

第2次作業-作業-HW2

學號：112111219

姓名：陳恩偉

作業撰寫時間：270 (mins · 包含程式撰寫時間)

最後撰寫文件日期：2024/10/21

本份文件包含以下主題：(至少需下面兩項，若是有多者可以自行新增)

- ☒ 說明內容
- ☒ 個人認為完成作業須具備觀念

1. 問題如下圖所述，並回答下面問題。

Ans:

a. 有行及列，所以為二維陣列

```
alphabet = [  
    ['!', '@', '#', '$', '%', '^', '&', '*', '(', ')'],  
    ['Q', 'W', 'E', 'R', 'T', 'Y', 'U', 'I', 'O', 'P'],  
    ['A', 'S', 'D', 'F', 'G', 'H', 'J', 'K', 'L', ':'],  
    ['Z', 'X', 'C', 'V', 'B', 'N', 'M', '<', '>', '?'],  
]
```

b.
在二維陣列查找指定符號後依照指令移動後找出指定字源並輸出

2. 給定一個包含 n 個不同數字的數組，這些數字的範圍是從 0 到 n 。找出數組中缺失的那一個數字。

Ans:

```
def find(nums):  
    n = len(nums)  
    sum_total = n * (n + 1) // 2  
    sum_nums = sum(nums)  
    lost = sum_total - sum_nums  
    return lost  
  
nums1 = [3, 0, 1]  
nums2 = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12]
```

```
print(f"nums1的輸出: {find(nums1)}")
print(f"nums2的輸出: {find(nums2)}")
```

從0開始，所以可以利用 $1+2+n$ 去掉全部的總和得到缺失的那個數字

3. 請回答下面問題：

Ans:

a.
 設 $n = 10$
 $2^n = 2^{(10)} = 1024$
 $2^n + 1 = 2^{(10)} + 1 = 1024 + 1 = 1025$
 當 n 很大時， 2^n 會變得非常大，而 $+1$ 相對來說就很小了 那個 $+1$ 在 n 越大的情況下越可以忽略不計。
 $2^n + 1$ 的增長速度和 2^n 差不多，因此它們時間複雜度是相等的

b.
 設 $n = 10$
 $2^{\{2n\}} = 2^{\{20\}} = 1,048,576$
 $2n = 2 * 10 = 20$
 當 n 很大時， $2^{(2n)}$ 的增長速度遠遠超過 $2n$ 。這意味著 $2^{[2n]}$ 的時間複雜度是指數級別的，而 $O(2n)$ 是線性級別的。

4. 請問以下各函式，在進行呼叫後，請計算(1)執行次數 $T(n)$ ，並(2)透過執行次數判斷時間複雜度為何(請用Big-Oh進行表示)？

Ans:

a.
 (1) $n(n+1)/2$
 (2) $O(n^2)$

b.
 (1) $\log_2(n)$
 (2) $O(\log n)$

c.
 (1) $n*m$
 (2) $O(n*m)$

d.
 (1) $\log_2(n) * \log_2(m)$
 (2) $O(n \log n)$

個人認為完成作業須具備觀念

需要對程式運作、python語法有一定的了解，且需要了解時間複雜度及log的運算過程，熟悉python語法及程式的運算方式才可運算出執行次數及時間複雜度。

