

網路程式設計 HW-廣播網路程式作業

一、執行畫面

練習 1：基本廣播示範(client, server)

此程式 server 每秒傳送一則訊息給 client，client 隨時接收 server 傳送的訊息。

[一對一] 左方為 server 端的執行畫面，右方為 client 端的執行畫面，啟動左方的 server 後，再啟動右方的 client，當 client 關閉後，server 仍正常運作，當 client 再次啟動，則繼續接收訊息。

```

C:\data\Code\Cpp\network_programming\cmake-build-debug (main -> origin)
λ .\broadcast_basic_server.exe
server start broadcasting on port: 5678
server broadcast: 11111111111111111111
server broadcast: 22222222222222222222
server broadcast: 33333333333333333333
server broadcast: 44444444444444444444
server broadcast: 55555555555555555555
server broadcast: 66666666666666666666
server broadcast: 77777777777777777777
server broadcast: 88888888888888888888
server broadcast: 99999999999999999999
server broadcast: 00000000000000000000
server broadcast: 11111111111111111111
server broadcast: 22222222222222222222
server broadcast: 33333333333333333333
server broadcast: 44444444444444444444
server broadcast: 55555555555555555555
server broadcast: 66666666666666666666
server broadcast: 77777777777777777777
server broadcast: 88888888888888888888
server broadcast: 99999999999999999999
server broadcast: 00000000000000000000
server broadcast: 11111111111111111111

C:\data\Code\Cpp\network_programming\cmake-build-debug (main -> origin)
λ .\broadcast_basic_client.exe
client start listen broadcasting on port: 5678
client from 192.168.56.1: 22222222222222222222
client from 192.168.56.1: 33333333333333333333
client from 192.168.56.1: 44444444444444444444
client from 192.168.56.1: 55555555555555555555
client from 192.168.56.1: 66666666666666666666
client from 192.168.56.1: 77777777777777777777
client from 192.168.56.1: 88888888888888888888
^C
C:\data\Code\Cpp\network_programming\cmake-build-debug (main -> origin)
λ .\broadcast_basic_client.exe
client start listen broadcasting on port: 5678
client from 192.168.56.1: 33333333333333333333
client from 192.168.56.1: 44444444444444444444
client from 192.168.56.1: 55555555555555555555
client from 192.168.56.1: 66666666666666666666
client from 192.168.56.1: 77777777777777777777
client from 192.168.56.1: 88888888888888888888
client from 192.168.56.1: 99999999999999999999
client from 192.168.56.1: 00000000000000000000
client from 192.168.56.1: 11111111111111111111
  
