

Assignment- 4

Sibasis Malla
120CS0179

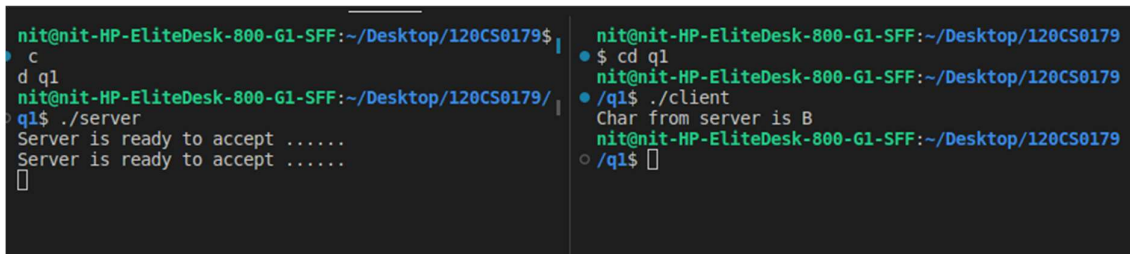
```
/*TCP_Client*/
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
int main()
{
    int sid;
    char c;
    struct sockaddr_in server_address;
    int server_addlen;
    server_address.sin_family = AF_INET;
    server_address.sin_addr.s_addr = 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, "B", 1);
    read(sid, &c, 1);
    printf("Char from server is %c\n", c);
    close(sid);
    return (0);
}
```

```
/*TCP_Server*/
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
int main()
{
    int serid, sessid;
    char c;
    struct sockaddr_in server_address, client_address;
    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);
}

```



```

nit@nit-HP-EliteDesk-800-G1-SFF:~/Desktop/120CS0179$
c
d q1
nit@nit-HP-EliteDesk-800-G1-SFF:~/Desktop/120CS0179/
q1$ ./server
Server is ready to accept .....
Server is ready to accept .....
q1$

nit@nit-HP-EliteDesk-800-G1-SFF:~/Desktop/120CS0179
$ cd q1
nit@nit-HP-EliteDesk-800-G1-SFF:~/Desktop/120CS0179
/q1$ ./client
Char from server is B
nit@nit-HP-EliteDesk-800-G1-SFF:~/Desktop/120CS0179
/q1$

```

Q2

client

```

#include <sys/types.h>

#include <sys/socket.h>

#include <netinet/in.h>

#include <arpa/inet.h>

#include <netdb.h>

#include <stdio.h>

#include <unistd.h>

#include <string.h>

#include <strings.h>

#include <stdlib.h>

#define MAX_MSG 100

#define SERVER_ADDR "127.0.0.1"

#define CLIENT_ADDR "127.0.0.1"

#define SERVER_PORT 3786

```

```

#define CLIENT_PORT 8229

main()
{
int sd, rc, i, n;

struct sockaddr_in
clientAddr,
servAddr;

char
line[MAX_MSG];

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

servAddr.sin_family = AF_INET;

servAddr.sin_addr.s_addr = inet_addr(SERVER_ADDR);

servAddr.sin_port = htons(SERVER_PORT);

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

clientAddr.sin_family = AF_INET;

clientAddr.sin_addr.s_addr = INADDR_ANY;

clientAddr.sin_port = htons(0);

sd = socket(AF_INET, SOCK_STREAM, 0);

printf("successfully created stream socket \n");

bind(sd, (struct sockaddr *)&clientAddr, sizeof(clientAddr));

printf("bound local port successfully\n");

connect(sd, (struct sockaddr *)&servAddr, sizeof(servAddr));

printf("connected to server successfully\n");

do
{

```

```

printf("Enter 1st number : ");

scanf("%s", line);

send(sd, line, strlen(line) + 1, 0);

printf("data sent (%s)\n", line);

printf("Enter 2nd number : ");

scanf("%s", line);

send(sd, line, strlen(line) + 1, 0);

printf("data sent (%s)\n", line);

n = recv(sd, line, MAX_MSG, 0);

printf("received from server %s\n", line);

} while (strcmp(line, "quit"));

printf("closing connection with the server\n");

close(sd);
}

```

Server

```

#include <netinet/in.h>

#include <arpa/inet.h>

#include <netdb.h>

#include <stdio.h>

#include <unistd.h>

#include <strings.h>

#include <string.h>

#include <math.h>

#include <stdlib.h>

#define MAX_MSG 100

#define SERVER_ADDR "127.0.0.1"

```

```

#define SERVER_PORT 3786

main()

{

int sd, newSd, cliLen, n;
float num1,num2;
float sum;

struct sockaddr_in cliAddr, servAddr;

char line[MAX_MSG], line1[MAX_MSG], line2[MAX_MSG];

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

servAddr.sin_family = AF_INET;

servAddr.sin_addr.s_addr = inet_addr(SERVER_ADDR);

servAddr.sin_port =

htons(SERVER_PORT);

sd = socket(AF_INET, SOCK_STREAM, 0);

printf("successfully created stream socket \n");

pid_t childpid;

bind(sd, (struct sockaddr *)&servAddr,

sizeof(servAddr));

printf("bound local port successfully\n");

socklen_t addr_size;

listen(sd, 5);

while (1)

{

printf("waiting for client connection on port TCP %u\n", SERVER_PORT);

cliLen = sizeof(cliAddr);

newSd = accept(sd, (struct sockaddr *)&cliAddr, &addr_size);

```

```

printf("received connection from host [IP %s ,TCP port %d]\n",
inet_ntoa(cliAddr.sin_addr), ntohs(cliAddr.sin_port));

if ((childpid = fork()) == 0)
{
close(sd);

do
{
memset(line, 0x0, MAX_MSG);

n = recv(newSd, line, MAX_MSG, 0);

num1 = atof(line);

n = recv(newSd, line, MAX_MSG, 0);

num2 = atof(line);

sum = num1 + num2;

sprintf(line1, "%f", sum);

printf("received from host [IP %s ,TCP port %d] : %s\n",
inet_ntoa(cliAddr.sin_addr), ntohs(cliAddr.sin_port), line1);

send(newSd, line1, strlen(line1) + 1, 0);

} while (abs(strcmp(line, "quit")));

printf("closing connection with host [IP %s ,TCP port %d]\n",
inet_ntoa(cliAddr.sin_addr), ntohs(cliAddr.sin_port));
}

close(newSd);
}
}

```

```

connected to server successfully
Enter 1st number : 10.2
nit@nit-HP-EliteDesk-800-G1-SFF:~/Desktop/120CS017
9/q2 $ ./client
successfully created stream socket
bound local port successfully
connected to server successfully
Enter 1st number : 10.2
data sent (10.2)
Enter 2nd number : 11.3
data sent (11.3)
received from server 21.500000
Enter 1st number :

nit@nit-HP-EliteDesk-800-G1-SFF:~/Desktop/120CS017/q
2 $ ./server
successfully created stream socket
bound local port successfully
waiting for client connection on port TCP 3786
received connection from host [IP 0.0.0.0 ,TCP port 0
]
waiting for client connection on port TCP 3786
received from host [IP 0.0.0.0 ,TCP port 0] : 21.5000
00

```