上機習題 #5 (recursion之應用)

題目：進行Ackerman function 之測試

說明：Ackerman function定義如下：(可參考課本習題3.1.7)

# A(m,n)=n+1 if m=0

A(m,n)=A(m-1,1) if m≠0，n=0

A(m,n)=A(m-1,A(m,n-1)) if m≠0，n≠0

Ackerman function 是一個成長非常快速的函數，其計算時，會有大量的recursive call。執行此function會使用大量的stack。m及n之值過大時，stack會不敷使用(會爆掉)，因此程式會當掉！請撰寫一個程式以計算Ackerman function之值，並找出在電腦上執行時所能容忍的m及n之最大值。即m=2、m=3、m=4、…、m=10時，n最大各可達多少。

可設定不同大小之stack來測試此程式 (方法參見下頁說明)。

本題除了繳交一般的上機習題檔案外，另需繳交一份簡易報告(Word檔案)，內容包含下列：

* 電腦的執行環境，包括使用的電腦硬體系統（CPU速度、主記憶體容量）、作業系統、編譯器名稱及版本。
* 設定stack的方式
* 你所設定的stack 大小，至少要有三種：  
  一個使用預設 Stack 大小 (不做任何設定)，  
  設定兩種不同 Stack 大小 (兩種大小設定差距要夠大，如 1000 倍以上)。
* 程式執行結果，即m=2、m=3、m=4、…、m=10時，n最大各可達多少。

撰寫C程式時，stack大小之設定方法如下：

在BCB中

Project → Options → Linker → PE file options

當中可看到

Min stack size : 0x00002000 (8192)

Max stack size : 0x00100000 (1048576)

0x代表為16進位

8192 ≦ Min\_Stack\_Size ≦ Max\_Stack\_Size ≦ 1048576

在 VC 中

Project → Setting → Link

將 Category 選項改成 Output

可以看到有一個 Stack Allocations 當中

Reserve → virtual memory

Commit → physical memory

Default 的 stack size 是 1MB

最大值為 0xffffffff (4,294,967,295)

或是可以在 compiler 完後..

在 command mode 下

下參數 /Fnumber , number 即 set stack size ..

number可以用十進位表示, 單位是 bytes

在 Dev-C++ 中

Tools → Compiler Options → Compiler

勾選 Add these commands to the linker command line

在這裡下這樣的參數

-Wl,-stack,{stack size, kb}

ex: -Wl,-stack,50000

在 Code::Blocks 中

Settings → Compiler and debugger settings → Compiler Settings → Other options

在這裡下這樣的參數

-Wl,-stack,{stack size, kb}

ex: -Wl,-stack,50000

其他環境請善用網路搜尋。