```

[多對一] 左上方為第 1 個 server 端的執行畫面，左下方為第 2 個 server 端的執行畫面，右方為 client 端的執行畫面，先啟動 client，再啟動左上方的 server，3 秒後再啟動左下方的 server，可看到 client 可收到所有 server 的廣播封包。

```

C:\data\Code\Cpp\network_programming\cmake-build-debug (main -> origin)
λ .\broadcast_basic_server.exe
server start broadcasting on port: 5678
server broadcast: 11111111111111111111
server broadcast: 22222222222222222222
server broadcast: 33333333333333333333
server broadcast: 44444444444444444444
server broadcast: 55555555555555555555
server broadcast: 66666666666666666666
server broadcast: 77777777777777777777
server broadcast: 88888888888888888888
server broadcast: 99999999999999999999
server broadcast: 00000000000000000000
server broadcast: 11111111111111111111
server broadcast: 22222222222222222222
server broadcast: 33333333333333333333
server broadcast: 44444444444444444444
server broadcast: 55555555555555555555
server broadcast: 66666666666666666666

C:\data\Code\Cpp\network_programming\cmake-build-debug (main -> origin)
λ .\broadcast_basic_server.exe
server start broadcasting on port: 5678
server broadcast: 11111111111111111111
server broadcast: 22222222222222222222
server broadcast: 33333333333333333333
server broadcast: 44444444444444444444
server broadcast: 55555555555555555555
server broadcast: 66666666666666666666
server broadcast: 77777777777777777777
server broadcast: 88888888888888888888
server broadcast: 99999999999999999999
server broadcast: 00000000000000000000
server broadcast: 11111111111111111111
server broadcast: 22222222222222222222
server broadcast: 33333333333333333333

C:\data\Code\Cpp\network_programming\cmake-build-debug (main -> origin)
λ .\broadcast_basic_client.exe
client start listen broadcasting on port: 5678
client from 192.168.56.1: 11111111111111111111
client from 192.168.56.1: 22222222222222222222
client from 192.168.56.1: 33333333333333333333
client from 192.168.56.1: 11111111111111111111
client from 192.168.56.1: 44444444444444444444
client from 192.168.56.1: 22222222222222222222
client from 192.168.56.1: 55555555555555555555
client from 192.168.56.1: 33333333333333333333
client from 192.168.56.1: 66666666666666666666
client from 192.168.56.1: 44444444444444444444
client from 192.168.56.1: 77777777777777777777
client from 192.168.56.1: 55555555555555555555
client from 192.168.56.1: 88888888888888888888
client from 192.168.56.1: 66666666666666666666
client from 192.168.56.1: 99999999999999999999
client from 192.168.56.1: 77777777777777777777
client from 192.168.56.1: 00000000000000000000
client from 192.168.56.1: 88888888888888888888
client from 192.168.56.1: 11111111111111111111
client from 192.168.56.1: 99999999999999999999
client from 192.168.56.1: 22222222222222222222
client from 192.168.56.1: 00000000000000000000
client from 192.168.56.1: 33333333333333333333
client from 192.168.56.1: 11111111111111111111
client from 192.168.56.1: 44444444444444444444
client from 192.168.56.1: 22222222222222222222
client from 192.168.56.1: 55555555555555555555
client from 192.168.56.1: 33333333333333333333
client from 192.168.56.1: 66666666666666666666
  
```

[一對多] 左方一樣為 server 端的執行畫面，右上方為本機 client 端的執行畫面，右下方為其他電腦 client 端的執行畫面，先啟動所有的 client，再啟動 server，可看到所有的 client 皆可收到廣播封包的訊息。

```

C:\data\Code\Cpp\network_programming\cmake-build-debug (main -> origin)
λ .\broadcast_basic_server.exe
server start broadcasting on port: 5678
server broadcast: 11111111111111111111
server broadcast: 22222222222222222222
server broadcast: 33333333333333333333
server broadcast: 44444444444444444444

C:\data\Code\Cpp\network_programming\cmake-build-debug (main -> origin)
λ .\broadcast_basic_client.exe
client start listen broadcasting on port: 5678
client from 192.168.56.1: 11111111111111111111
client from 192.168.56.1: 22222222222222222222
client from 192.168.56.1: 33333333333333333333
client from 192.168.56.1: 44444444444444444444

C:\data\Code\Cpp\network_programming\cmake-build-debug (main -> origin)
λ .\broadcast_basic_client.exe
client start listen broadcasting on port: 5678
client from 192.168.56.1: 11111111111111111111
client from 192.168.56.1: 22222222222222222222
client from 192.168.56.1: 33333333333333333333
client from 192.168.56.1: 44444444444444444444
  
