

# CNHW1- IRC Robot Report

---

工海三 B05505004 朱信歷

## Program structure

---

由於聊天機器人他需要同時應付多個使用者，同時也有可能需要從standard input 讀聊天的內容，因此我是透過select這個函式並且將timeout設成無限大來實踐他。

另外我同時還有維護兩個list，分別是正在聊天，以及正在猜數字的使用者。如果有人正在聊天，那麼select function就會同時等待socket或是stdin的輸入，那如果沒有則單純等待socket的輸入。

當socket有東西傳進來，bot會先比對是不是server傳來的PING，再來比對有無在聊天或是猜數字的list裡面。如果在裡面則會直接進入相對應的函式，不讓使用者在完成前使用別的功能。接著才會判斷 `!guess`、`!song`、`!chat`、星座等指令做出相對應的回覆。

除此之外由於有可能與多個使用者同時進行一對一的聊天，因此當聊天的對象大於一時，可以透過指令：`<nick> <message>` 來指定訊息要傳送的對象。那如果沒有指定則預設傳給第一個聊天的對象。

## Challenge & Solution

---

在進行這項作業我首先遇到的困難是編碼的問題，雖然以前有處理過編碼的問題，但這次不知道為什麼還是弄了蠻久的，後來發現是如果你的程式碼函有非 UTF-8 的編碼就需要在程式碼最前面加上encoding declaration，像是這次的繁體中文就需要宣告 `# coding=Big5`。除此之外如果是使用者輸入涵有中文又是另外一種處理方式，在傳的時候要用 `byte(message,"UTF-8")` 的方式將字串轉成Byte objects，讀取時則是用 `decode("UTF-8")` 將Byte objects轉回字串，如此才不會出現編碼的問題。

再來是在寫爬蟲時，因為我以前有寫過一些HTML，所以一開始就信誓旦旦地想說一定可以很快就寫完，一看瀏覽器按 `F12` 後影片標題的html都有 `id="video-title"` 就想說用beautifulsoup抓這個id準沒錯，誰知道弄了很久都抓不到，還以為是套件的問題。後來研究他回傳的pure html才發現根本沒有這個tag，推測他後來可能是用javascript 的方式把它改掉的，所以在最後生成的HTML碼找不到。最後我是搜尋 `"/watch?v="` 來找有存網址的tag。

## Reflections

---

在寫這次作業前我原本以為應該很快就可以寫完，畢竟只要用先前系統程式設計學過的多工，以及簡單的if-else判斷即可。可是沒想到真的寫起來其實還是有蠻多細節需要注意的，像是我前面提到的編碼以及爬蟲的細節。另外透過這次作業其實我也真正的實踐Client-server架構的應用，同時也更了解socket programming實做的一些內容、技巧，算是有蠻大的收穫。