basic client and server program using tcp (client sending string and server receiving data from client,port number and ip address providing through command line arguments)

Header.h

--------------

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

#include<sys/types.h>

#include<sys/socket.h>

#include<unistd.h>

#include<netinet/in.h>

#include<netinet/ip.h>

#include<arpa/inet.h>

// server.c

#include"header.h"

void client\_info(struct sockaddr\_in client\_info)

{

printf("client port address: %d\n",client\_info.sin\_port);

printf("client ip address: %s\n",inet\_ntoa(client\_info.sin\_addr));

}

int main(int argc,char \*\*argv)

{

if(argc!=3)

{

printf("usage: ./server server\_port server\_ip\_address\n");

return 0;

}

int sfd,len,nsfd;

struct sockaddr\_in server\_id,client\_id;

/\*\*\*\*\*\*\*\*\*\* socket () \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

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

if(sfd<0)

{

perror("Socket");

return 0;

}

perror("Socket");

/\*\*\*\*\*\*\*\*\*\*\* bind() \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

server\_id.sin\_family=AF\_INET;

server\_id.sin\_port=htons(atoi(argv[1]));

server\_id.sin\_addr.s\_addr=inet\_addr(argv[2]);

len=sizeof(server\_id);

if(bind(sfd,(struct sockaddr\*) &server\_id,len)<0)

{

perror("Bind");

return 0;

}

perror("Bind");

/\*\*\*\*\*\*\*\*\*\*\*\*\* listen \*\*\*\*\*\*\*\*\*\*\*\*\*\*/

if(listen(sfd,5)<0)

{

perror("Listen");

return 0;

}

perror("Listen");

/\*\*\*\*\*\*\*\*\*\*\*\* accept \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

printf("before accept client request...(waiting for client)\n");

nsfd=accept(sfd,(struct sockaddr\*)&client\_id,&len);

if(nsfd<0)

{

perror("Accept");

return 0;

}

printf("client request Accepted...\n");

client\_info(client\_id);

char s[20];

read(nsfd,s,sizeof(s)); // reading data from client

printf("client data: %s\n",s);

close(sfd);

close(nsfd);

}

//client.c

#include"header.h"

int main(int argc,char \*\*argv)

{

if(argc!=3)

{

printf("usage: ./client server\_port server\_ip\_address\n");

return 0;

}

int sfd,len;

struct sockaddr\_in server\_id;

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

if(sfd<0)

{

perror("Socket");

return 0;

}

perror("Socket");

server\_id.sin\_family=AF\_INET;

server\_id.sin\_port=htons(atoi(argv[1]));

server\_id.sin\_addr.s\_addr=inet\_addr(argv[2]);

len=sizeof(server\_id);

if(connect(sfd,(struct sockaddr\*)&server\_id,len)<0)

{

perror("connect");

return 0;

}

perror("connect");

char s[20];

printf("enter string from user\n");

scanf("%s",s);

write(sfd,s,strlen(s)+1); //writing data to server

close(sfd);

}