**Readme**

第七組

組長：509411026 程毓盛

組員：509411008 蔡宜紋、509411011 魯恆均、509411018 莊詩婷

Table of Contents

[2 運作說明： 1](#_Toc86785043)

[3 步驟: 1](#_Toc86785044)

[4 使用函式庫： 2](#_Toc86785045)

[4.1 socket 2](#_Toc86785046)

[4.2 bind 2](#_Toc86785047)

[4.3 sendto 2](#_Toc86785048)

[4.4 recvfrom 3](#_Toc86785049)

[4.5 close 3](#_Toc86785050)

[4.6 fflush 3](#_Toc86785051)

[4.7 bzero 3](#_Toc86785052)

# 運作說明：

使用 LINUX環境C語言寫一對程式，分別為UDP的client端與server端

且server端在收到client端的字串後，會自動回傳相同的字串給client端

# 步驟:

1. 執行server程式, 等候
2. 執行client程式, 輸入傳送至server字串, ex.[ABC]
3. Server端接收到字串[ABC],再回傳相同字串[ABC]給Client端
4. Client端收到[ABC]字串

# 使用函式庫：

## socket

int socket(int domain,int type, int protocol)

返回值:

非負值成功, -1 出錯

其中family指明瞭協議族/域,通常AF\_INET、AF\_INET6、AF\_LOCAL等;Type是套介面型別,主要SOCK\_STREAM、SOCK\_DGRAM、SOCK\_RAW;

protocol一般取為0。成功時,返回一個小的非負整數值,與檔案描述符類似。此例使用socket()來建立一個UDP socket，第二個引數為SOCK\_DGRAM

## bind

Int bind(int sockfd,const struct sockaddr\* myaddr,socklen\_t addrlen)

返回值:

0成功, -1 出錯

當socket函式返回一個描述符時,只是存在於其協議族的空間中,並沒有分配一個具體的協議地址(這裡指IPv4/IPv6和埠號的組合),bind函式可以將一組固定的地址繫結到sockfd上。

其中sockfd是socket函式返回的描述符;myaddr指定了想要繫結的IP和埠號,均要使用網路位元組序-即大端模式;addrlen是前面struct sockaddr(與sockaddr\_in等價)的長度。

## sendto

sendto(int sockfd,const void \*buf, size\_t len, int flags, const struct sockaddr \*dst\_addr, socklen\_t addrlen);

由於無連線的資料報模式下,沒有建立連線,需指明目的地址,addrlen通常為sizeof(sockaddr)的長度。成功時返回傳送的位元組數,失敗返回-1。

返回值:

>0成功, -1 出錯

## recvfrom

recvfrom(int sockfd,void \*buf, size\_t len, int flags,struct sockaddr \*src\_addr, socklen\_t \*addrlen)

sockfd:接收端套接字描述

buf:用於接收資料的應用緩衝區地址

len:指名緩衝區大小

flags:通常為0

src\_addr:資料來源端的地址

addrlen:src\_addr地址的長度

注意後兩個引數是輸出引數,其中addrlen既是輸入又是輸出引數,即值-結果引數,需要在呼叫時,指明src\_addr的長度。另外,如果不關心資料傳送端的地址,可以將後兩者均設定為NULL。

## close

close預設功能是將套接字作“已關閉”標記,並立即返回到呼叫程序,該套接字描述符不能再為該程序所用

## fflush

fflush:是把c庫中的緩衝呼叫write函式寫到磁碟[其實是寫到核心的緩衝區

## bzero

sin\_zero用來將sockaddr\_in結構填充到與struct sockaddr同樣的長度，可以用bzero()或memset()函數將其置為零。

中正資工 網路程式設計 HW2 509411018 莊詩婷 2021/11/2

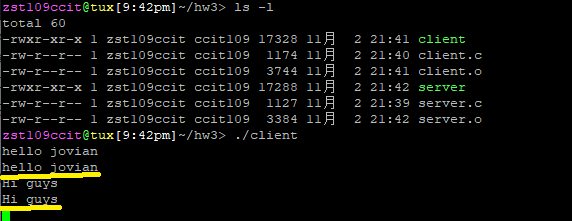
一．程式: 如附檔HW2\_Group7\_509411018.rar

執行結果

Server:



Client:



二．利用Wireshark抓到UCP的封包

