

# Number Analysis — Final Project

楊育哲  
41143264

January 15, 2025

## CONTENTS

<b>1</b>	<b>Ex 13.3</b>	<b>2</b>
1.1	Introduction . . . . .	2
1.2	Source Code . . . . .	2
1.3	Result . . . . .	5
<b>2</b>	<b>GUI Project</b>	<b>6</b>
2.1	Introduction . . . . .	6
2.2	Source Code . . . . .	7
2.3	Result . . . . .	8
2.4	心得 . . . . .	10
2.5	References . . . . .	10

# CHAPTER 1

EX 13.3

## 1.1 Introduction

根據期末報告說明，尾數為 4 的應實作範例 13.3，並使用新版程式 (如範例 13.5、13.6 等) 撰寫。因此我參考了範例 13.3 和 13.6 兩程式。

## 1.2 Source Code

註：tex 的設定我還不太熟悉，程式部分的中文會顯示空白，例如 Label 那邊，不過可以從 Result 看出我的程式是沒問題的。

```
clear
mw0 = uifigure('Name', 'mw0');
mw0.Position = [30 60 450 650];
g1 = uigridlayout(mw0, [3 2]);
g1.RowHeight = {'fit', 'fit', '1X'};
g1.ColumnWidth = {'fit', '1x'};
imageLabel = uiimage(g1);
imageLabel.Layout.Row = 1;
imageLabel.Layout.Column = [1 2];
imageLabel.ImageSource = fullfile('cat.jpg');
mw1 = uiaxes(g1);
```

```
mw1.Layout.Row = 2;
mw1.Layout.Column = [1 2];
[x, y, z] = peaks;
f1 = surf1(mw1, x, y, z);
axis(mw1, [-3 3 -3 3 -8 8])
colormap(mw1, [0.5 0.5 0.5])
cdata = mw1.Colormap;
initcdata = cdata;
scroll1 = uicontrol('parent', mw0, ...
    'style', 'slider', ...
    'min', 0, ...
    'max', 1, ...
    'value', cdata(1), ...
    'position', [20 150 400 20], ...
    'callback', ['srl1k=get(scroll1, 'value');, ', ...
        'cdata(1)=srl1k;, ', ...
        'set(mw1, 'colormap', cdata);']);
scroll2 = uicontrol('parent', mw0, ...
    'style', 'slider', ...
    'min', 0, ...
    'max', 1, ...
    'value', cdata(2), ...
    'position', [20 100 400 20], ...
    'callback', ['srl2k=get(scroll2, 'value');, ', ...
        'cdata(2)=srl2k;, ', ...
        'set(mw1, 'colormap', cdata); ']);
scroll3 = uicontrol('parent', mw0, ...
    'style', 'slider', ...
    'min', 0, ...
    'max', 1, ...
    'value', cdata(3), ...
    'position', [20 50 400 20], ...
    'callback', ['srl3k=get(scroll3, 'value');, ', ...
        'cdata(3)=srl3k;, ', ...
        'set(mw1, 'colormap', cdata); ']);
Labell = uicontrol('parent', mw0, ...
    'style', 'text', ...
    'string', ' ', ...
    'position', [20 170 400 20]);
```

```
Label2 = uicontrol('parent', mw0, ...
    'style', 'text', ...
    'string', ' ', ...
    'position', [20 120 400 20]);

Label3 = uicontrol('parent', mw0, ...
    'style', 'text', ...
    'string', ' ', ...
    'position', [20 70 400 20]);

Label4 = uicontrol('parent', mw0, ...
    'style', 'text', ...
    'string', ' peaks ', ...
    'position', [10 450 150 20]);

Button1 = uicontrol('parent', mw0, ...
    'style', 'pushbutton', ...
    'string', 'Reset', ...
    'position', [120 5 60 40], ...
    'callback', ['set(scroll1, 'value', initcdata(1));, ', ...
        'set(scroll2, 'value', initcdata(2));, ', ...
        'set(scroll3, 'value', initcdata(3));, ', ...
        'set(mw1, 'colormap', initcdata);']);

Button2 = uicontrol('parent', mw0, ...
    'style', 'pushbutton', ...
    'string', 'Close', ...
    'position', [300 5 60 40], ...
    'callback', 'close');
```