```

練習 2：Server 同時傳送多種資料

此程式 server 每秒傳送三則訊息給 client，client 隨時接收 server 傳送的訊息。

左方為 server 端的執行畫面，右方為 client 端的執行畫面，啟動右方的 client 後，再啟動左方的 server，可看到 client 端可接收到訊息。

```

C:\data\Code\Cpp\network_programming\cmake-build-debug (main -> origin)
λ .\broadcast_multiple_data_server.exe
server start broadcasting on port: 5678
server broadcast: 11111111111111111111
server broadcast: bbbbbb11111111111111111111
server broadcast: 888888888888888888888888
server broadcast: 222222222222222222222222
server broadcast: cccccccccccccccccccccccc
server broadcast: CCCCCCCCCCCCCCCCCCCCCCCC
server broadcast: 3333333333333333333333333333
server broadcast: dddddddddddddddddddddddddd
server broadcast: DDDDDDDDDDDDDDDDDDDDDDDDDDDDD
server broadcast: 4444444444444444444444444444
server broadcast: eeeeeeeeeeeeeeeeeeeeeeeeee
server broadcast: EEEEEEEEEEEEEEEEEEEEEEEE
server broadcast: 5555555555555555555555555555
server broadcast: ffffffffffffffffffffffffffffffff
server broadcast: FFFFFFFFFFFFFFFFFFFFFFFFFFFF
server broadcast: 6666666666666666666666666666
server broadcast: 9999999999999999999999999999
server broadcast: GGGGGGGGGGGGGGGGGGGGGGGGGGGGG
server broadcast: 77777777777777777777777777777777
server broadcast: hhhhhhhhhhhhhhhhhhhhhhhhhhhhhhh
server broadcast: HHHHHHHHHHHHHHHHHHHHHHHHHHHHHH
server broadcast: 88888888888888888888888888888888
server broadcast: iiiiiiiiiiiiiiiiiiiiiiiiiiiiii
server broadcast: IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
server broadcast: 99999999999999999999999999999999
server broadcast: jjjjjjjjjjjjjjjjjjjjjjjjjjjjjjj
server broadcast: JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ
server broadcast: 00000000000000000000000000000000
server broadcast: kkkkkkkkkkkkkkkkkkkkkkkkkkkkkkk
server broadcast: KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK
server broadcast: 1111111111111111111111111111111111
server broadcast: lllllllllllllllllllllllllllllllll
server broadcast: LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL
server broadcast: 2222222222222222222222222222222222
server broadcast: mmmmmmmmmmmmmmmmmmmmmmmmmmmmmmm
server broadcast: MMMMMMMMMMMMMMMMMMMMMMMMMMMMMM

C:\data\Code\Cpp\network_programming\cmake-build-debug (main -> origin)
λ .\broadcast_basic_client.exe
client start listen broadcasting on port: 5678
client from 192.168.56.1: 11111111111111111111
client from 192.168.56.1: bbbbbb11111111111111111111
client from 192.168.56.1: 888888888888888888888888888888
client from 192.168.56.1: 2222222222222222222222222222
client from 192.168.56.1: cccccccccccccccccccccccc
client from 192.168.56.1: CCCCCCCCCCCCCCCCCCCCCCCC
client from 192.168.56.1: 33333333333333333333333333333333
client from 192.168.56.1: dddddddddddddddddddddddddd
client from 192.168.56.1: DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
client from 192.168.56.1: 44444444444444444444444444444444
client from 192.168.56.1: eeeeeeeeeeeeeeeeeeeeeeeeee
client from 192.168.56.1: EEEEEEEEEEEEEEEEEEEEEEEE
client from 192.168.56.1: 55555555555555555555555555555555
client from 192.168.56.1: ffffffffffffffffffffffffffffffff
client from 192.168.56.1: FFFFFFFFFFFFFFFFFFFFFFFFFFFF
client from 192.168.56.1: 66666666666666666666666666666666
client from 192.168.56.1: 99999999999999999999999999999999
client from 192.168.56.1: GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
client from 192.168.56.1: 77777777777777777777777777777777
client from 192.168.56.1: hhhhhhhhhhhhhhhhhhhhhhhhhhhhhhh
client from 192.168.56.1: HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH
client from 192.168.56.1: 88888888888888888888888888888888
client from 192.168.56.1: iiiiiiiiiiiiiiiiiiiiiiiiiiiiii
client from 192.168.56.1: IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
client from 192.168.56.1: 99999999999999999999999999999999
client from 192.168.56.1: jjjjjjjjjjjjjjjjjjjjjjjjjjjjjjj
client from 192.168.56.1: JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ
client from 192.168.56.1: 00000000000000000000000000000000
client from 192.168.56.1: kkkkkkkkkkkkkkkkkkkkkkkkkkkkkkk
client from 192.168.56.1: KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK
client from 192.168.56.1: 1111111111111111111111111111111111
client from 192.168.56.1: lllllllllllllllllllllllllllllllll
client from 192.168.56.1: LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL
client from 192.168.56.1: 2222222222222222222222222222222222
client from 192.168.56.1: mmmmmmmmmmmmmmmmmmmmmmmmmmmmmmm
client from 192.168.56.1: MMMMMMMMMMMMMMMMMMMMMMMMMMMMMM

