

1.解題說明

這題需要使用寫出一個程式來執行阿克曼函數(Ackermann's function)，它的 m 值與 n 值只需很小的數字就能增長很快(m 為主)，而這次得用非遞迴的方式寫出這個程式。

$$A(m,n) = \begin{cases} n + 1 & , \text{ if } m = 0 \\ A(m - 1, 1) & , \text{ if } n = 0 \\ A(m - 1, A(m, n - 1)) & , \text{ otherwise} \end{cases}$$

2.效能分析

時間複雜度

根據 m 值變化

$$(0,n) = n + 1$$

$$(1,n) = n + 2$$

$$(2,n) = 2n + 3$$

.

.

.

空間複雜度

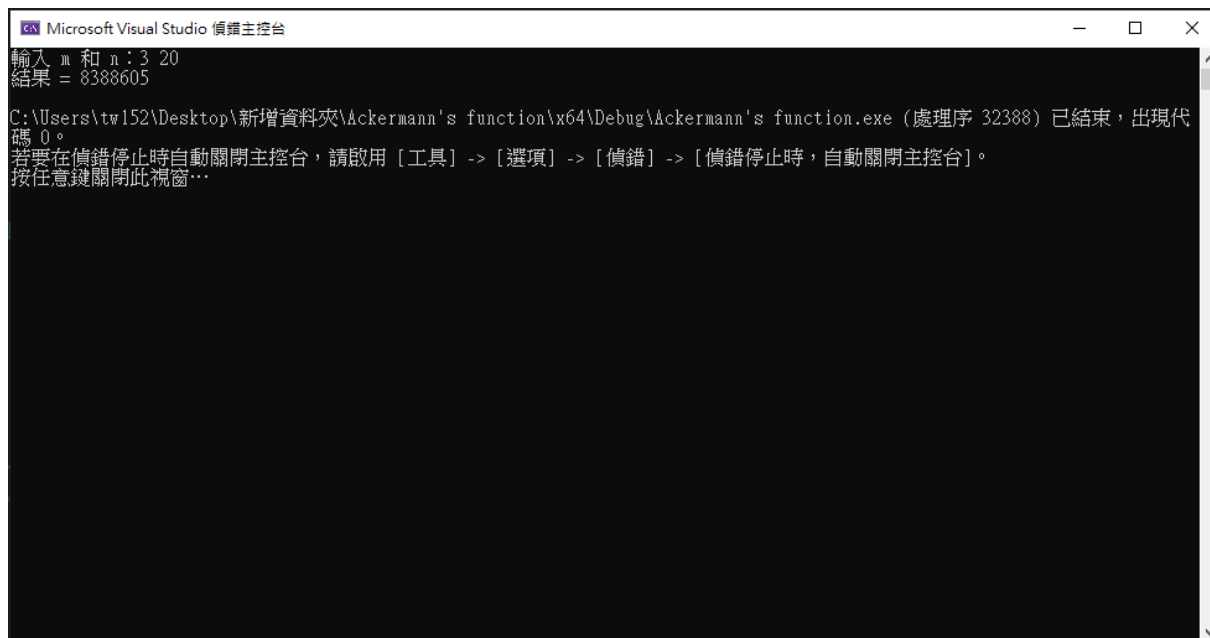
$$S(P) = 1$$

根據堆疊的大小,但我這個程式堆疊空間是固定的。

3.測試與驗證

結果的大小會因我定義堆疊的大小改變

```
unsigned long ackermann(int m, int n) {  
    sf stack[1000]; // 定義堆疊的大小  
    int top = -1; // 堆疊的指標  
    unsigned long result = 0;
```



