PROGRAM TITLE:Develop a Client Server Application using TCP/IP where the client will input user's information like name, age, date of birth and % of marks in 10+2 examination and send them to the server. The server will display the user's information.

PROGRAM CODE:
server.c

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

#define MAXPENDING 5
#define RECVBUFSIZE 20

main()
{
      int servSock, clientAddrLen, clientSock, recvBufSize,i;
      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);
            }
            for(i=0;i<4;i++)
```

```c
                {
                        if(recvBufSize=recv(clientSock,recvBuf,RECVBUFSIZE,0)<0)
                        {
                                printf("\n\tReceive Error.\n");
                        }
                        else
                        {
                                if(i==0)
                                        printf("\n\tUser Name:%s",recvBuf);
                                else if(i==1)
                                        printf("\n\tUser Age:%s",recvBuf);
                                else if(i==2)
                                        printf("\n\tUser DOB(DD/MM/YYYY):%s",recvBuf);
                                else
                                        printf("\n\tUser Percentage:%s",recvBuf);
                                printf("\n");
                        }
                }
                close(clientSock);
                continue;
                close(clientSock);
        }
        close(servSock);
}
```

client.c
```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
sendBufName[BUFSIZE],sendBufAge[BUFSIZE],sendBufDOB[BUFSIZE],sendBufPer[BUFSIZE];
        printf("\n\tEnter User Details:\n\tName:");
        gets(sendBufName);
        printf("\tAge:");
        gets(sendBufAge);
        printf("\tDate of Birth(DD/MM/YYYY):");
        gets(sendBufDOB);
        printf("\tPercentage in 10+2 exam:");
        gets(sendBufPer);
        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,sendBufName,sizeof(sendBufName))<0)
        {
                printf("\n\tSend Error Name.\n");
                exit(1);
        }
        if(write(clientSock,sendBufAge,sizeof(sendBufAge))<0)
        {
                printf("\n\tSend Error Age.\n");
                exit(1);
        }
        if(write(clientSock,sendBufDOB,sizeof(sendBufDOB))<0)
        {
                printf("\n\tSend Error DOB.\n");
                exit(1);
        }
        if(write(clientSock,sendBufPer,sizeof(sendBufPer))<0)
        {
                printf("\n\tSend Error Percentage.\n");
                exit(1);
        }
        printf("\n\tCLIENT: Sent.\n");
        close(clientSock);
}
```

OUTPUT:

<u>Server</u>

```
[student@localhost 4]$ ./server

        SERVER: Socket Created.

        SERVER: Binded Successfully.

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

        User Name:Soumik De

        User Age:21

        User DOB(DD/MM/YYYY):06/12/1995

        User Percentage:92
^C
```

<u>Client</u>

```
[student@localhost 4]$ ./client

        Enter User Details:
        Name:Soumik De
        Age:21
        Date of Birth(DD/MM/YYYY):06/12/1995
        Percentage in 10+2 exam:92

        CLIENT: Socket Created.

        CLIENT: Connected.

        CLIENT: Sent.
```