---

### 1.3 Result

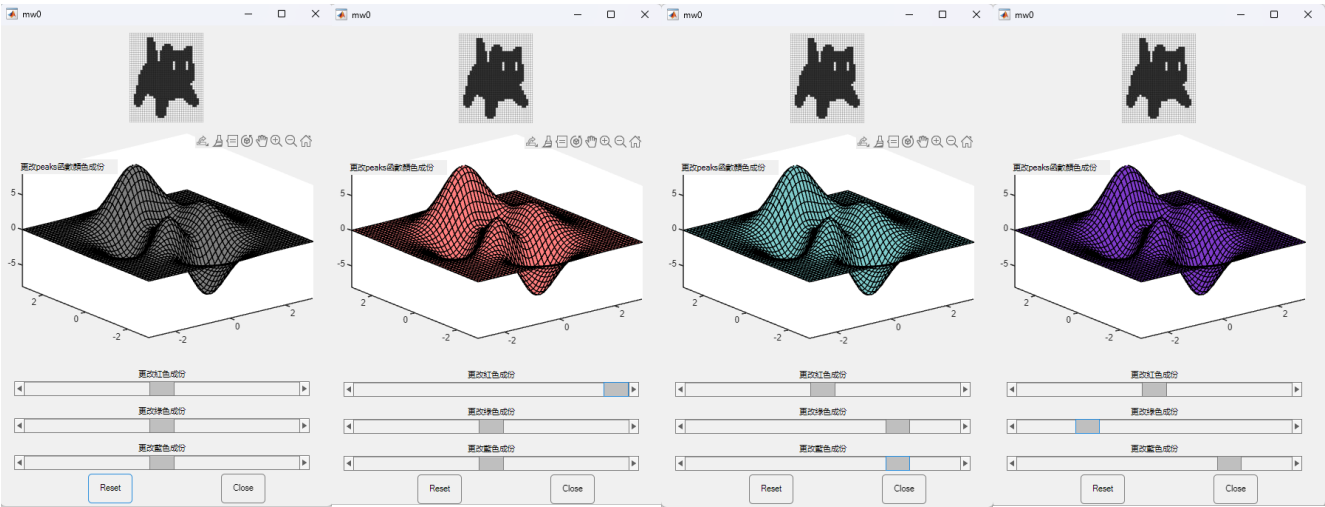


Figure 1.1: 三個狀況，證明程式運作正常

## CHAPTER 2

## GUI PROJECT

### 2.1 Introduction

題目中提到要自己常用的 GUI 工具，我想到的是 LeetCode 中的提交分析，參考下圖。



Figure 2.1: 我在 2025/01/14 Daily 的解題分析

可以看到圖中我框選了 ‘Analyze Complexity’，這項功能可以分析程式的複雜度。

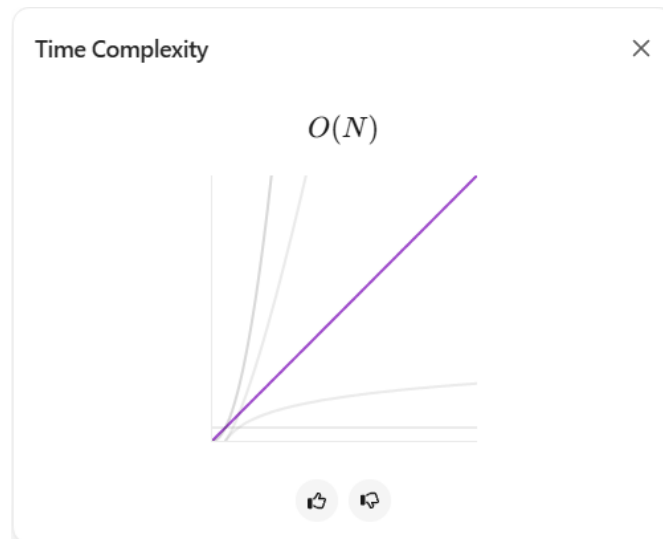


Figure 2.2: 時間複雜度分析顯示

做這項專題的動機在於：我們學校競程社群的解題平台是 codeforces 或瘋狂程設，而這兩個都沒有程式碼分析的功能。就先前的狀況，可以發現學弟妹解題不注重複雜度問題。再者，聽說社群有打算建一個解題系統，我想有一個分析複雜度的工具對於學弟妹應該有幫助。

