Name - Kaushal Oza
Roll no - 40
SRN – 201900754
DIV – A

♦ CN Assignment -7 ♦

Multi User Chat Application Using Socket Programming-

■ Server Side:

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


void error(const char *msg)
{
        perror(msg);
        exit(1);
}



int main(int argc, char *argv[])
{
```

```
        if(argc<2)
        { fprintf(stderr, "Port number not provided!!\n");
                exit(1);
        }

        int sockfd, newsockfd, portno, n;
        char buffer[255];

        struct sockaddr_in serv_addr, cli_addr;

        socklen_t clilen; sockfd = socket(AF_INET,

        SOCK_STREAM, 0);

        if(sockfd < 0)
        {
                error("\nError opening Socket!!");
        }

        bzero((char *) &serv_addr, sizeof(serv_addr));

        portno = atoi(argv[1]);   //converts string into int

        serv_addr.sin_family = AF_INET; serv_addr.sin_addr.s_addr =
        INADDR_ANY; serv_addr.sin_port = htons(portno);  //host to
        network
short

        if(bind(sockfd, (struct sockaddr *) &serv_addr, sizeof(serv_addr)) < 0)
        { error("\nBinding Failed!");
        } listen(sockfd,

        5); clilen =

        sizeof(cli_addr);
```

```
newsockfd =

accept(sockfd,

(struct sockaddr

*) &cli_addr,

&clilen);

if(newsockfd < 0)
{ error("\nError on accept!");
}


while(1)
{
        bzero(buffer, 255); n =
        read(newsockfd, buffer, 255);
        if(n < 0)
        { error("\nError on reading!");
        }

        printf("\nClient: %s\n", buffer); bzero(buffer, 255);
        fgets(buffer, 255, stdin);        //reads bytes from string

        n= write(newsockfd, buffer, strlen(buffer));

        if(n < 0)
        { error("\nError on writing");
        } int i = strncmp("Bye", buffer,

        3);
```

```
            if(i == 0)
            {
                  break;
            }
      }

      close(newsockfd)
; close(sockfd); return
0; }
```

■ Client Side:

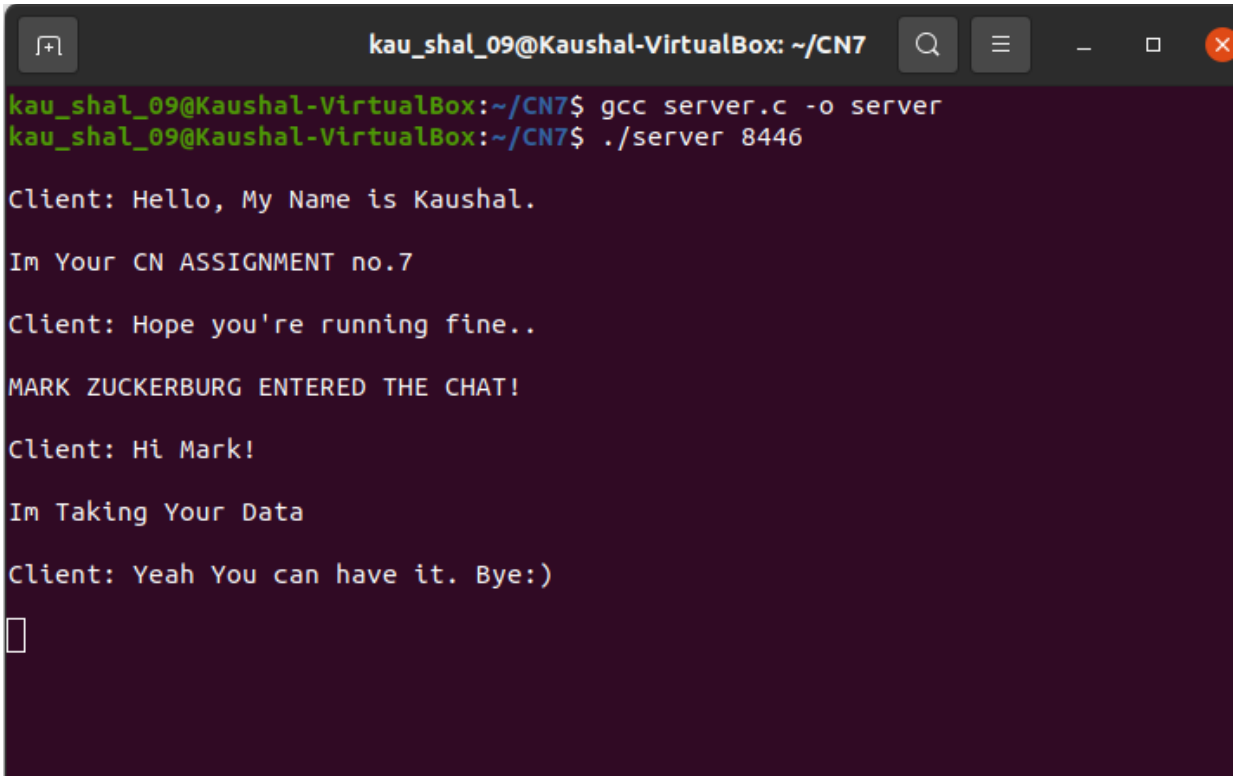
```
#include <stdio.h>
#include <stdlib.h>
#include <arpa/inet.h>
#include <netinet/in.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <unistd.h>
#include <string.h>
#include <netdb.h>


