EX4 – CHAT USING TCP

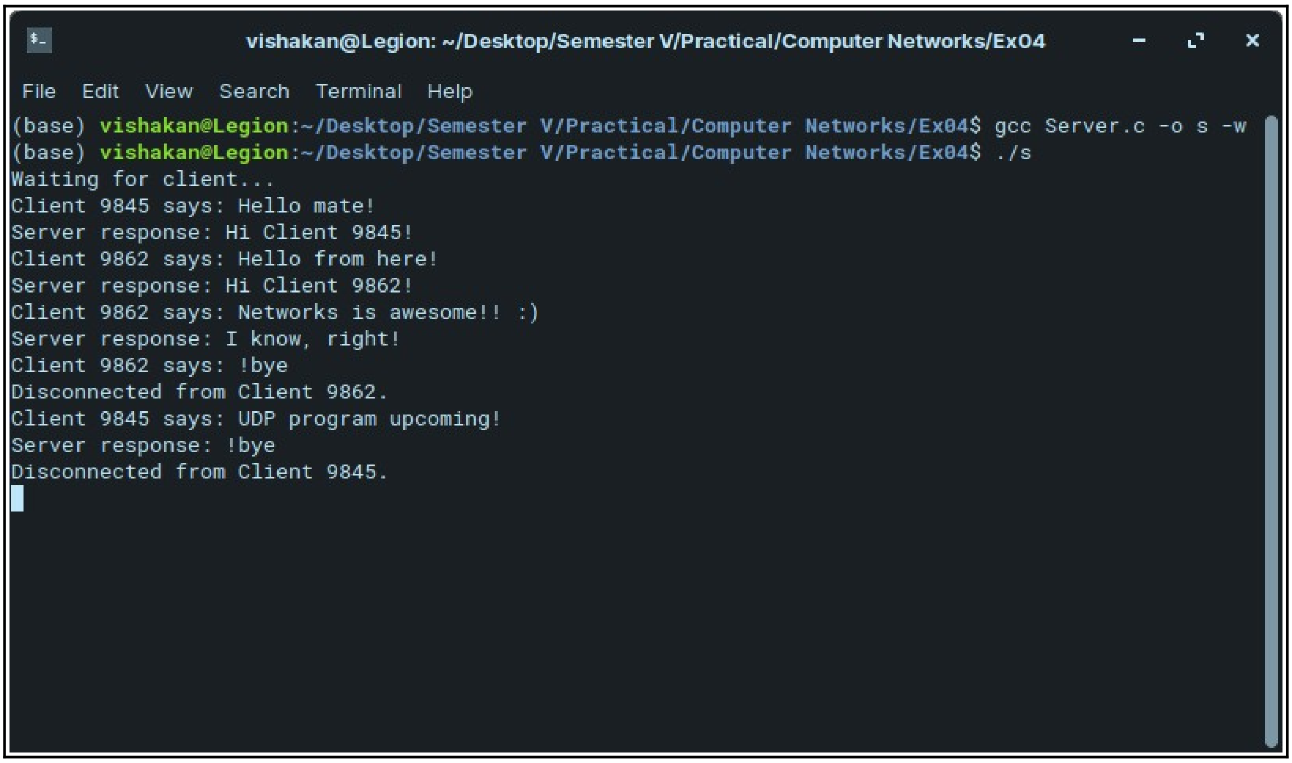
- S. Vishakan CSE – C 18 5001 196

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, curfd;
        struct sockaddr_in server_address, client_address;
        char buffer[1024];
        pid_t child;

        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, 10); //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);
```

```
    while(1){ //server is always up
        newfd = accept(sockfd, (struct sockaddr*)&client_address, &len);

        if(newfd < 0){
            printf("Error Occurred\n");
        return;
        }

        if((child = fork()) == 0){ //child process becomes new server
            int p_id = getpid();
            close(sockfd); //closing sockfd since child doesn't handle it

            while(1){ //parent process handles connected client
                bzero(buffer, sizeof(buffer));
                recv(newfd, buffer, 1024, 0);

                if(strcmp(buffer, "!bye") == 0){ //if "!bye" is received
                    printf("Client %d says: %s\n", p_id, buffer);
                    printf("Disconnected from Client %d.\n", p_id);
                    close(newfd);
                    return;
                }
                else{
                    printf("Client %d says: %s\n", p_id, buffer);
                    bzero(buffer, sizeof(buffer));
                    printf("Server response: ");
                    scanf(" %[^\n]", buffer);
                    send(newfd, buffer, strlen(buffer), 0);

                    if(strcmp(buffer, "!bye") == 0){ //if "!bye" is sent
                        printf("Disconnected from Client %d.\n", p_id);
                        close(newfd);
                        return;
                    }
                }
            }
        }

