCK_DGRAM,0);
	if(client_sock<0)
	{
		printf("Scokket creation Failed...!!!\n");
		return;
	}
	struct sockaddr_in server_addr;
	server_addr.sin_family=AF_INET;
	server_addr.sin_port=htons(8000);
	server_addr.sin_addr.s_addr=inet_addr("127.0.0.1");

	char buffer[50]="Client 2 Ready";
	if(sendto(client_sock,buffer,sizeof(buffer),0,(struct
sockaddr*)&server_addr,sizeof(server_addr))<0)
	{
		printf("Message Sending Failed..!!\n");
		close(client_sock);
		return;
	}
	int addrlen=sizeof(server_addr);
	printf("Waiting for message from server...\n");
	 if(recvfrom(client_sock,buffer,sizeof(buffer),0,(struct
sockaddr*)&server_addr,&addrlen)<0)
	{
		printf("Message Reception Failed..!!\n");
		close(client_sock);
		return;
	}
	printf("Reversed String: %s\n",buffer);
	close(client_sock);
	return;
}
```

server.c

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>
#include<string.h>

void main()
{
	int udp_sock=socket(AF_INET,SOCK_DGRAM,0);
	if(udp_sock<0)
	{
		printf("Socket Creation Failed...!!!\n");
```

```
        return;
}

struct sockaddr_in server_addr;
server_addr.sin_family=AF_INET;
server_addr.sin_port=htons(8000);
server_addr.sin_addr.s_addr=inet_addr("127.0.0.1");

 if(bind(udp_sock,(struct    sockaddr*)&server_addr,sizeof(server_addr))<0)
{
        printf("Binding Failed...!!!\n");
        close(udp_sock);
        return;
}

char string[50];

printf("Waiting for message... \n");

struct sockaddr_in client_addr;
int addrlen=sizeof(client_addr);
 if(recvfrom(udp_sock,string,sizeof(string),0,(struct    sockaddr*)&client_addr,&addrlen)<0)
{
        printf("Failed to receive message..!!\n");
        close(udp_sock);
        return;
}
char rev[50];
printf("Received String: %s\n",string);
for (int i=0,j=strlen(string)-1;j>=0;j--,i++)
{
        rev[i]=string[j];
}
rev[strlen(rev)]='\0';

 if(recvfrom(udp_sock,string,sizeof(string),0,(struct    sockaddr*)&client_addr,&addrlen)<0)
{
        printf("Failed to receive message..!!\n");
        close(udp_sock);
        return;
}
printf("%s\n",string);
 if(sendto(udp_sock,rev,sizeof(rev),0,(struct    sockaddr*)&client_addr,sizeof(client_addr))<0)
{
        printf("Send Failed!!!\n");
        close(udp_sock);
        return;
}

printf("Message Sent to client 2!\n");

close(udp_sock);
```

```
        return;
}
```

OUTPUT:

./server

Waiting for message....
Received String: abcd
Client 2 Ready
Message Sent to client 2!

./client1

Enter the String:abcd
Message Sent to server...

./client2
Waiting for message from server...
Reversed String: dcba

client1_tcp

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>

void main()
{
        int tcp_client1=socket(AF_INET,SOCK_STREAM,0);
        if(tcp_client1<0)
        {
                printf("Socket Creation Failed!!!\n");
                return;
        }

        struct sockaddr_in caddr;
        caddr.sin_family=AF_INET;
        caddr.sin_port=htons(8080);
        caddr.sin_addr.s_addr=inet_addr("127.0.0.1");

         if(connect(tcp_client1,(struct   sockaddr*)&caddr,sizeof(caddr))<0)\
        {
                printf("Connection Failed..!!\n");
                close(tcp_client1);
                return;
        }

        char buffer[50];
```

```
	printf("Enter the String:");
	scanf("%s",buffer);

