

程式執行方法：

```
$ python3 main
```

實作方法：

本作業以 python (python 3)完成：

1. 首先，從"config"檔讀入指定之 channel (CHAN)
2. 在設定好 bot 所需的參數後，傳送指定資料以連接至指定 server (irc.freenode.net, port = 6667, channel = CHAN)。
3. 連接上後，對聊天室傳送 "Hello! I am robot." (PRIVMSG to CHAN)

```
17:23 -!- iceBot00 [~icewolf@linux8.csie.ntu.edu.tw] has joined #CN_DEMO
17:23 < iceBot00> Hello! I am robot.
```

4. 接下來便是以 while 迴圈不斷聽取聊天室的內容，並依照特定內容做出不同回應：

- a. 若收到來自系統的 "PING" (IRC 用它來確認 robot 是否在掛機)，則回覆系統 "PONG"，以避免被系統踢出聊天室。

```
if "PING" in str(IRCMsg):
    IRCsocket.send(bytes("PONG :pingis\n", "UTF-8"))
```

- b. 若收到" @repeat"指令，則回傳@repeat 後面的所有內容(直到該句子結束，\r\n)

```
17:25 <@b04902099> @repeat hello world!
17:25 < iceBot00> hello world!
```

- c. 若收到" @convert"指令，則會將 10(16)進位數字轉成 16(10)進位，若 repeat 後面的字串並非 10 或 16 進位之數字，則不予回應。

```
17:25 <@b04902099> @convert 0x19
17:25 < iceBot00> 25
17:25 <@b04902099> @convert 666
17:25 < iceBot00> 0x29a
```

- d. 若收到"@ip"指令，則進入遞迴 function (searchip) 中判斷出所有符合 ipv4 的 ip address，計算總數後，一並輸出至聊天室。

```
17:26 <@b04902099> @ip 12345
17:26 < iceBot00> 4
17:26 < iceBot00> 1.2.3.45
17:26 < iceBot00> 1.2.34.5
17:26 < iceBot00> 1.23.4.5
17:26 < iceBot00> 12.3.4.5
```

- e. 若收到"@help"指令，則回傳" @repeat <Message>\r\n"、" @convert <Number>\r\n"、" @ip <String>"至聊天室。

```
17:26 <@b04902099> @help
17:26 < iceBot00> @repeat <Message>
17:26 < iceBot00> @convert <Number>
17:26 < iceBot00> @ip <String>
```

遇到的挑戰與解決方法：

1. 一開始直接依照 tutorial 講義中的範例 code，無法連接至目標聊天室，需要設定好 nick 以及 user 後，才能正確連上聊天室。
2. 以 python 撰寫程式，需先將要傳出的字串轉換成 bytes 後再 send。

參考資料：

Create an IRC Bot with Python 3

<https://linuxacademy.com/blog/geek/creating-an-irc-bot-with-python3/>