    }

close(newfd);

return 0;
}
```

Output:



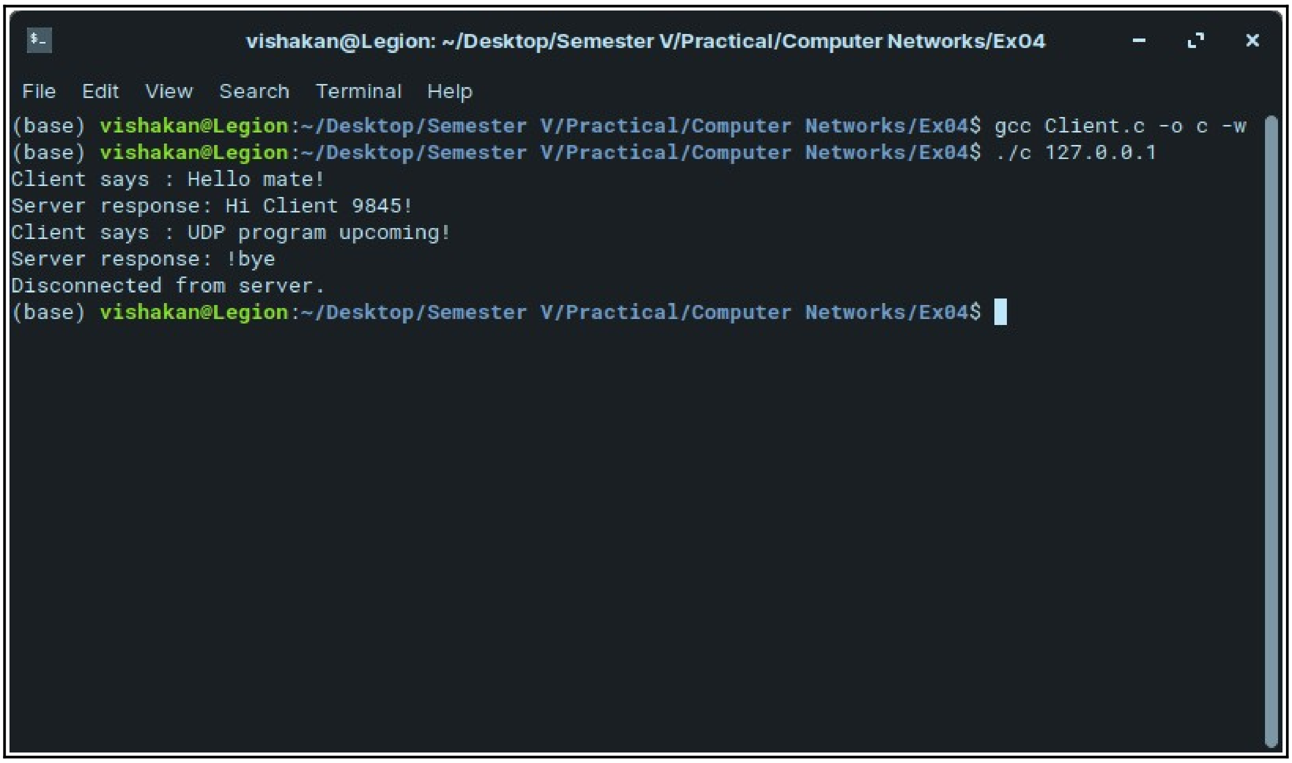
vishakan@Legion: ~/Desktop/Semester V/Practical/Computer Networks/Ex04
File Edit View Search Terminal Help