	 if(send(tcp_client1,buffer,sizeof(buffer),0)<0)
	{
		printf("Message Sending Failed..!!\n");
		close(tcp_client1);
		return;
	}
	printf("Message sent..!!!\n");
	close(tcp_client1);
	return;
}
```

client2_tcp

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>

void main()
{
	int tcp_client1=socket(AF_INET,SOCK_STREAM,0);
	if(tcp_client1<0)
	{
		printf("Socket Creation Failed!!!\n");
		return;
	}

	struct sockaddr_in caddr;
	caddr.sin_family=AF_INET;
	caddr.sin_port=htons(8080);
	caddr.sin_addr.s_addr=inet_addr("127.0.0.1");

	 if(connect(tcp_client1,(struct   sockaddr*)&caddr,sizeof(caddr))<0)\
	{
		printf("Connection Failed..!!\n");
		close(tcp_client1);
		return;
	}

	char buffer[50];

	if(recv(tcp_client1,buffer,sizeof(buffer),0)<0)
	{
		printf("Message Reception Failed..!!\n");
		close(tcp_client1);
		return;
	}
```

```
        printf("Message Received: %s\n",buffer);
        close(tcp_client1);
        return;
}
```

server_tcp

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>
#include<string.h>


void main()
{
        int tcp_server=socket(AF_INET,SOCK_STREAM,0);
        if(tcp_server<0)
        {
                printf("Socket Creation Failed!!!\n");
                return;
        }

        struct sockaddr_in server_addr;
        server_addr.sin_family=AF_INET;
        server_addr.sin_port=htons(8080);
        server_addr.sin_addr.s_addr=inet_addr("127.0.0.1");

        if(bind(tcp_server,(struct    sockaddr*)&server_addr,sizeof(server_addr))<0)
        {
                printf("Bining Failed!!!\n");
                close(tcp_server);
                return;
        }
        printf("Binding Successfull...\n");

        if(listen(tcp_server,5)<0)
        {
                printf("Listen Failed!!!\n");
                close(tcp_server);
                return;
        }
        printf("Listening...\n");

        struct sockaddr_in caddr;
        int isock,clen=sizeof(caddr);

        if ((isock=accept(tcp_server,(struct sockaddr*)&caddr,&clen))<0)
        {
                printf("Failed to establish connection!!!");
                close(tcp_server);
```

```
                return;
        }
        printf("connected to client 1...");
        char buffer[50];
         if (recv(isock,buffer,sizeof(buffer),0)<0)
        {
                printf("Failed to Recevie!!!\n");
                close(isock);
                close(tcp_server);
                return;
        }

        char rev[50];
        printf("Received String: %s\n",buffer);
        for (int i=0,j=strlen(buffer)-1;j>=0;j--,i++)
        {
                rev[i]=buffer[j];
        }
        rev[strlen(rev)]='\0';

         if ((isock=accept(tcp_server,(struct sockaddr*)&caddr,&clen))<0)
        {
                printf("Failed to establish connection!!!");
                close(tcp_server);
                return;
        }
        printf("connected to client 2...");
         if (send(isock,rev,sizeof(rev),0)<0)
        {
                printf("Failed to Recevie!!!\n");
                close(isock);
                close(tcp_server);
                return;
        }
        printf("Message Sent to client 2!\n");
        close(isock);
        close(tcp_server);
        return;
}
```

OUTPUT:

```
./server

Binding Successfull...
Listening...
connected to client 1...
Received String: abcd
connected to client 2...
Message Sent to client 2!
```

./client1

Enter the String:abcd
Message Sent to server...

./client2
Waiting for message from server...
Reversed String: dcba

client1.c

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>


void main()
{
        int client_sock=socket(AF_INET,SOCK_DGRAM,0);
        if(client_sock<0)
        {
                printf("Socket Creation Failed..!!!\n");
                return;
        }
        struct sockaddr_in server_addr;
        server_addr.sin_family=AF_INET;
        server_addr.sin_port=htons(8000);
        server_addr.sin_addr.s_addr=inet_addr("127.0.0.1");

        int num=10;

        if(sendto(client_sock,&num,sizeof(num),0,(struct
sockaddr*)&server_addr,sizeof(server_addr))<0)
        {
                printf("Message Sending Failed..!!\n");
                close(client_sock);
                return;
        }

        printf("Message Sent to server...\n");
        close(client_sock);
        return;
}
```

client2.c

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>

void main()
{
        int
        client_sock=socket(AF_INET,SOCK_DGRAM,0)
        ;  if(client_sock<0)
        {
                printf("Scokket creation
                Failed...!!!\n");  return;
```