```

練習 3：Client 過濾接收的資料(模仿選台)

此程式 server 每秒傳送三則訊息給 client，client 可選擇要進入的頻道，並隨時接收 server 傳送的訊息，根據頻道編號顯示出對應的訊息。

左方為 server 端的執行畫面，右方為 client 端的執行畫面，啟動右方的 client 後，再啟動左方的 server。

首先，先在 client 端選擇第 2 個頻道，可看到 client 端可接收到小寫字母 a-z 的訊息。接下來，在 client 端選擇第 3 個頻道，可看到 client 端可接收到大寫字母 A-Z 的訊息。最後，在 client 端選擇第 1 個頻道，可看到 client 端可接收到數字 0-9 的訊息，如此重複執行。

```

server broadcast: sssssssssssssssssss
server broadcast: SSSSSSSSSSSSSSSSSSS
server broadcast: 9999999999999999999
server broadcast: tttttttttttttttttttttt
server broadcast: TTTTTTTTTTTTTTTTTTTTT
server broadcast: 000000000000000000000
server broadcast: uuuuuuuuuuuuuuuuuuuuuu
server broadcast: UUUUUUUUUUUUUUUUUUUUU
server broadcast: 111111111111111111111
server broadcast: vvvvvvvvvvvvvvvvvvvvvv
server broadcast: VVVVVVVVVVVVVVVVVVVVV
server broadcast: 222222222222222222222
server broadcast: wwwwwwwwwwwwwwwwwwwwww
server broadcast: WWWWWWWWWWWWWWWWWWWWW
server broadcast: 333333333333333333333
server broadcast: xxxxxxxxxxxxxxxxxxxxxxx
server broadcast: XXXXXXXXXXXXXXXXXXXXX
server broadcast: 444444444444444444444
server broadcast: yyyyyyyyyyyyyyyyyyyyy
server broadcast: YYYYYYYYYYYYYYYYYYY
server broadcast: 555555555555555555555
server broadcast: zzzzzzzzzzzzzzzzzzzzzz
server broadcast: ZZZZZZZZZZZZZZZZZZZZZ
server broadcast: 666666666666666666666
server broadcast: aaaaaaaaaaaaaaaaaaaaaa
server broadcast: AAAAAAAAAAAAAAAAAAAAAA
server broadcast: 777777777777777777777
server broadcast: bbbbbbbbbbbbbbbbbbbbbb
server broadcast: BBBBBBBBBBBBBBBBBBBB
server broadcast: 888888888888888888888
server broadcast: cccccccccccccccccc
server broadcast: CCCCCCCCCCCCCCCCCCCC
server broadcast: 999999999999999999999
server broadcast: dddddddddddddddddd
server broadcast: DDDDDDDDDDDDDDDDDDDDD
server broadcast: 000000000000000000000
server broadcast: eeeeeeeeeeeeeeee
server broadcast: EEEEEEEEEEEEEEEEEEE
server broadcast: 111111111111111111111
server broadcast: ffffffffffffffffffff
server broadcast: FFFFFFFFFFFFFFFFFFFF

C:\data\Code\Cop\network_programming\cmake-build-debug (main -> origin)
λ .\broadcast_change_channel_client.exe
client start listen broadcasting on port: 5678
Channel(1-3): 2
Enter channel: 2
client from 192.168.56.1: bbbbbbbbbbbbbbbbbbb
client from 192.168.56.1: cccccccccccccccccc
client from 192.168.56.1: dddddddddddddddddd
client from 192.168.56.1: eeeeeeeeeeeeeeeeee
client from 192.168.56.1: ffffffffffffffffffff
client from 192.168.56.1: gggggggggggggggggggg
client from 192.168.56.1: hhhhhhhhhhhhhhhhhhhh
client from 192.168.56.1: iiiiiiiiiiiiiiiiiiii
client from 192.168.56.1: jjjjjjjjjjjjjjjjjjjj
client from 192.168.56.1: kkkkkkkkkkkkkkkkkkk
Channel(1-3): 3
Enter channel: 3
client from 192.168.56.1: LLLLLLLLLLLLLLLLLLL
client from 192.168.56.1: MMMMMMMMMMMMMMMMMM
client from 192.168.56.1: NNNNNNNNNNNNNNNNNNN
client from 192.168.56.1: OOOOOOOOOOOOOOOOOO
client from 192.168.56.1: PPPPPPPPPPPPPPPPPPPP
client from 192.168.56.1: QQQQQQQQQQQQQQQQQQ
client from 192.168.56.1: RRRRRRRRRRRRRRRRRR
client from 192.168.56.1: SSSSSSSSSSSSSSSSSSS
client from 192.168.56.1: TTTTTTTTTTTTTTTTTT
client from 192.168.56.1: UUUUUUUUUUUUUUUUUU
Channel(1-3): 1
Enter channel: 1
client from 192.168.56.1: 11111111111111111111
client from 192.168.56.1: 22222222222222222222
client from 192.168.56.1: 33333333333333333333
client from 192.168.56.1: 44444444444444444444
client from 192.168.56.1: 55555555555555555555
client from 192.168.56.1: 66666666666666666666
client from 192.168.56.1: 77777777777777777777
client from 192.168.56.1: 88888888888888888888
client from 192.168.56.1: 99999999999999999999
client from 192.168.56.1: 00000000000000000000
Channel(1-3): |
  