```
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex04$ gcc Server.c -o s -w
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex04$ ./s
Waiting for client...
Client 9845 says: Hello mate!
Server response: Hi Client 9845!
Client 9862 says: Hello from here!
Server response: Hi Client 9862!
Client 9862 says: Networks is awesome!! :)
Server response: I know, right!
Client 9862 says: !bye
Disconnected from Client 9862.
Client 9845 says: UDP program upcoming!
Server response: !bye
Disconnected from Client 9845.
```

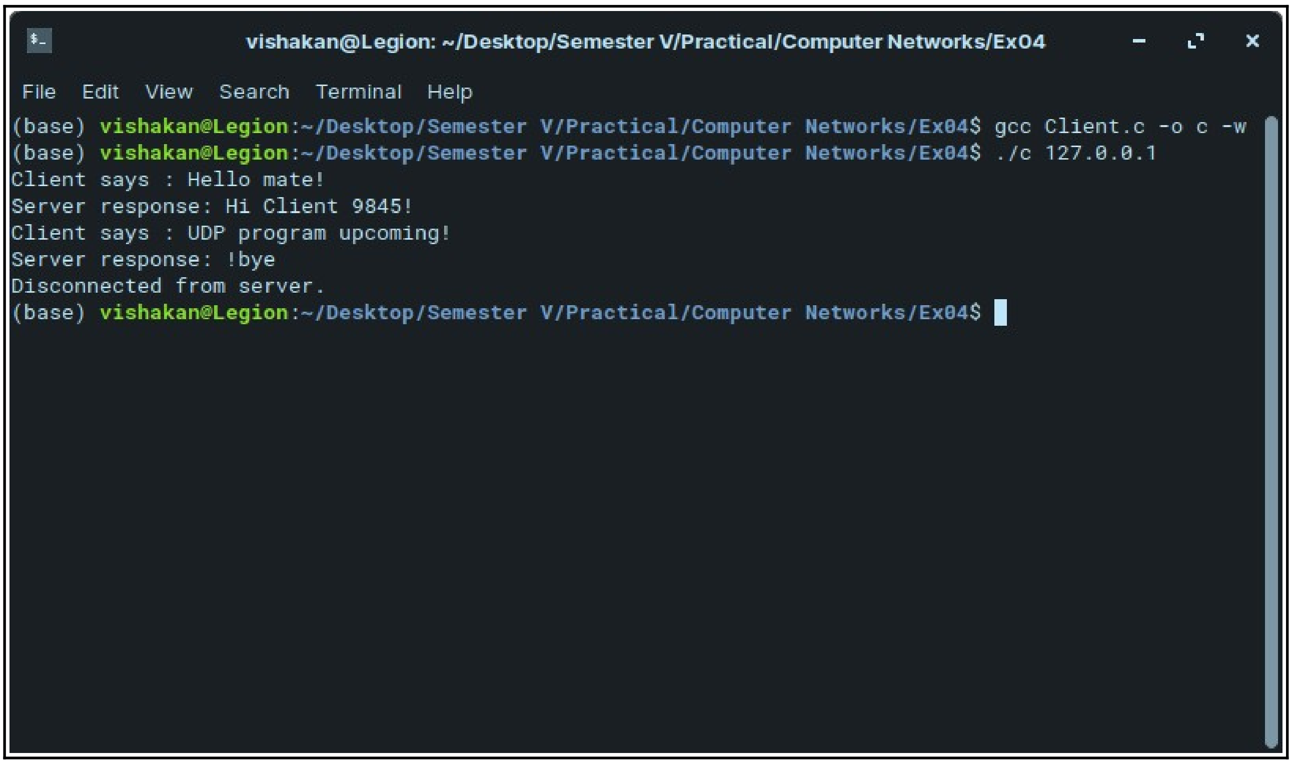
Client Program:

```
#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);

        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;
        server_address.sin_addr.s_addr = inet_addr(argv[1]);
        server_address.sin_port = htons(7229);
        connect(sockfd, (struct sockaddr*)&server_address, sizeof(server_address));

        while(1){ //client disconnects once "!bye" is sent or received
                printf("Client says : ");
                bzero(buffer, sizeof(buffer));
                scanf(" %[^\n]", buffer);
                send(sockfd, buffer, strlen(buffer), 0);

                if(strcmp(buffer, "!bye") == 0){ //if "!bye" is sent
                        close(sockfd);
                        printf("Disconnected from server.\n");
                        break;
                }
                else{
                        bzero(buffer, sizeof(buffer));
                        len = recv(sockfd, buffer, 1024, 0);
                        printf("Server response: %s\n", buffer);

                        if(strcmp(buffer, "!bye") == 0){ //if "!bye" is received
                                close(sockfd);
                                printf("Disconnected from server.\n");
                                break;
                        }
                }
        }

return 0;
}
```

Output:


vishakan@Legion: ~/Desktop/Semester V/Practical/Computer Networks/Ex04
File Edit View Search Terminal Help

```
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex04$ gcc Client.c -o c -w
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex04$ ./c 127.0.0.1
Client says : Hello mate!
Server response: Hi Client 9845!
Client says : UDP program upcoming!
Server response: !bye
Disconnected from server.
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex04$
```



vishakan@Legion: ~/Desktop/Semester V/Practical/Computer Networks/Ex04
File Edit View Search Terminal Help

```
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex04$ gcc Client.c -o c -w
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex04$ ./c 127.0.0.1
Client says : Hello mate!
Server response: Hi Client 9845!
Client says : UDP program upcoming!
Server response: !bye
Disconnected from server.
(base) vishakan@Legion:~/Desktop/Semester V/Practical/Computer Networks/Ex04$
```