void error(const char *msg)
{
      perror(msg);
      exit(1);
}
```

```
int main(int argc, char *argv[])
{
	int sockfd, newsockfd, portno, n;
	struct sockaddr_in serv_addr;
	struct hostent *server;

	char buffer[255];
	if(argc < 3)
	{
		fprintf(stderr, "usage %s hostname port\n", argv[0]);
		exit(1);
	}

	portno = atoi(argv[2]); sockfd =
	socket(AF_INET, SOCK_STREAM, 0);
	if(sockfd < 0)
	{
		error("\nError opening Socket!!");
	}

	server = gethostbyname(argv[1]);
	if(server == NULL)
	{ fprintf(stderr, "\nError, no such host!");
	}

	bzero((char *) &serv_addr, sizeof(serv_addr)); serv_addr.sin_family
	= AF_INET; bcopy((char * ) server->h_addr, (char *)
	&serv_addr.sin_addr.s_addr,
server->h_length); serv_addr.sin_port = htons(portno); if(connect(sockfd,
(struct sockaddr *) &serv_addr, sizeof(serv_addr)) < 0)
	{
		error("\nConnection Failed!");
	}
	while(1)
	{
```

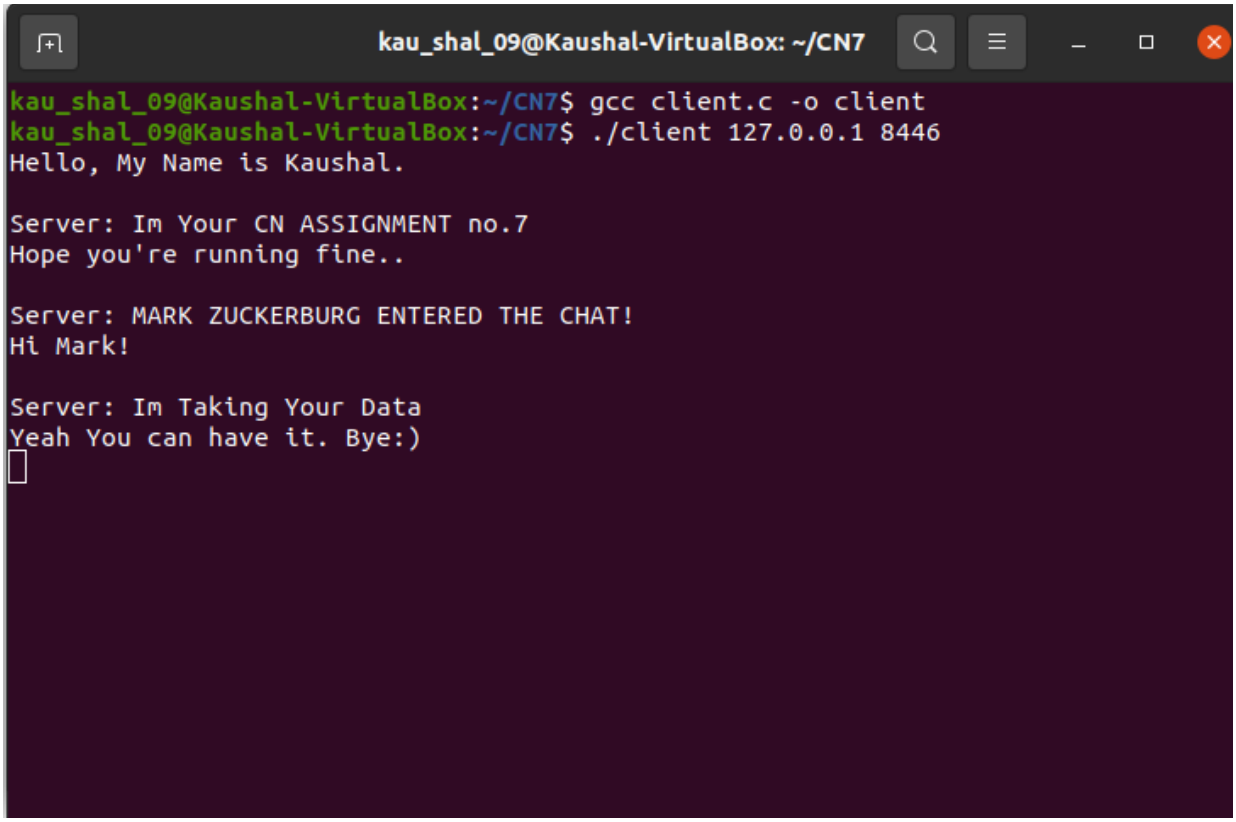
```
            bzero(buffer, 255); fgets(buffer, 255,
            stdin); n = write(sockfd, buffer,
            strlen(buffer)); if(n < 0)
            { error("\nError on writing");
            }