```

練習 4：Client 使用 thread 作出選台效果

此程式 server 每秒傳送三則訊息給 client，client 可選擇要進入的頻道，並隨時接收 server 傳送的訊息。

左方為 server 端的執行畫面，右方為 client 端的執行畫面，啟動右方的 client 後，再啟動左方的 server。

client 可隨時切換要進入的頻道，輸出的訊息會立即做更換。

```

C:\data\Code\Cpp\network_programming\cmake-build-debug (main -> origin)
λ .\broadcast_multiple_data_server.exe
server start broadcasting on port: 5678
server broadcast: 11111111111111111111
server broadcast: bbbbbb1111111111111111
server broadcast: 00000000000000000000
server broadcast: 22222222222222222222
server broadcast: cccccccccccccccccccc
server broadcast: CCCCCCCCCCCCCCCCCC
server broadcast: 33333333333333333333
server broadcast: ddddddddddddddddddd
server broadcast: DDDDDDDDDDDDDDDDDDD
server broadcast: 44444444444444444444
server broadcast: eeeeeeeeeeeeeeeeeee
server broadcast: EEEEEEEEEEEEEEEEEEE
server broadcast: 55555555555555555555
server broadcast: ffffffffffffffffffffff
server broadcast: FFFFFFFFFFFFFFFFFFFF
server broadcast: 66666666666666666666
server broadcast: gggggggggggggggggggggg
server broadcast: GGGGGGGGGGGGGGGGGGGG
server broadcast: 77777777777777777777
server broadcast: hhhhhhhhhhhhhhhhhhhhh
server broadcast: HHHHHHHHHHHHHHHHHHHH
server broadcast: 88888888888888888888
server broadcast: iiiiiiiiiiiiiiiiiiiiii
server broadcast: IIIIIIIIIIIIIIIIIIIII
server broadcast: 99999999999999999999
server broadcast: jjjjjjjjjjjjjjjjjjjjj
server broadcast: JJJJJJJJJJJJJJJJJJJJ
server broadcast: 00000000000000000000
server broadcast: kkkkkkkkkkkkkkkkkkkkk
server broadcast: KKKKKKKKKKKKKKKKKKKK
server broadcast: 11111111111111111111
server broadcast: llllllllllllllllllllll
server broadcast: LLLLLLLLLLLLLLLLLLLLL
server broadcast: 22222222222222222222
server broadcast: mmmmmmmmmmmmmmmmmmm
server broadcast: MMMMMMMMMMMMMMMMMMMM

C:\data\Code\Cpp\network_programming\cmake-build-debug (main -> origin)
λ .\broadcast_thread_channel_client.exe
client start listen broadcasting on port: 5678
1
Enter channel: 1
client from 192.168.56.1: 11111111111111111111
client from 192.168.56.1: 22222222222222222222
client from 192.168.56.1: 33333333333333333333
3
Enter channel: 3
client from 192.168.56.1: EEEEEEEEEEEEEEEEEEE
client from 192.168.56.1: FFFFFFFFFFFFFFFFFFFF
client from 192.168.56.1: GGGGGGGGGGGGGGGGGGGG
1
Enter channel: 1
client from 192.168.56.1: 77777777777777777777
2
Enter channel: 2
client from 192.168.56.1: iiiiiiiiiiiiiiiiiiiiii
client from 192.168.56.1: jjjjjjjjjjjjjjjjjjjj
1
Enter channel: 1
client from 192.168.56.1: 00000000000000000000
2
Enter channel: 2
client from 192.168.56.1: llllllllllllllllllllll
client from 192.168.56.1: mmmmmmmmmmmmmmmmmmm
|

