

# 語音識別與自然語言處理 HW4

執行結果：

```
PS E:\FUC-Speech-Recognition-and-NLP\HW4> .\min_Edit_Distance.exe
Top 5 sentences with minimum edit distance:
Candidate #0: intention ## MED 3 ## OptPath: inten++tion
Candidate #1: intonation ## MED 4 ## OptPath: int*n+ation
Candidate #2: intentions ## MED 4 ## OptPath: inten+*i-o-n-s-
Candidate #3: infestation ## MED 5 ## OptPath: in*e*t-ation
Candidate #4: inextension ## MED 5 ## OptPath: ine-x-tens++ion
Time taken: 45000.000000 us
```

程式描述：

我使用自訂的結構來儲存原始字串和最短路徑的字串，並且在結構裡面會保存他的 MED cost。

## 主程式

首先宣告 5 個自訂的結構(`WordCost wordCosts[MAX_WORDS]`)，使用 `while` 迴圈每次讀入一行直到 `EOF`，針對每一個字執行講義中的 DP 演算法，並且每次執行完判斷他是不是前 5 小的 `cost`，在判斷 `cost` 的時候是每次都會將那五個自訂的結構的 `cost` 使用 `quick sort` 排序過，以此加快查詢速度。

然後遍歷完整個 `txt` 檔案後，將最五小的 `cost` 的字再根據保存的他的 `dp` 的每一步驟的表，由表的右上查到左下，根據刪除、新增、替換來去給予原始字串對應的符號，透過 `memmove` 將字串向右移動一位，然後在原本該字元位子天加符號(+ - \*)。