ARYAMAN MISHRA

19BCE1027

CLIENT

#include<sys/types.h>

#include<sys/socket.h>

#include<stdio.h>

#include<netinet/in.h>

#include <unistd.h>

#include<string.h>

#include<strings.h>

#include <arpa/inet.h>

//#define buffsize 150

void main()

{

int b,sockfd,sin\_size,con,n,len,sport;

//char buff[256];

char operator;

int op1,op2,result;

printf("Enter server port\n");

scanf("%d",&sport);

if((sockfd=socket(AF\_INET,SOCK\_STREAM,0))>0)

printf("socket created sucessfully\n");

//printf("%d\n", sockfd);

struct sockaddr\_in servaddr;

servaddr.sin\_family=AF\_INET;

servaddr.sin\_addr.s\_addr=inet\_addr("127.0.0.1");

servaddr.sin\_port=htons(sport);

sin\_size = sizeof(struct sockaddr\_in);

if((con=connect(sockfd,(struct sockaddr \*) &servaddr, sin\_size))==0); //initiate a connection on a socket

printf("connect sucessful\n");

printf("Enter operation:\n +:Addition \n -: Subtraction \n /: Division \n\*:Multiplication \n");

scanf(" %c",&operator);

printf("Enter operands:\n");

scanf("%d %d", &op1, &op2);

write(sockfd,&operator,10);

write(sockfd,&op1,sizeof(op1));

write(sockfd,&op2,sizeof(op2));

read(sockfd,&result,sizeof(result));

printf("Operation result from server=%d\n",result);

close(sockfd);

}

SERVER

#include<sys/types.h>

#include<sys/socket.h>

#include<stdio.h>

#include<netinet/in.h>

#include <unistd.h>

#include<string.h>

#include <arpa/inet.h>

void main()

{

int b,sockfd,connfd,sin\_size,l,n,len,sport;

char operator;

int op1,op2,result;

printf("Enter server port\n");

scanf("%d",&sport);

if((sockfd=socket(AF\_INET,SOCK\_STREAM,0))>0)

printf("socket created sucessfully\n"); //socket creation

//printf("%d\n", sockfd); //on success 0 otherwise -1

struct sockaddr\_in servaddr;

struct sockaddr\_in clientaddr;

servaddr.sin\_family=AF\_INET;

servaddr.sin\_addr.s\_addr=inet\_addr("127.0.0.1");

servaddr.sin\_port=htons(sport);

if((bind(sockfd, (struct sockaddr \*)&servaddr,sizeof(servaddr)))==0)

printf("bind sucessful\n");

//printf("%d\n",b);

if((listen(sockfd,5))==0) //listen for connections on a socket

printf("listen sucessful\n");

//printf("%d\n",l);

sin\_size = sizeof(struct sockaddr\_in);

if((connfd=accept(sockfd,(struct sockaddr \*)&clientaddr,&sin\_size))>0);

printf("accept sucessful\n");

//printf("%d\n",connfd);

read(connfd, &operator,10);

read(connfd,&op1,sizeof(op1));

read(connfd,&op2,sizeof(op2));

switch(operator) {

case '+': result=op1 + op2;

printf("Result is: %d + %d = %d\n",op1, op2, result);

break;

case '-':result=op1 - op2;

printf("Result is: %d - %d = %d\n",op1, op2, result);

break;

case '\*':result=op1 \* op2;

printf("Result is: %d \* %d = %d\n",op1, op2, result);

break;

case '/':result=op1 / op2;

printf("Result is: %d / %d = %d\n",op1, op2, result);

break;

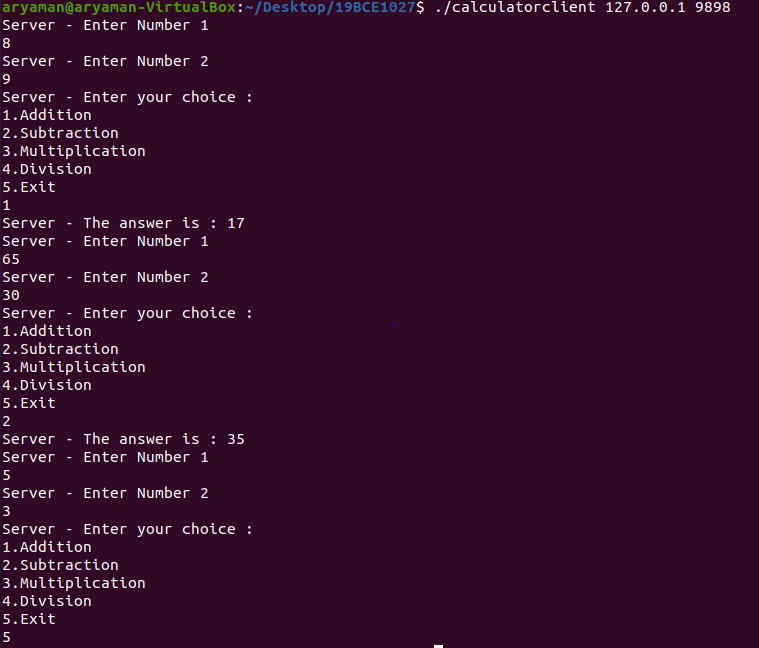
default:

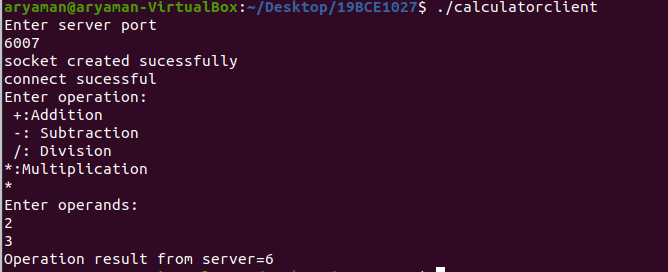
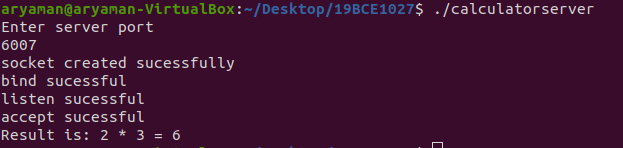
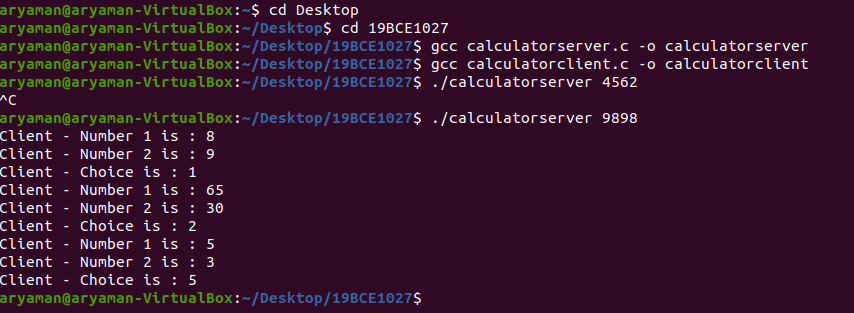
printf("ERROR: Unsupported Operation");

}

write(connfd,&result,sizeof(result));

close(sockfd);

}



2)SERVER

#include<stdio.h>

#include<sys/types.h>

#include<netinet/in.h>

#include<string.h>

#include<unistd.h>

int main()

{

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

char sendmsg[100],rcvmsg[100];

struct sockaddr\_in servaddr,cliaddr;

printf("Enter server port\n");

scanf("%d",&sport);

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

if(sd<0)

printf("Can't 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(sport);

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

if(sd2<0)

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

else

printf("\n Binded\n");

listen(sd,5);

clilen=sizeof(cliaddr);

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

if(nsd<0)

printf("Cant accept \n");

else

printf("Accept \n");

do

{

recv(nsd,rcvmsg,100,0);

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

fgets(sendmsg,100,stdin);

len=strlen(sendmsg);

sendmsg[len-1]='\0';

send(nsd,sendmsg,100,0);

}

while(strcmp(sendmsg,"bye")!=0);

close(sd);

close(nsd);

}

CLIENT

#include<stdio.h>

#include<sys/types.h>

#include<netinet/in.h>

#include<string.h>

#include<unistd.h>

int main()

{

int csd,cport,len;

char sendmsg[100],rcvmsg[100];

struct sockaddr\_in servaddr,cliaddr;

printf("Enter server port\n");

scanf("%d",&cport);

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

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

if(csd<0)

printf("cant create\n");

else

printf("Socket is created\n");

servaddr.sin\_family=AF\_INET;

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

servaddr.sin\_port=htons(cport);

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

printf("cant connect");

else

printf("Connected");

do

{

fgets(sendmsg,100,stdin);

len=strlen(sendmsg);

sendmsg[len-1]='\0';

send(csd,sendmsg,100,0);

recv(csd,rcvmsg,100,0);

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

}

while(strcmp(rcvmsg,"bye")!=0);

close(csd);

}

