EX:4 **FILE TRANSFER PROTOCOL**

**SERVER:**

#include<stdio.h>

#include<sys/types.h>

#include<netinet/in.h>

#include<stdio.h>

#include<sys/types.h>

#include<netinet/in.h>

#include<string.h>

#include<stdlib.h>

int main()

{

FILE \*fp;

int sd,newsd,ser,n,a,cli,pid,bd,port,clilen;

char name[100],fileread[100],fname[100],ch,file[100],rcv[100];

struct sockaddr\_in servaddr,cliaddr;

printf("Enter the port address\n");

scanf("%d",&port);

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

if(sd<0)

printf("Cant create\n");

else

printf("Socket is created\n");

servaddr.sin\_family=AF\_INET;

servaddr.sin\_addr.s\_addr=htonl(INADDR\_ANY);

servaddr.sin\_port=htons(port);

a=sizeof(servaddr);

bd=bind(sd,(struct sockaddr \*)&servaddr,a);

if(bd<0)

printf("Cant bind\n");

else

printf("Binded\n");

listen(sd,5);

clilen=sizeof(cliaddr);

newsd=accept(sd,(struct sockaddr \*)&cliaddr,&clilen);

if(newsd<0)

{

printf("Cant accept\n");

}

else

printf("Accepted\n");

n=recv(newsd,rcv,100,0);

rcv[n]='\0';

fp=fopen(rcv,"r");

if(fp==NULL)

{

send(newsd,"error",5,0);

close(newsd);

}

else

{

while(fgets(fileread,sizeof(fileread),fp))

{

if(send(newsd,fileread,sizeof(fileread),0)<0)

{

printf("Cant send\n");

}

}

if(!fgets(fileread,sizeof(fileread),fp))

{

send(newsd,"completed",999999999,0);

}

return(0);

}

}

**OUTPUT:**

[211716205024@Putty ~]$ vi FTPserver.c

[211716205024@Putty ~]$ gcc FTPserver.c

[211716205024@Putty ~]$ ./a.out

Enter the port address

1234

Socket is created

Binded

Accepted

**CLIENT:**

#include<stdio.h>

#include<stdlib.h>

#include<sys/socket.h>

#include<netinet/in.h>

int main()

{

FILE \*fp;

int csd,n,ser,s,cli,cport,newsd;

char name[100],rcvmsg[100],rcvg[100],fname[100];

struct sockaddr\_in servaddr;

printf("Enter the port");

scanf("%d",&cport);

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

if(csd<0)

{

printf("Error....\n");

}

else

printf("Socket is created\n");

servaddr.sin\_family=AF\_INET;

servaddr.sin\_addr.s\_addr=htonl(INADDR\_ANY);

servaddr.sin\_port=htons(cport);

if(connect(csd,(struct sockaddr \*)&servaddr,sizeof(servaddr))<0)

printf("Error in connection\n");

else

printf("connected\n");

printf("Enter the existing file name\t");

scanf("%s",name);

printf("Enter the new file name\t");

scanf("%s",fname);

fp=fopen(fname,"w");

send(csd,name,sizeof(name),0);

while(1)

{

s=recv(csd,rcvg,100,0);

rcvg[s]='\0';

if(strcmp(rcvg,"error")==0)

printf("File is not available\n");

if(strcmp(rcvg,"completed")==0)

{

printf("File is transferred........\n");

fclose(fp);

close(csd);

break;

}

else

fputs(rcvg,stdout);

fprintf(fp,"%s",rcvg);

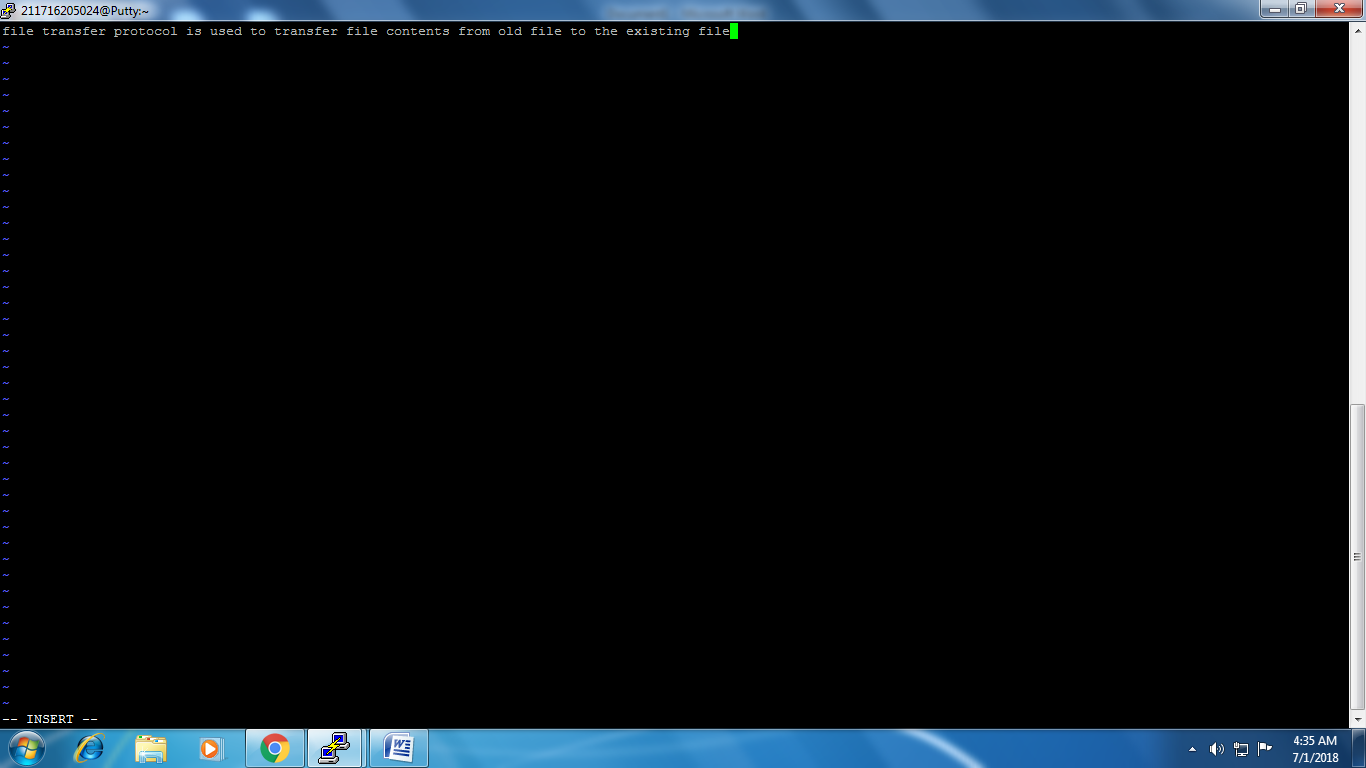
}

}

**OUTPUT:**

[211716205024@Putty ~]$ vi FTPclient.c

[211716205024@Putty ~]$ vi kailash.txt



[211716205024@Putty ~]$ gcc FTPclient.c

[211716205024@Putty ~]$ ./a.out

Enter the port1234

Socket is created

connected

Enter the existing file name kailash.txt

Enter the new file name eswaran.txt

file transfer protocol is used to transfer file contents from old file to the existing file

File is transferred........