```


練習 5：用不同 port 實現頻道與選台(client, server)

程式 server 每秒從每個 port 傳送一則訊息給 client，client 可選擇要進入的頻道，並隨時接收 server 傳送的訊息。

左方為 server 端的執行畫面，右方為 client 端的執行畫面，啟動右方的 client 後，再啟動左方的 server。

client 可隨時切換要進入的頻道，輸出的訊息會立即做更換。首先，先在 client 端選擇第 1 個頻道，可看到 client 端可接收到數字 0-9 的訊息。接下來，在 client 端選擇第 2 個頻道，可看到 client 端可接收到小寫字母 a-z 的訊息。最後，在 client 端選擇第 3 個頻道，可看到 client 端可接收到大寫字母 A-Z 的訊息，如此重複執行。

```

λ .\broadcast_channel_port_server.exe
server start broadcasting
(1) server broadcast on 5678 port: 11111111111111111111
(2) server broadcast on 5679 port: bbbbbbbbbbbbbbbbbbbb
(3) server broadcast on 5680 port: 88888888888888888888

(1) server broadcast on 5678 port: 22222222222222222222
(2) server broadcast on 5679 port: cccccccccccccccccc
(3) server broadcast on 5680 port: CCCCCCCCCCCCCCCCCC

(1) server broadcast on 5678 port: 33333333333333333333
(2) server broadcast on 5679 port: dddddddddddddddddd
(3) server broadcast on 5680 port: DDDDDDDDDDDDDDDDDDD

(1) server broadcast on 5678 port: 44444444444444444444
(2) server broadcast on 5679 port: eeeeeeeeeeeeeeeeee
(3) server broadcast on 5680 port: EEEEEEEEEEEEEEEEEE

(1) server broadcast on 5678 port: 55555555555555555555
(2) server broadcast on 5679 port: ffffffffffffffffffff
(3) server broadcast on 5680 port: FFFFFFFFFFFFFFFFFF

(1) server broadcast on 5678 port: 66666666666666666666
(2) server broadcast on 5679 port: gggggggggggggggggggg
(3) server broadcast on 5680 port: GGGGGGGGGGGGGGGGGG

(1) server broadcast on 5678 port: 77777777777777777777
(2) server broadcast on 5679 port: hhhhhhhhhhhhhhhhhhhh
(3) server broadcast on 5680 port: HHHHHHHHHHHHHHHHHH

(1) server broadcast on 5678 port: 88888888888888888888
(2) server broadcast on 5679 port: iiiiiiiiiiiiiiiiii
(3) server broadcast on 5680 port: IIIIIIIIIIIIIIIIIII

(1) server broadcast on 5678 port: 99999999999999999999
(2) server broadcast on 5679 port: jjjjjjjjjjjjjjjjjjjj
(3) server broadcast on 5680 port: JJJJJJJJJJJJJJJJJJ

(1) server broadcast on 5678 port: 00000000000000000000
(2) server broadcast on 5679 port: kkkkkkkkkkkkkkkkkkkk
(3) server broadcast on 5680 port: KKKKKKKKKKKKKKKKKK

C:\data\Code\Cop\network_programming\cmake-build-debug (main -> origin)
λ .\broadcast_channel_port_client.exe
client start listen broadcasting
1
Enter channel: 1
recvfrom() failed: 10004
client from 0.0.0.0:
client from 192.168.56.1: 11111111111111111111
client from 192.168.56.1: 22222222222222222222
client from 192.168.56.1: 33333333333333333333
2
Enter channel: 2
recvfrom() failed: 10004
client from 192.168.56.1: 33333333333333333333
client from 192.168.56.1: eeeeeeeeeeeeeeeeee
client from 192.168.56.1: ffffffffffffffffffff
3
Enter channel: 3
recvfrom() failed: 10004
client from 192.168.56.1: ffffffffffffffffffff
client from 192.168.56.1: GGGGGGGGGGGGGGGGGG
client from 192.168.56.1: HHHHHHHHHHHHHHHHHH
1
Enter channel: 1
recvfrom() failed: 10004
client from 192.168.56.1: HHHHHHHHHHHHHHHHHH
client from 192.168.56.1: 88888888888888888888
client from 192.168.56.1: 99999999999999999999
2
Enter channel: 2
recvfrom() failed: 10004
client from 192.168.56.1: 99999999999999999999
client from 192.168.56.1: kkkkkkkkkkkkkkkkkkkk
|

