

# Lab Homework Week 14 Report

Group 66: 102403015 程祥恩、102403016 邱威穎、102403020 曾子軒

**目標：**請利用本週課程所學內容完成程式。在終端機中畫出指定大小和指定顏色的框線。

**程式碼：**

```
exercise14.asm - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)
TITLE exercise14.asm[exercise14.asm]
INCLUDE Irvine32.inc

BoxWidth = 7
BoxHeight = 7

main EQU start@0

.data
boxTop    BYTE 0DAh, (BoxWidth - 2) DUP(0C4h), 0BFh
boxBody   BYTE 0B3h, (BoxWidth - 2) DUP(' '), 0B3h
boxBottom BYTE 0C0h, (BoxWidth - 2) DUP(0C4h), 0D9h
outputHandle DWORD 0
bytesWritten DWORD 0
count     DWORD 0
xyPosition COORD <10,5>
cellsWritten DWORD ?
attributes0 WORD BoxWidth DUP(0Bh)
attributes1 WORD (BoxWidth-1) DUP(0Ah),0Eh
attributes2 WORD BoxWidth DUP(0Ch)

.code
main PROC
    INVOKE GetStdHandle, STD_OUTPUT_HANDLE    ; Get the console output handle
    mov outputHandle, eax                     ; save console handle
    call Clrscr
    ; 畫出box的第一行

    INVOKE WriteConsoleOutputAttribute,
        outputHandle,
        ADDR attributes0,
        SIZEOF boxTop,
        xyPosition,
        ADDR count

    INVOKE WriteConsoleOutputCharacter,
        outputHandle,    ; console output handle
        ADDR boxTop,     ; pointer to the top box line
        SIZEOF boxTop,   ; size of box line
        xyPosition,      ; coordinates of first char
        ADDR count        ; output count

    inc xyPosition.Y      ; 座標換到下一行位置
    mov ecx, (BoxHeight) ; number of lines in body
```

```

L1: push ecx    ; save counter 避免invoke 有使用到這個暫存器
    INVOKE WriteConsoleOutputAttribute,
        outputHandle,
        ADDR attributes1,
        SIZEOF boxBody,
        xyPosition,
        ADDR count

    INVOKE WriteConsoleOutputCharacter,
        outputHandle,
        ADDR boxBody,    ; pointer to the box body
        SIZEOF boxBody,
        xyPosition,
        ADDR count ;

    inc xyPosition.Y    ; next line
    pop ecx             ; restore counter
    loop L1

    INVOKE WriteConsoleOutputAttribute,
        outputHandle,
        ADDR attributes2,
        SIZEOF boxBottom,
        xyPosition,
        ADDR count

    ; draw bottom of the box
    INVOKE WriteConsoleOutputCharacter,
        outputHandle,
        ADDR boxBottom,    ; pointer to the bottom of the box
        SIZEOF boxBottom,
        xyPosition,
        ADDR count ;

    call WaitMsg
    call Clrscr
    exit
main ENDP
END main

```

## 程式說明：

程式預先設定好所有相關資料，包括總長寬、圖形上、中、下部分的符號、顏色與起始座標，然後接著在主程式中依序印出上、中、下的部分圖形形成一彩色邊方形。首先 `GetStdHandle` 取得標準輸出物件並放入變數 `outHandle` 中，接著 `WriteConsoleOutputAttribute` 是設定特定位置上特定字元數的顏色，`WriteConsoleOutputCharacter` 是設定特定位置上特定字元數

的內容，因此呼叫這兩個方法參數的差異就在於一個是要印的內容一個是設定的顏色，就可以在特定位置印出特定顏色的字元了，印出上部分的蓋子後將位置 `xyPosition.Y` 加一代表換行，中間部分使用了迴圈印了 7 次，下部分也與上部分相同。

### 程式結果：



調整了編碼與字形後便可以看到正確的圖形。

### 心得

最近的課程都在都在教導我們如何透過 Microsoft 的 API 把資料顯示在命令提示字元上面，而且竟然還可以變換顏色，讓我顛覆了以前對組合語言的想像(原本以為組合語言只能拿來控制暫存器，和顯示一些簡單的文字)，不過指令有點多且複雜，還需要一些時間消化，也許最後的期末報告時可以派上用場吧！