            bzero(buffer, 255); n =
            read(sockfd, buffer, 255);

            if(n < 0) error("\nerror on

            reading!"); printf("\nServer: %s",

            buffer);


            int i = strncmp("Bye", buffer, 3);

            if(i == 0)
            {
                  break;
            }


      }

      close(sockfd);
      ™return 0;


}
```

OUTPUT:

Server


kau_shal_09@Kaushal-VirtualBox: ~/CN7
kau_shal_09@Kaushal-VirtualBox:~/CN7$ gcc server.c -o server
kau_shal_09@Kaushal-VirtualBox:~/CN7$ ./server 8446
Client: Hello, My Name is Kaushal.
Im Your CN ASSIGNMENT no.7
Client: Hope you're running fine..
MARK ZUCKERBURG ENTERED THE CHAT!
Client: Hi Mark!
Im Taking Your Data
Client: Yeah You can have it. Bye:)

Client


kau_shal_09@Kaushal-VirtualBox: ~/CN7
kau_shal_09@Kaushal-VirtualBox:~/CN7$ gcc client.c -o client
kau_shal_09@Kaushal-VirtualBox:~/CN7$ ./client 127.0.0.1 8446
Hello, My Name is Kaushal.
Server: Im Your CN ASSIGNMENT no.7
Hope you're running fine..
Server: MARK ZUCKERBURG ENTERED THE CHAT!
Hi Mark!
Server: Im Taking Your Data
Yeah You can have it. Bye:)