另外，我不清楚 LeetCode 分析複雜度的細節，但感覺是透過不同測資去測執行時間之類的，再以測資和對應的時間去推敲複雜度，這讓我想到之前教的曲線近似，因此打算以此想法進行實作，最後提一下要分析複雜度的程式會在外部執行多個測資，並將執行結果傳給 matlab 端，所以 matlab 端的輸入是一不限長度且寬度為 2(測資與執行時間) 的陣列。

## 2.2 Source Code

程式碼請參考一併附上的原始碼，或可以參考我的 Github repo (kiol1812/Time-Complexity-Analysis)。我的程式會呼叫執行 python 檔案與 g++ 編譯指令，使用的 python 版本是 3.11，因為我所使用的 matlab 是 R2023b，它有支援 3.11。我怕執行會有環境問題，所以 GitHub 端有放錄製好的 Demo 狀況。



## 2.3 Result

以下放一些截圖說明成果。第一個頁面，可以看到題目的敘述，左側有一整籃的題目，不過這部分我沒有實作，而是花心力在專研以一項測試資料作為展示。第二個頁面則是上傳程式碼與查看測試的頁面。最後，第三頁是分析所上傳的程式的頁面，

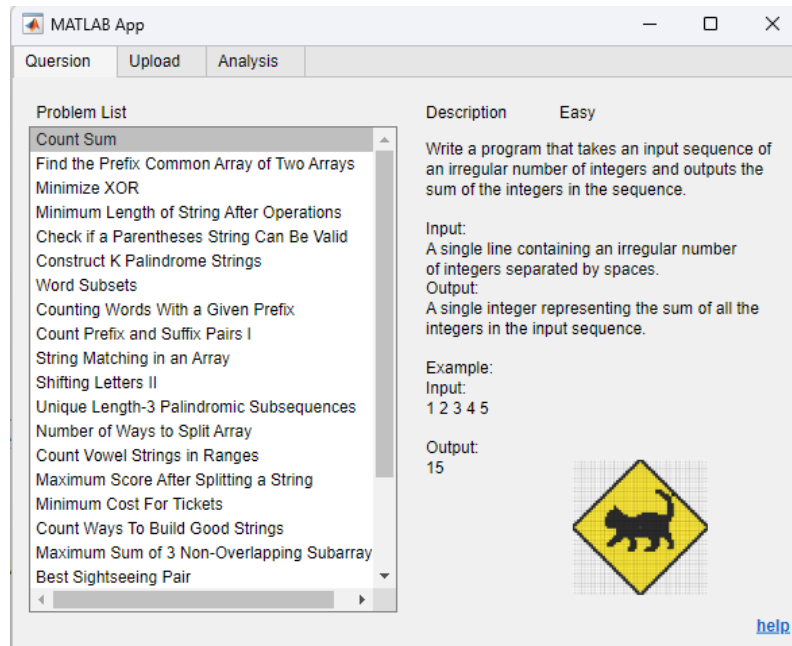


Figure 2.3: 頁面 1, 題目敘述

頁面二有兩個按鈕，分別是隨機測資和上傳，隨機測資按鈕 (Random Testcase) 會隨機撈取測試輸入輸出，並顯示在 Input 和 Output 的文字方塊中。而上傳 (Upload!) 則會將左方輸入的程式碼輸入至 source.cpp 並將其編譯成 1.exe，再透過逐一的測資比對，最後輸出結果到頁面三 (注意其中有編譯過程，雖然上方有語言可以選擇，但我只做了 c++ 的，其編譯是用 g++，這也可能有環境問題)。

