EX2 – ECHO CLIENT SERVER

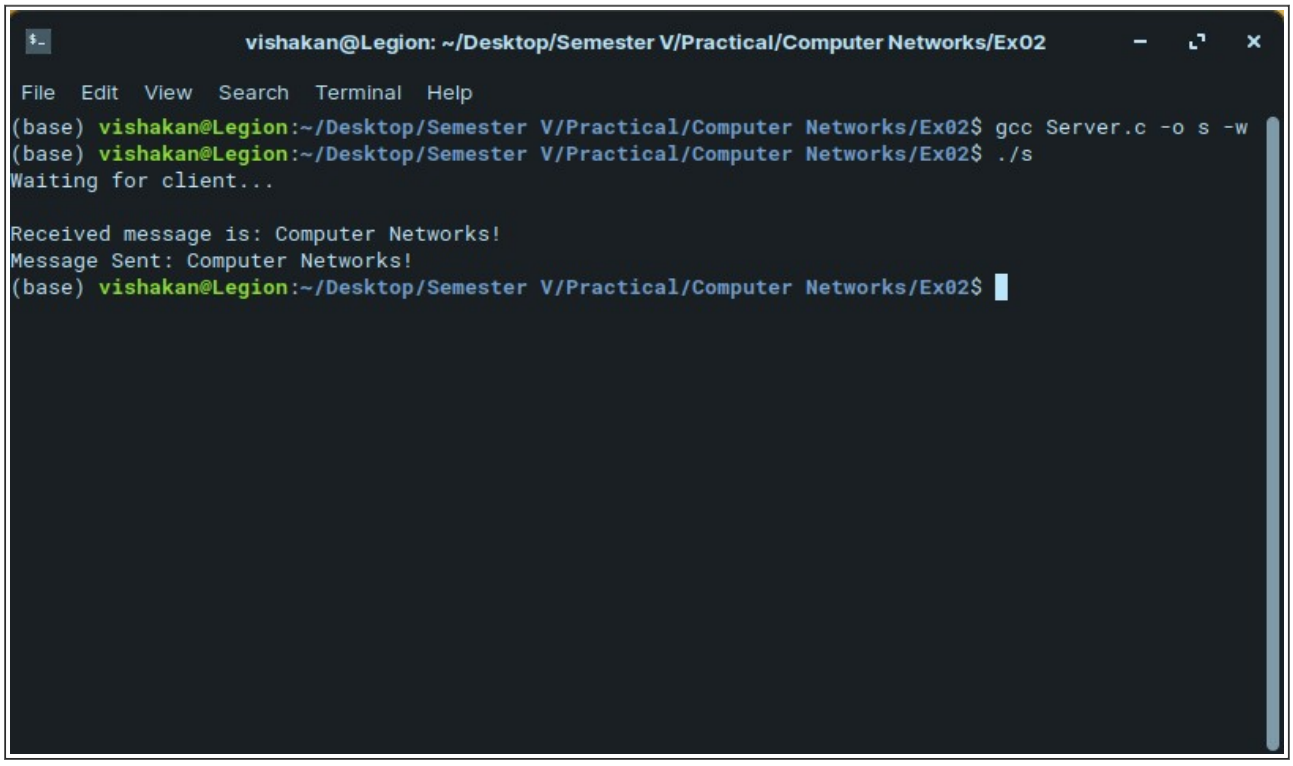
- S. Vishakan CSE – C 18 5001 196

Server Program:

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

int main(int argc, char **argv){
int sockfd, newfd, len, flag;
struct sockaddr_in server_address, client_address;
char buffer[1024];

sockfd = socket(AF_INET, SOCK_STREAM, 0); //AF_INET : IPv4 Protocol, SOCK_STREAM : reliable 2-way connection based service
//socket is a file descriptor that lets an applicaton R/W data from/to the network

if(sockfd < 0){ //Error has occurred.
perror("Socket cannot be created.\n");
exit(1);
}

bzero(&server_address, sizeof(server_address)); //Erases the data pointed to in the server_address by writing 0s
server_address.sin_family = AF_INET; //Use the Internet address family, AF_INET : IPv4 Protocol
server_address.sin_addr.s_addr = INADDR_ANY; //IP Address
server_address.sin_port = htons(7229); //Port Number; htons: host byte order -> network byte order, short

if(bind(sockfd, (struct sockaddr*)&server_address, sizeof(server_address)) < 0){
//Binding the socket to the port with server_address
perror("Bind error occurred.\n");
exit(1);
}

printf("Waiting for client...\n");
listen(sockfd, 2); //indicates that server will accept a conection. Parameter 2 indicates backlog (max # of active participants that can wait for a connection)
len = sizeof(client_address);
newfd = accept(sockfd, (struct sockaddr*)&client_address, &len);
//Accepts the first request on queue, creates another socket with the same props. of sockfd.
```

