

# 18CSC302J (Computer Networks Lab)

## Lab session - Implementation of File Transfer Protocol

Name :- Puneet Sharma

Reg. No. :- RA1911003010331

Class :-CSE F1

### SERVER CODE:

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

#define PORT 8231
#define SIZE 1024

void send_file(int sockfd, FILE *fp)
{
    char file_data[SIZE];
    while(fgets(file_data, SIZE, fp) != NULL)
    {
        if(send(sockfd, file_data, sizeof(file_data), 0) == -1)
        {
            perror("Error in sending file...");
            exit(1);
        }
    }
}
```

```
    }  
    bzero(file_data, SIZE);  
}  
return;  
}
```

```
int main()  
{  
    int sockfd, conn;  
    struct sockaddr_in server_addr, client_addr;  
    socklen_t client_len;  
  
    FILE *fhand;  
    char *filename = "data.txt";  
  
    sockfd = socket(AF_INET, SOCK_STREAM, 0);  
    if(sockfd < 0)  
    {  
        perror("Error in socket...");  
        exit(1);  
    }  
  
    printf("Server socket created successfully...\n");  
  
    bzero(&server_addr, sizeof(server_addr));  
    bzero(&client_addr, sizeof(client_addr));  
    server_addr.sin_family = AF_INET;  
    server_addr.sin_port = htons(PORT);  
    server_addr.sin_addr.s_addr = htonl(INADDR_ANY);  
  
    bind(sockfd, (struct sockaddr *)&server_addr, sizeof(server_addr));
```

```
listen(sockfd, 5);

printf("Server is listening...\n");

conn = accept(sockfd, (struct sockaddr *)&client_addr, &client_len);

printf("Server is connected...\n");

fhand = fopen(filename, "r");
if (fhand == NULL)
{
    perror("Error in reading file...");
    exit(1);
}
send_file(conn, fhand);

printf("File data sent successfully...\n");

close(sockfd);

return 0;
}
```

```
Ftp_server.c  x  data.txt  x  Remote_command_ x  recv.txt  x  +
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4  #include <arpa/inet.h>
5  #include <sys/types.h>
6  #include <sys/socket.h>
7  #include <netdb.h>
8  #include <netinet/in.h>
9  #include <unistd.h>
10 #include <sys/stat.h>
11
12 #define PORT 8231
13 #define SIZE 1024
14
15 void send_file(int sockfd, FILE *fp)
16 {
17     char file_data[SIZE];
18     while(fgets(file_data, SIZE, fp) != NULL)
19     {
20         if(send(sockfd, file_data, sizeof(file_data), 0) == -1)
21         {
22             perror("Error in sending file...");
23             exit(1);
24         }
25         bzero(file_data, SIZE);
26     }
27     return;
28 }
29
30 int main()
31 {
32     int sockfd, conn;
33     struct sockaddr_in server_addr, client_addr;
34     socklen_t client_len;
35
36     FILE *fhand;
37     char *filename = "data.txt";
38
39     sockfd = socket(AF_INET, SOCK_STREAM, 0);
40     if(sockfd < 0)
```

```
331/Ftp_server.c - S... 331/Remote_comm...
Run  Run Config | Com 331/Ftp_server.c  Runner: C  CWD  ENV
Server socket created successfully...
Server is listening...
Server is connected...
File data sent successfully...
Process exited with code: 0
```

## CLIENT CODE

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <arpa/inet.h>
```

```
#include <sys/types.h>
#include <sys/socket.h>
#include <netdb.h>
#include <netinet/in.h>
#include <unistd.h>
#include <sys/stat.h>

#define PORT 8231
#define SIZE 1024

void recv_file(int sockfd)
{
    int bytes_recv;
    FILE *fp;
    char *filename = "recv.txt";
    char recv_buffer[SIZE];

    fp = fopen(filename, "w");
    while (1)
    {
        bytes_recv = recv(sockfd, recv_buffer, SIZE, 0);
        if (bytes_recv <= 0)
        {
            break;
        }
        fprintf(fp, "%s", recv_buffer);
        bzero(recv_buffer, SIZE);
    }
    return;
}
```

```
int main()
{
    int sockfd;
    struct sockaddr_in server_addr;

    sockfd = socket(AF_INET, SOCK_STREAM, 0);
    printf("Client socket created successfully...\n");

    bzero(&server_addr, sizeof(server_addr));
    server_addr.sin_family = AF_INET;
    server_addr.sin_addr.s_addr = htonl(INADDR_ANY);
    server_addr.sin_port = htons(PORT);

    connect(sockfd, (struct sockaddr *)&server_addr, sizeof(server_addr));

    printf("Client is connected...\n");

    recv_file(sockfd);

    printf("File data received successfully...\n");

    close(sockfd);

    return 0;
}
```

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

#define PORT 8231
#define SIZE 1024

void recv_file(int sockfd)
{
    int bytes_rcv;
    FILE *fp;
    char *filename = "recv.txt";
    char recv_buffer[SIZE];

    fp = fopen(filename, "w");
    while (1)
    {
        bytes_rcv = recv(sockfd, recv_buffer, SIZE, 0);
        if (bytes_rcv <= 0)
        {
            break;
        }
        fprintf(fp, "%s", recv_buffer);
        bzero(recv_buffer, SIZE);
    }
    return;
}

int main()
{
    int sockfd;
    struct sockaddr_in server_addr;
```

```
331/Ftp_clint.c - Stoj x 331/Remote_commz
Run Run Config I Com 331/Ftp_clint.c Runner: C CWD ENV
Client socket created successfully...Client is connected...
File data received successfully...
Process exited with code: 0
```