頁面三顯示了上傳結果，如左上會有通過幾筆測試資料、上傳時間等，左下則是測試資料 (n) 對應時間，橘色的點是每筆測資的結果，而後方有一條近似曲線，我以多項式 4 去進行 polyfit (因為應該沒有人會寫  $N^5$  以上的程式，不確定邏輯有沒有錯)。

可以觀察到趨勢是呈現  $O(n)$ ，與預期結果相同。另外，右方是 logs，會顯示每筆測試資料是否通過與所花費時間。

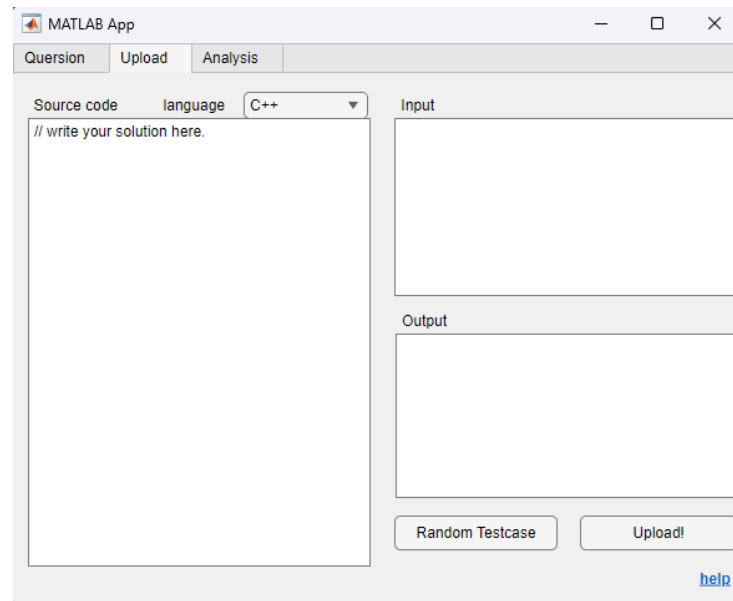


Figure 2.4: 頁面 2, 程式碼上傳

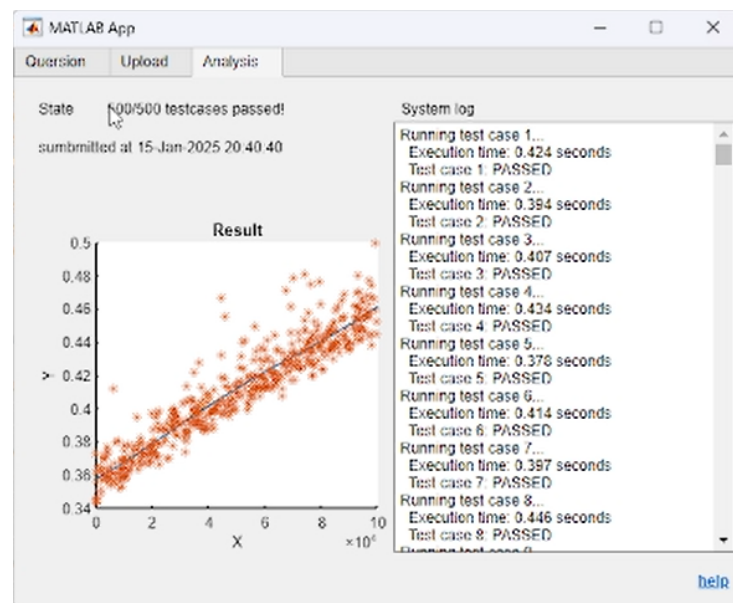


Figure 2.5: 頁面 3, 測試結果分析

## 2.4 心得

本次期末專題實作完成了我以前就想寫的複雜度分析工具，不過環境問題令我有點擔憂，不確定使用者是否能順利使用。另外目前撰寫的成果沒有彈性空間，未撰寫換題目、新增題目等功能。若無法執行，請參考我 GitHub 中放的執行影片。

## 2.5 References

- Github: MATLAB 與 Python 整合實現
- Matlab: fscanf
- Matlab: readline