```
        }
        struct sockaddr_in server_addr;
        server_addr.sin_family=AF_INET;
        server_addr.sin_port=htons(8000);
        server_addr.sin_addr.s_addr=inet_addr("127.0.0.1
        ");

        char buffer[50]="Client 2 Ready";
        if(sendto(client_sock,buffer,sizeof(buffer),0,(struct
sockaddr*)&server_addr,sizeof(server_addr))<0)
        {
                printf("Message Sending Failed..!!\n");
                close(client_sock);
                return;
        }
        int addrlen=sizeof(server_addr);
        printf("Waiting for message from
        server...\n");  int num;
         if(recvfrom(client_sock,&num,sizeof(int),0,(struct
        sockaddr*)&server_addr,&addrlen)<0)
        {
                printf("Message Reception Failed..!!\n");
                close(client_sock);
                return;
        }
        printf("Squared: %d\n",num);
        close(client_sock);
        return;
}
```

server.c

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>
#include<math.h>

void main()
{
        int
        udp_sock=socket(AF_INET,SOCK_DGRAM,0);
        if(udp_sock<0)
        {
                printf("Socket Creation
                Failed...!!!\n");  return;
        }

        struct sockaddr_in server_addr;
        server_addr.sin_family=AF_INET;
        server_addr.sin_port=htons(8000);
        server_addr.sin_addr.s_addr=inet_addr("127.0.0.1
        ");
```

```
        if(bind(udp_sock,(struct    sockaddr*)&server_addr,sizeof(server_addr))<0)
        {
                printf("Binding Failed...!!!\n");
                close(udp_sock);
                return;
        }


        printf("Waiting for message... \n");

        struct sockaddr_in
        client_addr;  int
        addrlen=sizeof(client_addr);
    int num;
        if(recvfrom(udp_sock,&num,sizeof(int),0,(struct   sockaddr*)&client_addr,&addrlen)<0)
        {
                printf("Failed to receive message..!!\n");
                close(udp_sock);
                return;
        }
        printf("Received Number: %d\n",num);
        num=pow(num,2);
    char string[50]="";
        if(recvfrom(udp_sock,string,sizeof(string),0,(struct
        sockaddr*)&client_addr,&addrlen)<0)
        {
                printf("Failed to receive message..!!\n");
                close(udp_sock);
                return;
        }
        printf("%s\n",string);
        if(sendto(udp_sock,&num,sizeof(int),0,(struct
sockaddr*)&client_addr,sizeof(client_addr))<0)
        {
                printf("Send Failed!!!\n");
                close(udp_sock);
                return;
        }

        printf("Message Sent to client 2!\n");

        close(udp_sock);
        return;
}
```

OUTPUT:

./server

Waiting for message....
Received Number: 10
Client 2 Ready
Message Sent to client 2!

./client1

Message Sent to server...

./client2

Waiting for message from server...
Squared: 100

client1_tcp.c

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>

void main()
{
        int tcp_client1=socket(AF_INET,SOCK_STREAM,0);
        if(tcp_client1<0)
        {
                printf("Socket Creation Failed!!!\n");
                return;
        }

        struct sockaddr_in caddr;
        caddr.sin_family=AF_INET;
        caddr.sin_port=htons(8080);
        caddr.sin_addr.s_addr=inet_addr("127.0.0.1");

         if(connect(tcp_client1,(struct   sockaddr*)&caddr,sizeof(caddr))<0)
        {
                printf("Connection Failed..!!\n");
                close(tcp_client1);
                return;
        }

        int num=10;

        if(send(tcp_client1,&num,sizeof(int),0)<0)
        {
                printf("Message Sending Failed..!!\n");
                close(tcp_client1);
                return;
```

```
        }
        printf("Message sent..!!!\n");
        close(tcp_client1);
        return;
}
```

client2_tcp.c

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>

void main()
{
        int tcp_client2=socket(AF_INET,SOCK_STREAM,0);
        if(tcp_client1<0)
        {
                printf("Socket Creation Failed!!!\n");
                return;
        }

        struct sockaddr_in caddr;
        caddr.sin_family=AF_INET;
        caddr.sin_port=htons(8080);
        caddr.sin_addr.s_addr=inet_addr("127.0.0.1");

         if(connect(tcp_client2,(struct   sockaddr*)&caddr,sizeof(caddr))<0)\
        {
                printf("Connection Failed..!!\n");
                close(tcp_client2);
                return;
        }
        int num;
        if(recv(tcp_client2,&num,sizeof(num),0)<0)
        {
                printf("Message Reception Failed..!!\n");
                close(tcp_client2);
                return;
        }
        printf("Message Received: %d\n",num);
        close(tcp_client2);
        return;
}
```

