

DCCN LAB 4

NAME: **MALOTH ADITYA**

ROLL NO.: **120CS0124**

Q.1 Read the instructions and run the simple client server program given in help document (TCP_Socket1.doc).

Client Code:

```
/*TCP_Client*/
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
int main()
{
    int sid;
    char c;
    struct sockaddr_in server_address;
    int server_addlen;

    server_address.sin_family=AF_INET;

    server_address.sin_addr.s_addr=inet_addr("127.0.0.1");

    server_address.sin_port=5080;

    server_addlen=sizeof(server_address);

    sid=socket(AF_INET,SOCK_STREAM,0);

    connect(sid,(struct sockaddr *)&server_address,server_addlen);

    write(sid,"A",1);
    read(sid,&c,1);

    printf("Char from server is %c\n",c);
    close(sid);
    return(0);
}
```

Server Code:

```
/*TCP_Server*/
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <stdio.h>
```

DCCN LAB 4

```
#include <unistd.h>
#include <stdlib.h>
int main()
{
    int serid, sessid;
    char c;
    struct sockaddr_in server_address, client_address;

    unsigned int server_addlen, client_addlen;

    server_address.sin_family = AF_INET;
    server_address.sin_addr.s_addr = inet_addr("127.0.0.1");
    server_address.sin_port = 5080;

    server_addlen = sizeof(server_address);
    client_addlen = sizeof(client_address);

    serid = socket(AF_INET, SOCK_STREAM, 0);

    bind(serid, (struct sockaddr*)&server_address, server_addlen);

    listen(serid, 10);

    while(1)
    {
        printf("Server is ready to accept ..... \n");
        sessid = accept(serid, (struct sockaddr*)&client_address, &client_addlen);
        read(sessid, &c, 1);
        write(sessid, &c, 1);
        close(sessid);
    }
    return(0);
}
```

Output:

Client Output:

```
nit@nit-HP-EliteDesk-800-G1-SFF:~/120CS0124/Lab 4 02 Feb$ gcc -o client1 client1.c
nit@nit-HP-EliteDesk-800-G1-SFF:~/120CS0124/Lab 4 02 Feb$ ./client1
Char from server is A
```

Server Output:

```
nit@nit-HP-EliteDesk-800-G1-SFF:~/120CS0124/Lab 4 02 Feb$ gcc -o server1 server1.c
nit@nit-HP-EliteDesk-800-G1-SFF:~/120CS0124/Lab 4 02 Feb$ ./server1
Server is ready to accept .....
Server is ready to accept .....
█
```

DCCN LAB 4

Q.2 Execute a client/server program for adding a two integer numbers requested by the client and evaluated at server and get back result at the client. You will be appreciated if you use command line arguments. (You can input ip address of the machine at the time of execution)(Hint: Use TCP_Socket2.doc file)

Client Code:

```
/*TCP_Client*/
#include <stdio.h>
#include <string.h>
#include <sys/socket.h>
#include <arpa/inet.h>

int main(int argc, char *argv[]) {
    int client_socket;
    struct sockaddr_in server;
    int num1, num2, result;

    client_socket = socket(AF_INET, SOCK_STREAM, 0);
    if (client_socket == -1) {
        printf("Could not create socket");
        return 1;
    }

    server.sin_addr.s_addr = inet_addr("127.0.0.1");
    server.sin_family = AF_INET;
    server.sin_port = htons(5000);

    if (connect(client_socket, (struct sockaddr *)&server, sizeof(server)) < 0) {
        printf("Connect failed");
        return 1;
    }

    printf("Enter first number: ");
    scanf("%d", &num1);
    printf("Enter second number: ");
    scanf("%d", &num2);

    num1 = htonl(num1);
    num2 = htonl(num2);
    send(client_socket, &num1, sizeof(num1), 0);
    send(client_socket, &num2, sizeof(num2), 0);
    recv(client_socket, &result, sizeof(result), 0);
    // result = ntohl(result);

    printf("Result: %d\n", result);

    close(client_socket);
}
```

DCCN LAB 4

```
    return 0;
}
```

Server Code:

```
/*TCP_Server*/
#include <stdio.h>
#include <string.h>
#include <sys/socket.h>
#include <arpa/inet.h>

int add_numbers(int num1, int num2) {
    return num1 + num2;
}

int main(int argc, char *argv[]) {
    int server_socket, client_socket;
    struct sockaddr_in server, client;
    int c;
    int num1, num2, result;

    server_socket = socket(AF_INET, SOCK_STREAM, 0);
    if (server_socket == -1) {
        printf("Could not create socket");
        return 1;
    }

    server.sin_family = AF_INET;
    server.sin_addr.s_addr = INADDR_ANY;
    server.sin_port = htons(5000);

    if (bind(server_socket, (struct sockaddr *)&server, sizeof(server)) < 0) {
        printf("Bind failed");
        return 1;
    }

    listen(server_socket, 3);

    printf("Waiting for incoming connections...\n");
    c = sizeof(struct sockaddr_in);

    while ((client_socket = accept(server_socket, (struct sockaddr *)&client,
    (socklen_t *)&c))) {
        printf("Connection accepted from %s:%d\n", inet_ntoa(client.sin_addr),
        ntohs(client.sin_port));
        recv(client_socket, &num1, sizeof(num1), 0);
        recv(client_socket, &num2, sizeof(num2), 0);
        result = add_numbers(ntohl(num1), ntohl(num2));
        send(client_socket, &result, sizeof(result), 0);
    }
}
```

DCCN LAB 4

```
    close(client_socket);  
    printf("Connection closed\n");  
}  
  
if (client_socket < 0) {  
    printf("Accept failed");  
    return 1;  
}  
  
return 0;  
}
```

Output:

Client Output:

```
nit@nit-HP-EliteDesk-800-G1-SFF:~/120CS0124/Lab 4 02 Feb$ ./client  
Enter first number: 588  
Enter second number: 125  
Result: 713
```

Server Output:

```
nit@nit-HP-EliteDesk-800-G1-SFF:~/120CS0124/Lab 4 02 Feb$ ./server  
Waiting for incoming connections...  
Connection accepted from 127.0.0.1:48238  
Connection closed
```