

# Mideterm review

Class5 - Ascii Code Converter

從0開始的思考方向(參考即可)

r09941007

# 第一步-理解程式需要具備哪些功能?

觀察範例的輸入輸出, 可統整出程式需具備:

1. 一開始提示使用者輸入一個字元.
2. 讀入使用者輸入後, 若非僅一字元, 需印出錯誤訊息後結束.
3. 印出該字元對應的ascii code.
4. 判斷該字元到底是符號, 大小寫字母, 還是數字.

## 第二步-構想第一步列出功能的先後順序 (流程圖 or 虛擬碼)

因為我懶得畫圖, 這邊採用pseudo code的方式

```
void AsciiConverter(){  
    If 輸入 is invalid  
        印出錯誤訊息  
    Return // return 代表此函式結束  
    Int asciiCode = 讀入字元相對應的ascii編碼  
    印出該字元的asciiCode  
    印出該字元是數字or大小寫字母or符號  
}
```

## 第三步-開始想各個功能(子問題)如何完成

1. 判斷輸入是否invalid:因為讀入的是字串, 也許可以利用...
2. 印出資訊與使用者互動:有學過的Console.WriteLine()可以做到!
3. 得到字元的asciiCode:找找看C#是否有轉換的方法
4. 判斷是哪種字元:也許我可以窮舉? 也許我可以利用asciiCode?

# 個人建議

第一步的具備哪些功能的**分析能力非常重要**，  
這種將**大問題分成小問題**的思考方式，不只是學習程式重要，現實生或的種種工作/  
學習/下決定也都會用到。

要應用在程式設計上面，還會**取決於你對電腦以及語法的熟悉程度**。  
你需要去思考電腦可以做哪些事情，這些事情**要如何排列組合**才能達到你要的效果。

# 結語

雖然前幾頁打了很多字，  
但當你熟練後，看到題目腦中就會自動去做上述的規畫惹。  
也不用一一列出來，自然而然的就會知道該怎麼做。

但這種思考方式，當你遇到不熟悉or龐大的問題時，  
還是很重要的，**一開始的規劃越有條理，**  
**後面實作程式時就會越順暢，越有效率。**

遇到不會的就**找資料**，看看別人的作法，能不能達到你要的結果  
然後**多練習**。會自然的訓練你的邏輯。