server_tcp.c

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>
```

```
#include<math.h>


void main()
{
	int tcp_server=socket(AF_INET,SOCK_STREAM,0);
	if(tcp_server<0)
	{
		printf("Socket Creation Failed!!!\n");
		return;
	}

	struct sockaddr_in server_addr;
	server_addr.sin_family=AF_INET;
	server_addr.sin_port=htons(8080);
	server_addr.sin_addr.s_addr=inet_addr("127.0.0.1");

	 if(bind(tcp_server,(struct    sockaddr*)&server_addr,sizeof(server_addr))<0)
	{
		printf("Bining Failed!!!\n");
		close(tcp_server);
		return;
	}
	printf("Binding Successfull...\n");

	 if(listen(tcp_server,5)<0)
	{
		printf("Listen Failed!!!\n");
		close(tcp_server);
		return;
	}
	printf("Listening...\n");

	struct sockaddr_in caddr;
	int isock,clen=sizeof(caddr);
	 if ((isock=accept(tcp_server,(struct sockaddr*)&caddr,&clen))<0)
	{
		printf("Failed to establish connection!!!");
		close(tcp_server);
		return;
	}
	printf("connected to client 1...");
	int num;
	 if (recv(isock,&num,sizeof(num),0)<0)
	{
		printf("Failed to Recevie!!!\n");
		close(isock);
		close(tcp_server);
		return;
	}

	printf("Received Number: %d\n",num);
	num=pow(num,2);
	 if ((isock=accept(tcp_server,(struct sockaddr*)&caddr,&clen))<0)
	{
```

```
                printf("Failed to establish connection!!!");
                close(tcp_server);
                return;
        }
        printf("connected to client 2...");
         if (send(isock,&num,sizeof(int),0)<0)
        {
                printf("Failed to Recevie!!!\n");
                close(isock);
                close(tcp_server);
                return;
        }
        printf("Message Sent to client 2!\n");
        close(isock);
        close(tcp_server);
        return;
}
```

OUTPUT:

./server

Binding Successfull...
Listening...
connected to client 1...Received Number: 10
connected to client 2...Message Sent to client 2!

./client1

Message sent..!!!

./client2

Message Received: 100

client1.c

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>


void main()
{
        int client_sock=socket(AF_INET,SOCK_DGRAM,0);
        if(client_sock<0)
        {
                printf("Socket Creation Failed..!!!\n");
                return;
        }
        struct sockaddr_in server_addr;
        server_addr.sin_family=AF_INET;
        server_addr.sin_port=htons(8000);
        server_addr.sin_addr.s_addr=inet_addr("127.0.0.1");

        float num=10.0;

        if(sendto(client_sock,&num,sizeof(float),0,(struct
sockaddr*)&server_addr,sizeof(server_addr))<0)
        {
                printf("Message Sending Failed..!!\n");
                close(client_sock);
                return;
        }

        printf("Message Sent to server...\n");
        close(client_sock);
        return;
}
```

client2.c

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>

void main()

{
        int client_sock=socket(AF_INET,SOCK_DGRAM,0);
        if(client_sock<0)
        {
                printf("Scokket creation
                Failed...!!!\n");  return;
```

```
        }
        struct sockaddr_in server_addr;
        server_addr.sin_family=AF_INET;
        server_addr.sin_port=htons(8000);
        server_addr.sin_addr.s_addr=inet_addr("127.0.0.1");

