**EXPERIMENT - 8**

| **Name** | Anumeya Sehgal |
| --- | --- |
| **Registration Number** | 23BAI1203 |
| **Course Code** | BCSE308P |
| **Course Title** | Computer Networks Lab |
| **Date** | 22 August, 2024 |

**AIM:** Chat Application using TCP Socket Programming

**SERVER PROGRAM:**

#include <netdb.h>

#include <netinet/in.h>

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <sys/socket.h>

#include <unistd.h>

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

int sockfd, newsockfd, portno;

socklen\_t clilen;

char buffer[256];

struct sockaddr\_in serv\_addr, cli\_addr;

ssize\_t n;

char response[256];

sockfd = socket(AF\_INET, SOCK\_STREAM, 0);

if (sockfd < 0) {

perror("ERROR opening socket");

exit(1);

}

memset(&serv\_addr, 0, sizeof(serv\_addr));

portno = 8000;

serv\_addr.sin\_family = AF\_INET;

serv\_addr.sin\_addr.s\_addr = INADDR\_ANY;

serv\_addr.sin\_port = htons(portno);

if (bind(sockfd, (struct sockaddr \*)&serv\_addr, sizeof(serv\_addr)) < 0) {

perror("ERROR on binding");

exit(1);

}

listen(sockfd, 5);

clilen = sizeof(cli\_addr);

printf("Server is listening on port %d...\n", portno);

newsockfd = accept(sockfd, (struct sockaddr \*)&cli\_addr, &clilen);

if (newsockfd < 0) {

perror("ERROR on accept");

exit(1);

}

printf("Client connected\n");

while (1) {

memset(buffer, 0, sizeof(buffer));

n = recv(newsockfd, buffer, sizeof(buffer) - 1, 0);

if (n < 0) {

perror("ERROR reading from socket");

break;

} else if (n == 0) {

printf("Client disconnected\n");

break;

}

printf("Message received: %s", buffer);

printf("Enter response: ");

if (fgets(response, sizeof(response), stdin) == NULL) {

break;

}

n = send(newsockfd, response, strlen(response), 0);

if (n < 0) {

perror("ERROR writing to socket");

break;

}

}

close(newsockfd);

close(sockfd);

return 0;

}

**CLIENT PROGRAM:**

#include <netdb.h>

#include <netinet/in.h>

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <sys/socket.h>

#include <unistd.h>

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

int sockfd, portno, n;

struct sockaddr\_in serv\_addr;

struct hostent \*server;

char buffer[256];

if (argc < 3) {

fprintf(stderr, "usage %s hostname port \n", argv[0]);

exit(0);

}

portno = atoi(argv[2]);

sockfd = socket(AF\_INET, SOCK\_STREAM, 0);

if (sockfd < 0) {

perror("ERROR opening socket");

exit(1);

}

server = gethostbyname(argv[1]);

if (server == NULL) {

fprintf(stderr, "ERROR, no such host \n");

exit(0);

}

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) {

perror("ERROR connecting");

exit(1);

}

while (1) {

printf("Please enter the message: ");

bzero(buffer, 256);

fgets(buffer, 255, stdin);

n = send(sockfd, buffer, strlen(buffer), 0);

if (n < 0) {

perror("ERROR writing to socket");

exit(1);

}

bzero(buffer, 256);

n = recv(sockfd, buffer, 255, 0);

if (n < 0) {

perror("ERROR reading from socket");

exit(1);

}

printf("%s\n", buffer);

}

close(sockfd);

return 0;

}

**OUTPUT:**