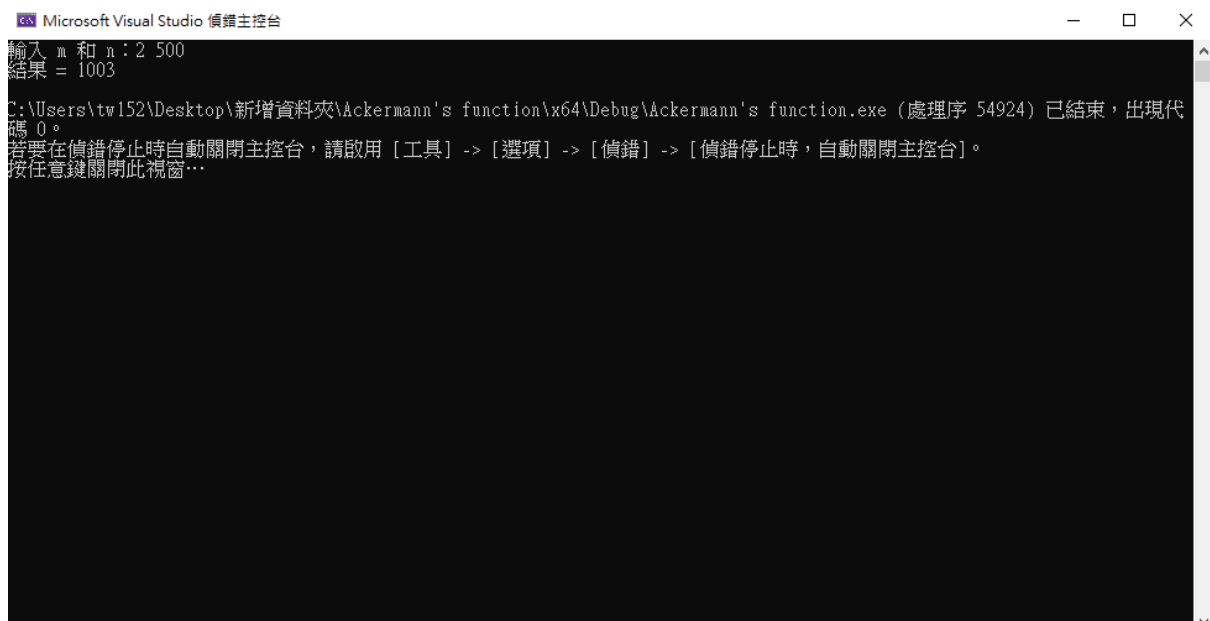
Microsoft Visual Studio 偵錯主控台

輸入 m 和 n: 3 20
結果 = 8388605

C:\Users\tw152\Desktop\新增資料夾\Ackermann's function\x64\Debug\Ackermann's function.exe (處理序 32388) 已結束，出現代碼 0。

若要在偵錯停止時自動關閉主控台，請啟用 [工具] -> [選項] -> [偵錯] -> [偵錯停止時，自動關閉主控台]。

按任意鍵關閉此視窗...



Microsoft Visual Studio 偵錯主控台

輸入 m 和 n: 2 500
結果 = 1003

C:\Users\tw152\Desktop\新增資料夾\Ackermann's function\x64\Debug\Ackermann's function.exe (處理序 54924) 已結束，出現代碼 0。

若要在偵錯停止時自動關閉主控台，請啟用 [工具] -> [選項] -> [偵錯] -> [偵錯停止時，自動關閉主控台]。

按任意鍵關閉此視窗...

```
if (m == 0)
{
    result = n + 1;
}
else if (m == 1)
{
    result = n + 2;
}
else if (m == 2)
{
    result = 2 * n + 3;
}
```

計算 $m \leq 3$ 時還是可行,可以用固定的公式算出

但只要 $m > 3$ 時,它就會開始變複雜, $A(m-1, A(m, n-1))$ 這東西就得將它前後分開存入 stack 中再依序解開它的結果,再往回推倒,而當 m 太大時會導致不管我 stack 會變的大到不行,或是說幾乎不可能設定的數字。

4. 申論與心得

這東西不用遞迴寫真的會想到腦袋炸掉,在 m 還小於 3 時還不需用到遞迴,但當它大於 3 時 $((3,0), (4,0), (5,0))$ 在非遞迴函式中還好,就會因為 $m > 0$ 和 $n > 0$

時, $A(m-1, A(m, n-1))$ 中 n 的部分會讓我程式中的 stack 暴增,所以當 m 是 4 或 5 時,這個程式依舊還是撐不下去,還有這東西真的複雜到哭,做到頭昏腦脹。