```

二、心得

在實作這次作業的過程中，我遇到了以下幾個問題。

第一個是我發現老師範例程式碼有用到 `memset` 函式的地方可能會有些問題，因為程式最上方 `define` 的傳送最大值 `MAXLINE` 定義為 20，可是 `memset` 函式帶入要修改的長度為 `MAXLINE`，也就是 20，這樣子一來會導致一個問題，就是這個 `char array` 沒有地方存放結尾字元 `\0`，所以會導致輸出時的句子出現了一些奇怪的情況，像是多輸出了方塊□的豆腐字元，或是 `o` 之類的字元等，只要將 `sizeof(str)` 改為 `sizeof(str) - 1` 少輸出一個字元即可解決此問題。

可參考 `stack overflow` 的這篇討論串：

<https://stackoverflow.com/questions/62359687/does-memset-change-the-array-length-when-another-array-is-memset>

另外一個是第五題當 `client` 端在切換頻道時，關閉 `socket` 再連線 `socket` 的期間，需要花一些時間，會導致暫時性的發生 `recvfrom` 函式無法正常運作的情況，如上方第五題右邊的截圖，我想到的解決方式是可以設定一個 `flag` 來判斷 `socket` 是否連接，在 `socket` 要關閉前，先將 `flag` 設為 `false`，等待連線成功，再將 `flag` 設為 `true`，若要執行 `recvfrom` 函式之前，需先判斷 `flag` 是否為 `true` 的狀態。

最後另外一個小建議是第五題老師 `client` 端的程式碼利用 `switch case` 判斷條件時，僅須把 `port` 的設定放在裡面即可，下面的 `bind` 可放在 `switch case` 之後，而在 `switch case` 中加入 `default` 的條件，當輸入要切換的頻道不符合規範時，可以就不改動 `port` 的設定，或是直接結束程式，這樣子可以讓程式碼更簡潔。

備註：以上所有的程式碼皆放置在我的 `GitHub repo`，詳見以下連結網址

https://github.com/linwebs/network_programming#week15