        char buffer[50]="Client 2 Ready";
        if(sendto(client_sock,buffer,sizeof(buffer),0,(struct
sockaddr*)&server_addr,sizeof(server_addr))<0)
        {
                printf("Message Sending Failed..!!\n");
                close(client_sock);
                return;
        }
        int addrlen=sizeof(server_addr);
        printf("Waiting for message from
        server...\n");  float num;
         if(recvfrom(client_sock,&num,sizeof(float),0,(struct
        sockaddr*)&server_addr,&addrlen)<0)
        {
                printf("Message Reception Failed..!!\n");
                close(client_sock);
                return;
        }
        printf("Squared: %f\n",num);
        close(client_sock);
        return;
}
```

server.c

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>
#include<math.h>

void main()
{
        int udp_sock=socket(AF_INET,SOCK_DGRAM,0);
        if(udp_sock<0)
        {
                printf("Socket Creation
                Failed...!!!\n");  return;
        }
        struct sockaddr_in server_addr;
        server_addr.sin_family=AF_INET;
        server_addr.sin_port=htons(8000);
        server_addr.sin_addr.s_addr=inet_addr("127.0.0.1
        ");

         if(bind(udp_sock,(struct    sockaddr*)&server_addr,sizeof(server_addr))<0)
        {
```

```
                printf("Binding Failed...!!!\n");
                close(udp_sock);
                return;
        }


        printf("Waiting for message... \n");

        struct sockaddr_in
        client_addr;  int
        addrlen=sizeof(client_addr);
    float
    num;
         if(recvfrom(udp_sock,&num,sizeof(int),0,(struct   sockaddr*)&client_addr,&addrlen)<0)
        {
                printf("Failed to receive message..!!\n");
                close(udp_sock);
                return;
        }
        printf("Received Number: %f\n",num);
        num=pow(num,1.5);
   char string[50]="";
        if(recvfrom(udp_sock,string,sizeof(string),0,(struct
        sockaddr*)&client_addr,&addrlen)<0)
        {
                printf("Failed to receive message..!!\n");
                close(udp_sock);
                return;
      }



      #include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>
#include<math.h>

void main()
{
        int
        udp_sock=socket(AF_INET,SOCK_DGRAM,0);
        if(udp_sock<0)
        {
                printf("Socket Creation
                Failed...!!!\n");  return;
        }

        struct sockaddr_in server_addr;
        server_addr.sin_family=AF_INET;
        server_addr.sin_port=htons(8000);
        server_addr.sin_addr.s_addr=inet_addr("127.0.0.1");

         if(bind(udp_sock,(struct    sockaddr*)&server_addr,sizeof(server_addr))<0)
```

```
        {
                printf("Binding Failed...!!!\n");
                close(udp_sock);
                return;


        }


        printf("Waiting for message... \n");

        struct sockaddr_in
        client_addr;  int
        addrlen=sizeof(client_addr);
     float
    num;
         if(recvfrom(udp_sock,&num,sizeof(int),0,(struct   sockaddr*)&client_addr,&addrlen)<0)
        {
                printf("Failed to receive message..!!\n");
                close(udp_sock);
                return;
        }
        printf("Received Number: %f\n",num);
        num=pow(num,1.5);
    char string[50]="";
        if(recvfrom(udp_sock,string,sizeof(string),0,(struct
        sockaddr*)&client_addr,&addrlen)<0)
        {
                printf("Failed to receive message..!!\n");
                close(udp_sock);
                return;
        }
        printf("%s\n",string);
        if(sendto(udp_sock,&num,sizeof(float),0,(struct
sockaddr*)&client_addr,sizeof(client_addr))<0)
        {
                printf("Send Failed!!!\n");
                close(udp_sock);
                return;
        }

        printf("Message Sent to client 2!\n");

        close(udp_sock);
        return;
}
        printf("%s\n",string);
        if(sendto(udp_sock,&num,sizeof(float),0,(struct
sockaddr*)&client_addr,sizeof(client_addr))<0)
        {
                printf("Send Failed!!!\n");
                close(udp_sock);
                return;
        }
        printf("Message Sent to client 2!\n");
```

```
        close(udp_sock);
        return;
}
```

OUTPUT:

```
./server

Waiting for message....
Received Number: 10.000000
Client 2 Ready
Message Sent to client 2!

./client1

Message Sent to server...

./client2

Waiting for message from server...
Squared: 31.622776
```

client1_tcp.c

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>

void main()
{
        int tcp_client1=socket(AF_INET,SOCK_STREAM,0);
        if(tcp_client1<0)
        {
                printf("Socket Creation Failed!!!\n");
                return;
        }

        struct sockaddr_in caddr;
        caddr.sin_family=AF_INET;
        caddr.sin_port=htons(8080);
        caddr.sin_addr.s_addr=inet_addr("127.0.0.1");

         if(connect(tcp_client1,(struct   sockaddr*)&caddr,sizeof(caddr))<0)\
        {
                printf("Connection Failed..!!\n");
                close(tcp_client1);
                return;
        }

