

陳奕嘉 Alvin Tan

## 程式設計作業 Bonus 題

### 題目：鏈接串列

實現方法：

這裏先展示輸出結果。

```
List 1: 12 43 56 34 98
List 2: 36 77 99

Combined List: 12 43 56 34 98 36 77 99

Choose one mode:
(1) Enter a new node (2) Exit : 1
Please enter a number to insert as a new node: 17
Done! The new list is : 17 12 43 56 34 98 36 77 99

Choose one mode:
(1) Enter a new node (2) Exit : 1
Please enter a number to insert as a new node: 19
Done! The new list is : 19 17 12 43 56 34 98 36 77 99

Choose one mode:
(1) Enter a new node (2) Exit : 2

Length of list: 10

Reversed List: 99 77 36 98 34 56 43 12 17 19
Press any key to continue . . .
```

i) 首先在主程式中輸入兩個陣列的資料。要形成鏈接串列就要讓數值帶有指標，便定義一個指向結構，具有 data 和指標\*next。在 createList 定義第一個節點后就把 array 的資料一個個串起來形成 linked list。

ii) 爲了把兩個串列連接起來，在 combineList 中從串列 1 的第一個節點一直跑找串列 1 的末節點，把指標指向串列 2 的首節點，把兩個串列連接起來。

iii) 要在串列第一個數字前增加增加新節點，要給 insertFirstNode 這個函數兩個參數，分別是第一個節點的位置和要增加的值。在函數中，把新增節點指向原來第一個節點，再把新增節點定義為第一個節點。

iv) 找 listLength 的方法如同(ii)，在合并後的串列一個個執行，每次 list\_len 變數增加一，直到末節點。

v) 這步驟比較複雜，先附圖。

```
Done! The new list is : 19 17 12 43 56 34 98 36 77 99

Choose one mode:
(1) Enter a new node (2) Exit : 2
Length of list: 10

17 12 43 56 34 98 36 77 99
12 43 56 34 98 36 77 99
43 56 34 98 36 77 99
56 34 98 36 77 99
34 98 36 77 99
98 36 77 99
36 77 99
77 99
99
List is empty.
Reversed List: 99 77 36 98 34 56 43 12 17 19
```

這部分在 `reverseList` 函數中運用三個節點 `prev`、`current`、`next` 和指標的轉換來操作。先將原串列第一個節點設置為 `current`，它前後都是 `null`，可以看作是製造另一個反過來的串列。然後用 `next` 來暫存 `current` 的指標，因為接著要改變 `current` 的 `next` 指標指向原串列前一個值。然後就一直改變 `current` 如上圖，重複以上步驟知道 `null`，由后往前的概念構造新的串列。

### 心得：

對於鏈接串列這章節，我認為是有一定的難度，尤其是指標的構造和轉換需要仔細揣摩才能搞懂。但學習之後，發現這種資料結構具有很高的靈活性，可以在串列前端、後端和中間插入新增資料。在內存空間上，不需要事先分配特定的空間，在新增資料當下便可在不相鄰的內存存放，并用指標鏈接起來，這便是鏈接串列的精髓。通過鏈接串列，可以很直觀地瞭解資料都在我們電腦內存裏發生了什麼，和用電腦的角度理解要怎麼下存放這些資料，是非常使用的課題。

### 參考資料：

<https://moodle.ncku.edu.tw/course/view.php?id=28947>

<https://chat.openai.com>