client

#include <stdio.h>

{

printf("\n\* Simulation packet data corrupted or incomplete");

printf("\n Sending RETRANSMIT for packet 1");

memset(&data,0,sizeof(data));

sprintf(data, "R1");

if (send(sockfd, data, strlen(data), 0)==-1)

{

printf("\nError in sending RETRANSMIT...");

exit(1);}

firstTime=0;

}

else

{

wait--;

if(!wait)

{

printf("\n\*\*\*packet Accepted -> Sending ACK");

wait=3;

memset(&data, 0, sizeof(data));

sprintf(data, "A");

digit[0]=(char) (currentPacket + 48);

digit[1]='\0';

strcat(data, digit);

send(sockfd, data, strlen(data),0);

}

}

}

while(currentPacket !=9);

printf("\nAll packets received...Exiting");

close(sockfd);

return(0);

}

#include <stdlib.h>

#include <string.h>

#include <netdb.h>

#include <sys/types.h>

#include <net.net/in.h>

#include <sys/socket.h>

#define PORT 3599

int main()

{

int sockfd, newSockFd, size, firstTime=1,currentPacket, wait=3;

char data[100], digit[2];

struct sockaddr\_in client;

memset(&client, 0, sizeof(client));

sockfd-socket (AF INET, SOCK STREAM, 0);

if(sockfd=-1)

{

} printf("Error in socket creation...");

else

{ printf("\nSocket Created...");

}

client.sin\_family=AF\_INET;

client.sin port=PORT;

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

printf("\nStarting up...");

size=sizeof(client);

printf("\nEstablishing Connection");

if (connect(sockfd, (struct sockaddr \*)&client, size)==-1)

{

printf("\nError in connecting to server...");

exit(1);

}

else

{ printf("\nConnection Established!");}

memset(&data, e, sizeof(data));

sprintf(data, "REQUEST");

if(send(sockfd, data, strlen(data),0)=-1)

{ printf("Error in send request for data...");

exit(1);

}

do

{ memset(&data, 0,sizeof (data));

recv(sockfd, data, 100,0);

currentPacket=atoi(data):

printf("\nGot packet:%d",currentPacket);

if (currentPacket=3&&firstTime)