        float num= 10.0;
```

```
	if(send(tcp_client1,&num,sizeof(num),0)<0) {

		printf("Message Sending Failed..!!\n");
		close(tcp_client1);
		return;
	}
	printf("Message sent..!!!\n");
	close(tcp_client1);
	return;
}
```

client2_tcp.c

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<arpa/inet.h>
#include<math.h>


void main()
{
	int tcp_server=socket(AF_INET,SOCK_STREAM,0);
	if(tcp_server<0)
	{
		printf("Socket Creation Failed!!!\n");
		return;
	}

	struct sockaddr_in server_addr;
	server_addr.sin_family=AF_INET;
	server_addr.sin_port=htons(8080);
	server_addr.sin_addr.s_addr=inet_addr("127.0.0.1");

	 if(bind(tcp_server,(struct    sockaddr*)&server_addr,sizeof(server_addr))<0)
	{
		printf("Bining Failed!!!\n");
		close(tcp_server);
		return;
	}
	printf("Binding Successfull...\n");

	 if(listen(tcp_server,5)<0)
	{
		printf("Listen Failed!!!\n");
```

```
		close(tcp_server);
		return;
	}
	printf("Listening...\n");

	struct sockaddr_in caddr;
	int isock,clen=sizeof(caddr);

	 if ((isock=accept(tcp_server,(struct sockaddr*)&caddr,&clen))<0)
	{
		printf("Failed to establish connection!!!");
		close(tcp_server);
		return;
	}
	printf("connected to client 1...");
	float num;
	 if (recv(isock,&num,sizeof(float),0)<0)
	{
		printf("Failed to Recevie!!!\n");
		close(isock);
		close(tcp_server);
		return;
	}

	printf("Received Number: %f\n",num);
	num=pow(num,1.5);
	 if ((isock=accept(tcp_server,(struct sockaddr*)&caddr,&clen))<0)
	{
		printf("Failed to establish connection!!!");
		close(tcp_server);
		return;
	}
	printf("connected to client 2...");
	 if (send(isock,&num,sizeof(num),0)<0)
	{
		printf("Failed to Recevie!!!\n");
		close(isock);
		close(tcp_server);
		return;
	}
	printf("Message Sent to client 2!\n");
	close(isock);
	close(tcp_server);
	return;
}
```

server_tcp.c

```
#include<stdio.h>
#include<unistd.h>
#include<sys/socket.h>
```

```
#include<sys/types.h>
#include<arpa/inet.h>
#include<math.h>


void main()
{
	int tcp_server=socket(AF_INET,SOCK_STREAM,0);
	if(tcp_server<0)
	{
		printf("Socket Creation Failed!!!\n");
		return;
	}

	struct sockaddr_in server_addr;
	server_addr.sin_family=AF_INET;
	server_addr.sin_port=htons(8080);
	server_addr.sin_addr.s_addr=inet_addr("127.0.0.1");

	 if(bind(tcp_server,(struct    sockaddr*)&server_addr,sizeof(server_addr))<0)
	{
		printf("Bining Failed!!!\n");
		close(tcp_server);
		return;
	}
	printf("Binding Successfull...\n");

	 if(listen(tcp_server,5)<0)
	{
		printf("Listen Failed!!!\n");
		close(tcp_server);
		return;
	}
	printf("Listening...\n");

	struct sockaddr_in caddr;
	int isock,clen=sizeof(caddr);

	 if ((isock=accept(tcp_server,(struct sockaddr*)&caddr,&clen))<0)
	{
		printf("Failed to establish connection!!!");
		close(tcp_server);
		return;
	}
	printf("connected to client 1...");
	float num;
	 if (recv(isock,&num,sizeof(float),0)<0)
	{
		printf("Failed to Recevie!!!\n");
		close(isock);
		close(tcp_server);
		return;
```

```
        }

        printf("Received Number: %f\n",num);
        num=pow(num,1.5);
         if ((isock=accept(tcp_server,(struct sockaddr*)&caddr,&clen))<0)
        {
                printf("Failed to establish connection!!!");
                close(tcp_server);
                return;
        }
        printf("connected to client 2...");
         if (send(isock,&num,sizeof(num),0)<0)
        {
                printf("Failed to Recevie!!!\n");
                close(isock);
                close(tcp_server);
                return;
        }
        printf("Message Sent to client 2!\n");
        close(isock);
        close(tcp_server);
        return;
}
```

OUTPUT:

./server

Binding Successful
Listening...
connected to client 1...Received Number: 10.000000
connected to client 2...Message Sent to client 2!

./client1

Message sent..!!!

./client2

Message Received: 31.622776