ASSIGNMENT NO:3 **DATE:14/02/2017**

PROGRAM TITLE:Develop a Client Server Application using TCP/IP where the client will send a filename to the server and the server will display the content of the file. Otherwise display a message "File doesn't exist on the server."
Note: The file is situated on the Server side.

PROGRAM CODE:

server.c

```
#include<stdio.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<stdlib.h>
#include<string.h>

#define MAXPENDING 5
#define RECVBUFSIZE 20

void display_file(const char *file_name)
{
	printf("\n\tThe file you requested is:");
	puts(file_name);
	FILE *f = fopen(file_name, "r");//Open the specified file in read only mode
	if(f != NULL)
	{
		printf("\n\tIts contents are::\n");
		int c;
		while ((c = fgetc(f)) != EOF)//Read character from file until EOF
		{
			putchar(c); //Output character
		}
		fclose(f);
	}
	else
	{
		printf("\n\tThe file doesn't exist on this server.\n");
	}
}
main()
{
	int servSock, clientAddrLen, clientSock, recvBufSize;
	float res;
	struct sockaddr_in clientAddr,serverAddr;
	char server_ip[] = "127.0.0.1";
	unsigned short server_port=25051;
	char recvBuf[RECVBUFSIZE];
	bzero(&serverAddr,sizeof(serverAddr));
	serverAddr.sin_family = AF_INET;//Internet Address family
	serverAddr.sin_port = htons(server_port);//Local Port address
	inet_aton(server_ip,(&serverAddr.sin_addr));
	if((servSock=socket(AF_INET,SOCK_STREAM,0))<0)
	{
		printf("\n\tSocket Error.\n");
		exit(1);
	}
	printf("\n\tSERVER: Socket Created.\n");
	if((bind(servSock,(struct sockaddr*)&serverAddr, sizeof(serverAddr)))<0)//-1
indicates failure
	{
		printf("\n\tBind Error.\n");
```

```
            close(servSock);//Closing the socket
            exit(1);
      }
      printf("\n\tSERVER: Binded Successfully.\n");
      if(listen(servSock,MAXPENDING)<0)//-1 indicates failure
      {
            printf("\n\tListen Error.\n");
            close(servSock);//Closing the socket
            exit(1);
      }
      printf("\n\tSERVER: Listening to Clients..\n\tPress Ctrl+C to stop the 
server.\n");
      while(1)//Run forever
      {
            clientAddrLen = sizeof(clientAddr);
            if((clientSock=accept(servSock,(struct sockaddr 
*)&clientAddr,&clientAddrLen))<0)
            {
                  printf("\n\tAccept Error.\n");
                  close(servSock);
                  exit(1);
            }
            if(recvBufSize=recv(clientSock,recvBuf,RECVBUFSIZE,0)<0)
            {
                  printf("\n\tReceive Error.\n");
                  close(clientSock);
                  continue;
            }
            display_file(recvBuf);
            close(clientSock);
      }
      close(servSock);
}
```

client.c

```
#include<stdio.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<stdlib.h>
#include<string.h>

#define BUFSIZE 20

main()
{
      int clientSock;
      struct sockaddr_in serverAddr;
      char server_ip[] = "127.0.0.1";
      unsigned short server_port=25051;
      char sendBuf[BUFSIZE];
      printf("\n\tEnter the filename:");
      gets(sendBuf);
      printf("\n\tYou asked for the file:");
      puts(sendBuf);
      bzero(&serverAddr,sizeof(serverAddr));
      serverAddr.sin_family = AF_INET;//Internet Address family
      serverAddr.sin_port = htons(server_port);//Local Port address
      inet_aton(server_ip,(&serverAddr.sin_addr));
      if((clientSock=socket(PF_INET,SOCK_STREAM,0))<0)
      {
            printf("\n\tSocket Error.\n");
            exit(1);
      }
      printf("\n\tCLIENT: Socket Created.\n");
```

```
	if((connect(clientSock,(struct sockaddr*)&serverAddr,sizeof(serverAddr)))<0)
	{
		printf("\nConnect Error\n");
		close(clientSock);
		exit(1);
	}
	printf("\n\tCLIENT: Connected.\n");
	if(write(clientSock,sendBuf,sizeof(sendBuf))<0)
	{
		printf("\n\tSend Error.\n");
		exit(1);
	}
	printf("\n\tCLIENT: Sent.\n");
	close(clientSock);
}
```

OUTPUT:

Server

```
[student@localhost 3]$ ./server

	SERVER: Socket Created.

	SERVER: Binded Successfully.

	SERVER: Listening to Clients..
	Press Ctrl+C to stop the server.

	The file you requested is:filename.txt

	Its contents are::
This is the testfile.
This file contains random lines.
Okay this is the end of this file.

	The file you requested is:nonexisting.txt

	The file doesn't exist on this server.
^C
```

Client

```
[student@localhost 3]$ ./server

	SERVER: Socket Created.

	SERVER: Binded Successfully.

	SERVER: Listening to Clients..
	Press Ctrl+C to stop the server.

	The file you requested is:filename.txt

	Its contents are::
This is the testfile.
This file contains random lines.
Okay this is the end of this file.

	The file you requested is:nonexisting.txt

	The file doesn't exist on this server.
```