```
//If no connection request pending, blocks the server until it receives connection request
from client.
//newfd : the new socket used for data transfer

flag = read(newfd, buffer, sizeof(buffer));
//Reads on the socket

printf("\nReceived message is: %s", buffer);
printf("\nMessage Sent: %s", buffer);
//connect(sockfd, (struct sockaddr*)&client_address, sizeof(client_address));
printf("\nMessage Sent: %s", buffer);
flag = write(sockfd, buffer, sizeof(buffer));

close(sockfd);
close(newfd); //Close the sockets

return 0;
}
```

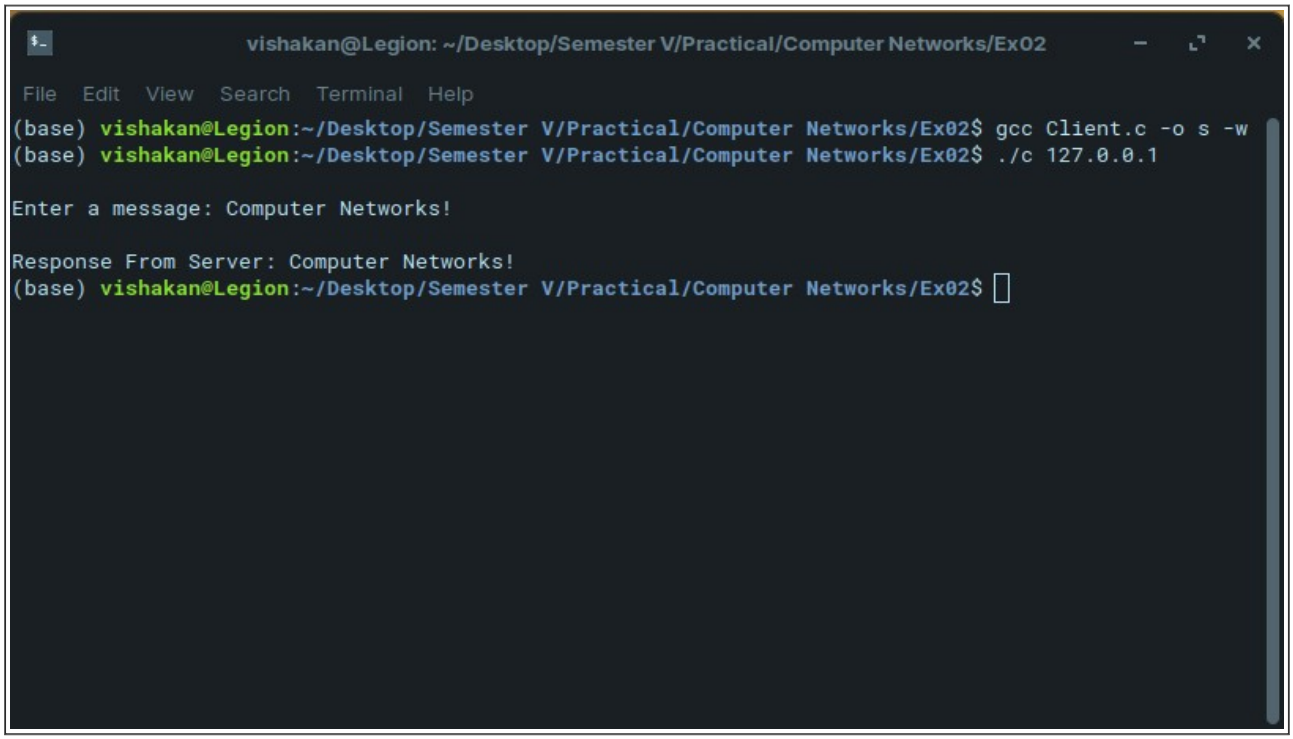
Output:



Client Program:

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

int main(int argc, char **argv){
int sockfd, flag, len;
struct sockaddr_in server_address, client_address;
char buffer[1024];

sockfd = socket(AF_INET, SOCK_STREAM, 0); //AF_INET : IPv4 Protocol, SOCK_STREAM :
reliable 2-way connection based service
//socket is a file descriptor that lets an applicaton R/W data from/to the network

if(sockfd < 0){ //Error has occurred.
perror("Socket cannot be created.\n");
exit(1);
}

bzero(&server_address, sizeof(server_address));

server_address.sin_family = AF_INET; //Use the Internet address family, AF_INET : IPv4
Protocol
server_address.sin_addr.s_addr = inet_addr(argv[1]); //IP Address, the argument to be
entered is the system's IPv4 Address (use ifconfig/ipconfig and find the 'inet' parameter)
server_address.sin_port = htons(7229); //Port Number; ntohs: network byte order -> host
byte order, short

connect(sockfd, (struct sockaddr*)&server_address, sizeof(server_address));
//Attempts to make a connection on a socket.

printf("Enter a message: ");
scanf(" %[^\n]", buffer);
flag = write(sockfd, buffer, sizeof(buffer));
//Writes on the socket

flag = read(sockfd, buffer, sizeof(buffer));
printf("Response From Server: %s\n", buffer);
//Reads information from the socket

close(sockfd); //Close the socket

return 0;
}
```

Output:


vishakan@Legion: ~/Desktop/Semester V/Practical/Computer Networks/Ex02
File Edit View Search Terminal Help
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex02$ gcc Client.c -o s -w
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex02$ ./c 127.0.0.1
Enter a message: Computer Networks!
Response From Server: Computer Networks!
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex02$