

# 資料結構 — HW

作者  
學號

November 20, 2024

# Contents

1	解題說明	2
2	程式實作	3
3	效能分析	4
4	測試與驗證	5
5	申論及開發報告	6

# Chapter 1

## 解題說明

以遞迴實作計算  $N$  階層的函式，已知階層計算公式如下：

$$N! = N + (N - 1)! = N + (N - 1) + \cdots + 2 + 1$$

由於我們得知遞迴函數屬性，因此實作程式如下：

```
1  int sigma(int n) {  
2      if (n < 0)  
3          throw "n < 0";  
4      else if (n <= 1)  
5          return n;  
6      return n + sigma(n - 1);  
7  }
```

Figure 1.1: sigma

# Chapter 2

## 程式實作

```
1  #include <iostream>
2  using namespace std;
3
4  int sigma(int n) {
5      if (n < 0)
6          throw "n < 0";
7      else if (n <= 1)
8          return n;
9      return n + sigma(n - 1);
10 }
11
12 int main() {
13     int result = sigma(3);
14     cout << result << '\n';
15 }
```

Figure 2.1: sum.cpp 實作細節

# Chapter 3

## 效能分析

$$f(n) = O(n)$$

### 時間複雜度

$$T(P) = n \times C$$

每層迴圈所需  $C$  時間、 $n$  次遞迴。

### 空間複雜度

$$S(P) = 1 \times n$$

1 個變數、 $n$  次遞迴。

# Chapter 4

## 測試與驗證

```
1  #include <iostream>
2  using namespace std;
3
4  int sigma() { ... }
5
6  int main() {
7      int result = sigma(3);
8      cout << result << '\n';
9  }
```

Figure 4.1: 主函式細節

```
1  $ g++ main.cpp -o main.exe && ./main.exe
2  6
```

Figure 4.2: shell 編譯指令與輸出結果

此函式遞迴終止條件為當  $n$  為 0 或 1，若欲求得  $3!$ ，則呼叫  $\text{sigma}(3)$ ，進入函式後，首先第一層  $n = 3 > 1$  所以回傳  $n + \text{sigma}(n-1)$ ，即  $3 + \text{sigma}(2)$ ，接著第二層計算  $\text{sigma}(2)$ ， $n = 2 > 1$ ，所以回傳  $2 + \text{sigma}(1)$ ，接下來到第三層時， $n = 1 \leq 1$ ，符合終止條件 ( $n \leq 1$ )，因此回傳  $n$ ，即 1。

$$\text{sigma}(3) = 3 + \text{sigma}(2) = 3 + 2 + \text{sigma}(1) = 3 + 2 + 1 = 6$$

# Chapter 5

## 申論及開發報告

您的報告內容於此...