ECHO:

Server.c

#include <stdio.h>

#include <sys/types.h>

#include <netinet/in.h>

#include <string.h>

#include <time.h>

int main()

{

int sd, sd2, nsd, clilen, sport, len;

int port;

time\_t ticks;

char sendmsg[100], rcvmsg[100];

struct sockaddr\_in servaddr, cliaddr;

printf("Enter the server port: \n");

scanf("%d", &sport);

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

ticks = time(NULL);

if (sd < 0)

printf("Can't create\n");

else

printf("Socket is created");

printf(" at %d\n", sport);

servaddr.sin\_family = AF\_INET;

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

servaddr.sin\_port = htons(sport);

sd2 = bind(sd, (struct sockaddr \*)&servaddr, sizeof(servaddr));

if (sd2 < 0)

printf("Can't bind\n");

else

printf("Binded\n");

listen(sd, 5);

clilen = sizeof(cliaddr);

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

if (nsd < 0)

printf("can't accept\n");

else

printf("Accepted\n");

int Flag = 1;

while (Flag != 0)

{

char Hell[100];

printf("Server: ");

scanf(" %[^\n]",Hell);

if (Hell[0] == 'S' && Hell[1] == 'P')

{

Flag = 0;

break;

}

strcpy(sendmsg, Hell);

send(nsd, sendmsg, 100, 0);

printf("Msg received from Client: ");

recv(nsd, rcvmsg, 100, 0);

printf("%s\n", rcvmsg);

}

}

Client.c

#include <stdio.h>

#include <sys/types.h>

#include <netinet/in.h>

#include <string.h>

int main()

{

int csd, cport, len;

char sendmsg[100], revmsg[100];

struct sockaddr\_in servaddr;

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

scanf("%d", &cport);

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

if (csd < 0)

printf("Can't create\n");

else

printf("socket is created");

printf(" at %d\n", cport);

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)) == -1)

printf("Can't Connect\n");

else

printf("Connected successfully\n");

int Flag = 1;

char Hell[100];

while (Flag != 0)

{

recv(csd, revmsg, 100, 0);

strcpy(Hell,revmsg);

if (Hell[0] == 'S' && Hell[1] == 'P')

{

Flag = 0;

break;

}

printf("Server Msg: %s\n", Hell);

strcpy(sendmsg, revmsg);

send(csd, sendmsg, 100, 0);

}

}

Output:

