國 立 台 灣 科 技 大 學



微 算 機 概 論 實 習

指 導 教 授：陸敬互 教 授

**------------------------------------------------------------------------**

微算機概論實習報告

（原創）

期末報告

班 級 ： 四電機二乙

學 生 ： 陳○○、李○○

學 號 ： B110300XX、B110300XX

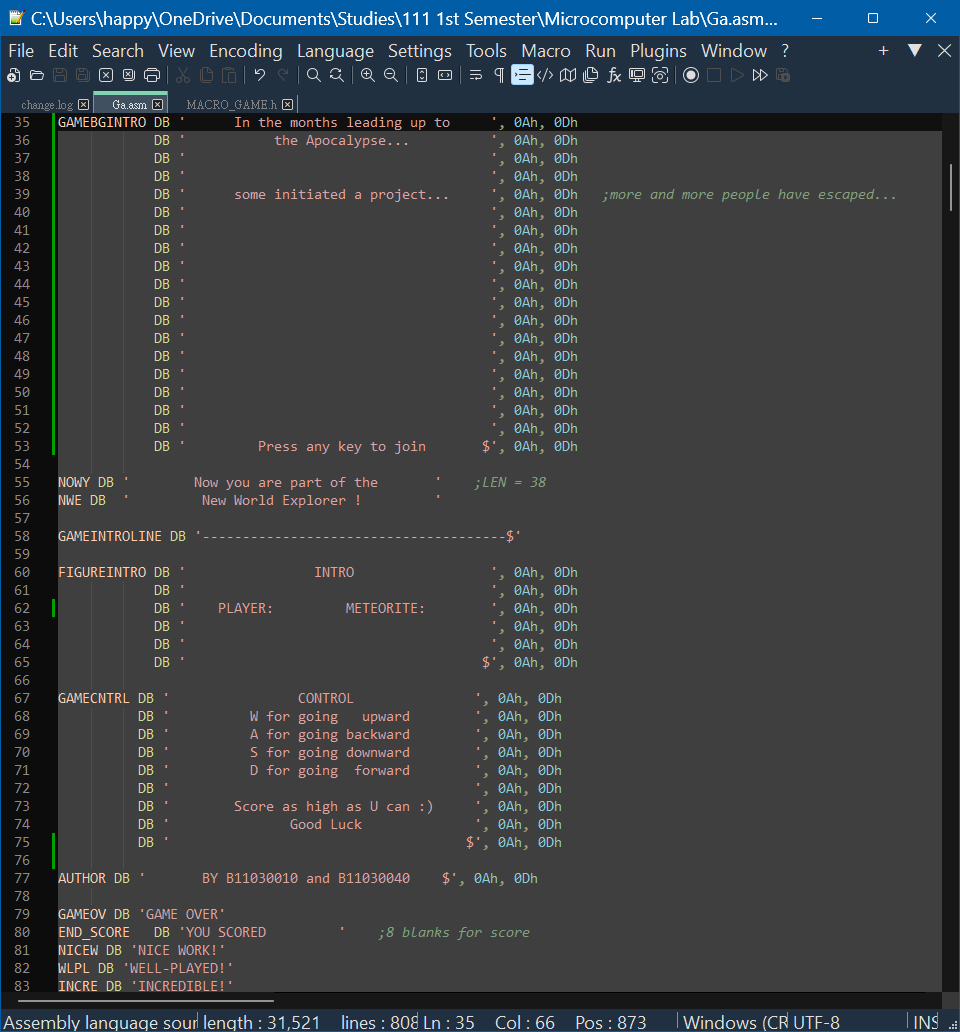
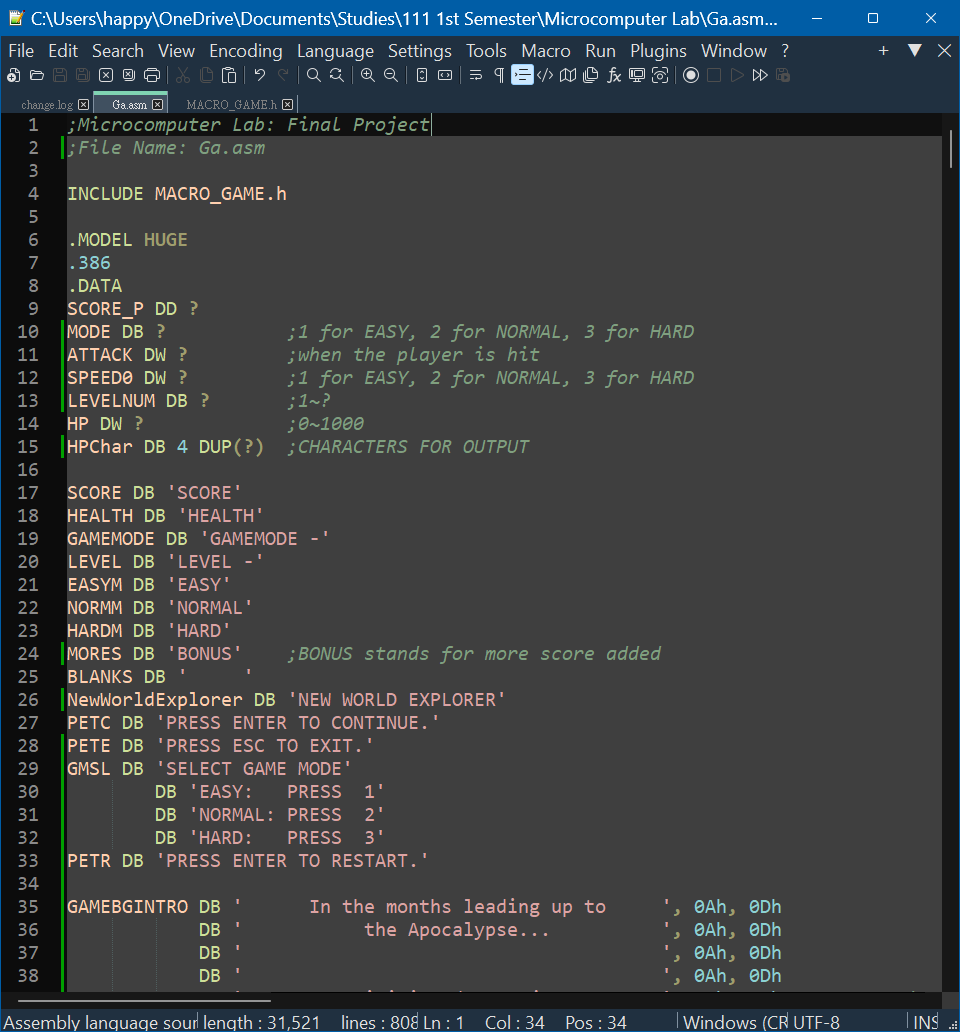
建檔日期： 2022/12/14

(中文字型:標楷體 英文字型:Times New Roman)

1. 學習成果(程式功能說明，說明程式分幾個部分，各部分在做甚麼，搭配截圖code說明)

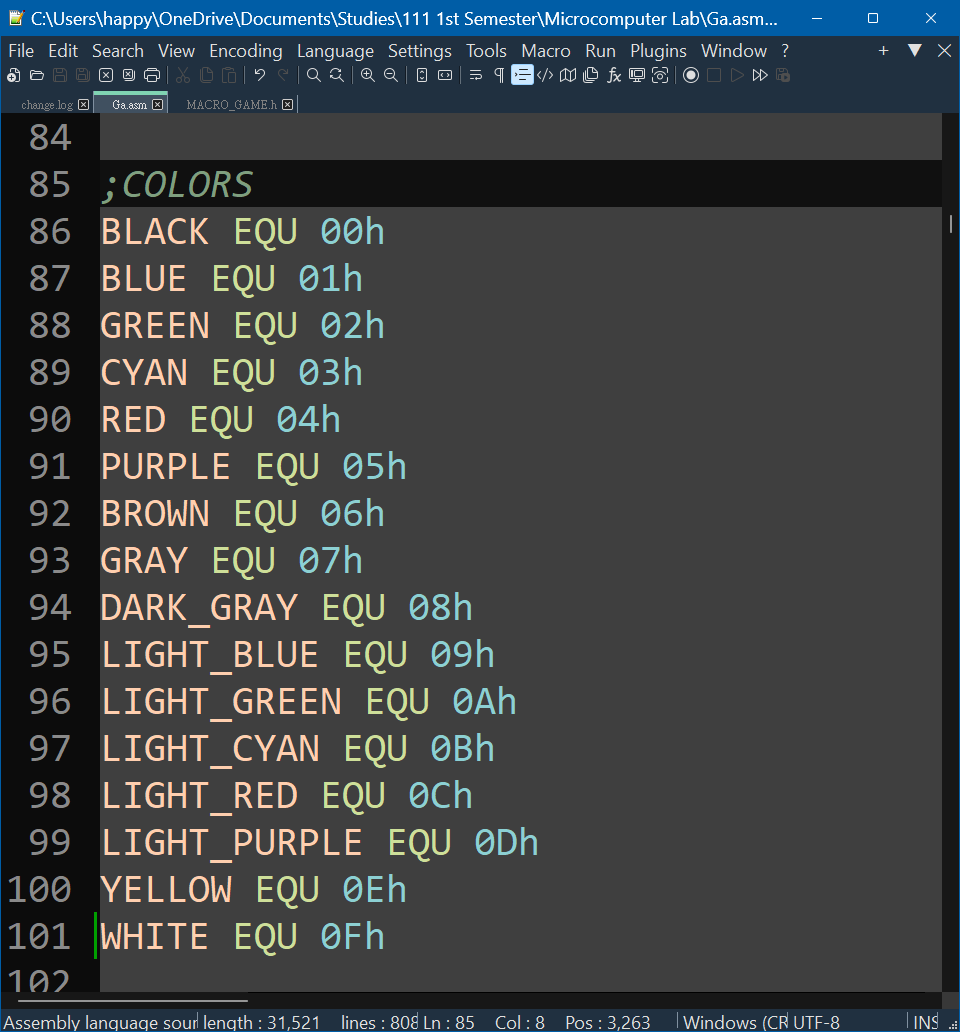
**遊戲名稱：新世界探索者（New World Explorer）**

* + 變數
    1. 介面與字串



此區大多為用於介面的字串，其中包括了一些功能性的變數，因為分數採用double word的資料型態，因此加入了.386

* + 1. 圖案
       1. 顏色



我們使用EQU賦予十六色各一個名稱（標籤），以提升可用性與易讀性

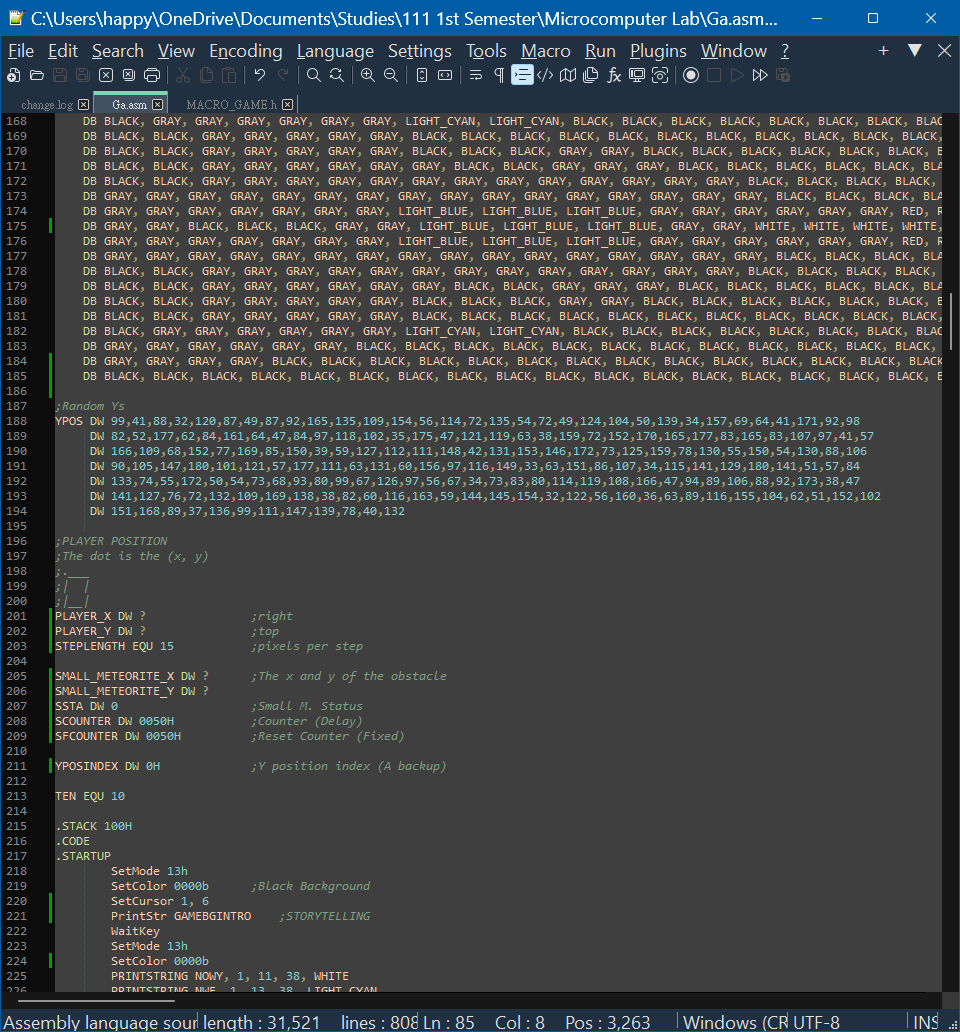
* + - 1. 小隕石、大隕石與玩家



這部分使用人工，將一個個色塊放進一陣列中，在遊戲中則以迴圈方式將圖案輸出。小隕石與玩家大小為20\*20 pixels2，大隕石則為40\*40pixels2

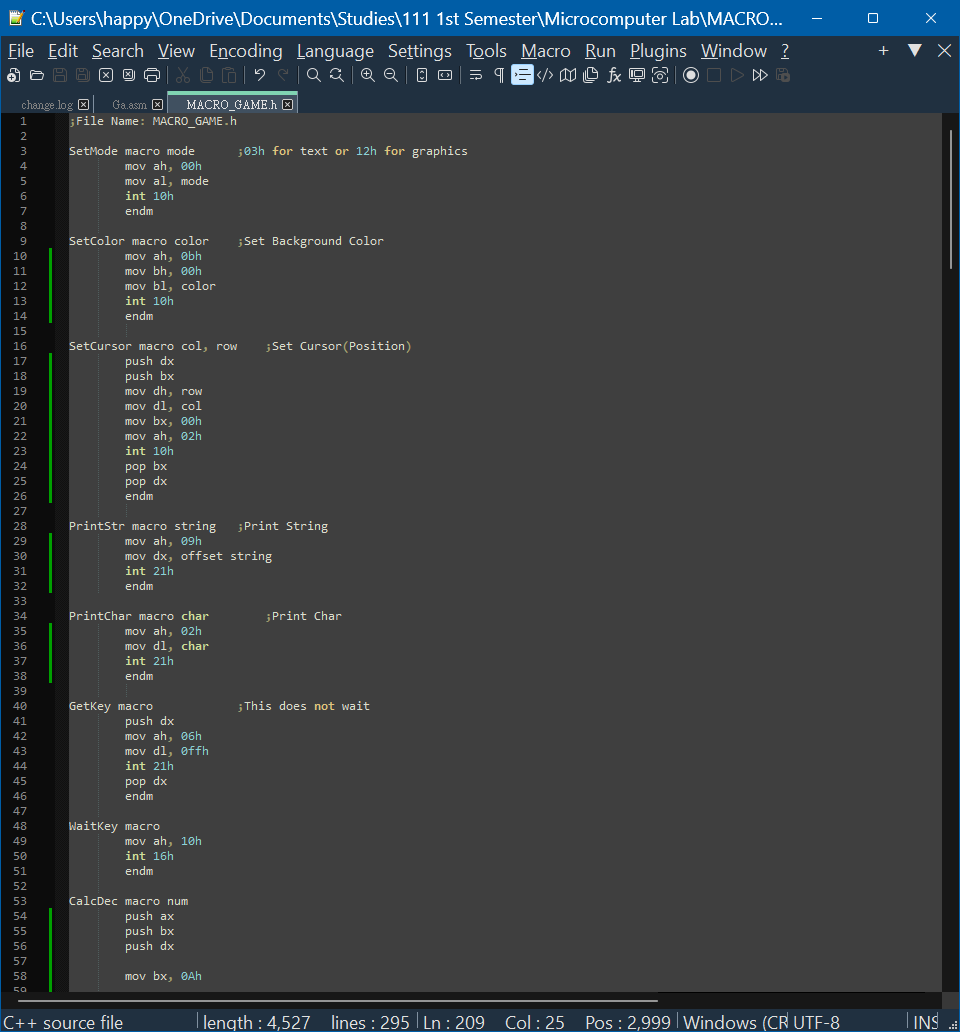
* + 1. 參數

YPOS為一隨機數字之陣列，使障礙物能夠隨機出現，共200個，YPOSINDEX為YPOS的索引。玩家與障礙物都各有一組位置(x,y)，代表其圖案之左上方的那個像素，而玩家每一步的移動（STEPLENGTH）為15個像素，COUNTER與STA分別用於延遲與表示障礙物的存在與否，詳細內容在巨集與副程式的介紹中。



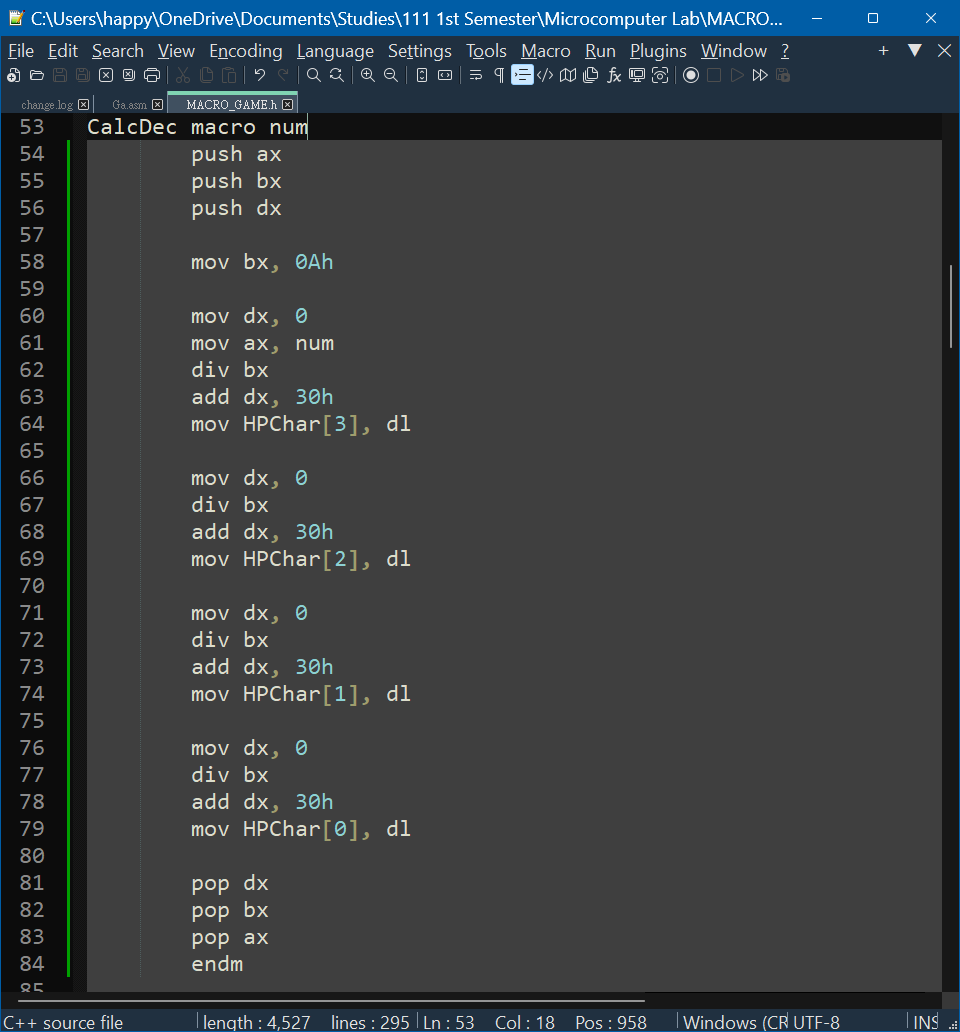
* + 巨集：

1. 簡單的功能



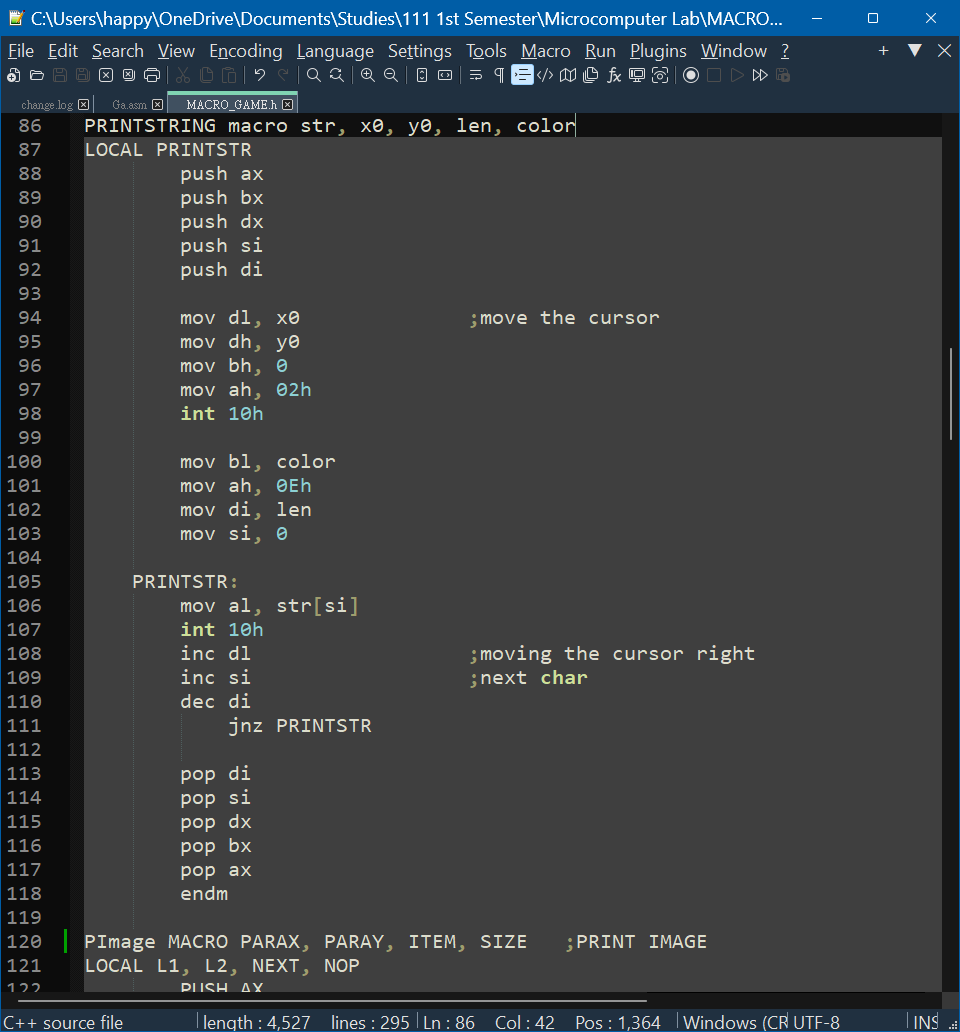
這些為基本的指令，在這邊不贅述

1. CalcDec 十六進制轉十進制



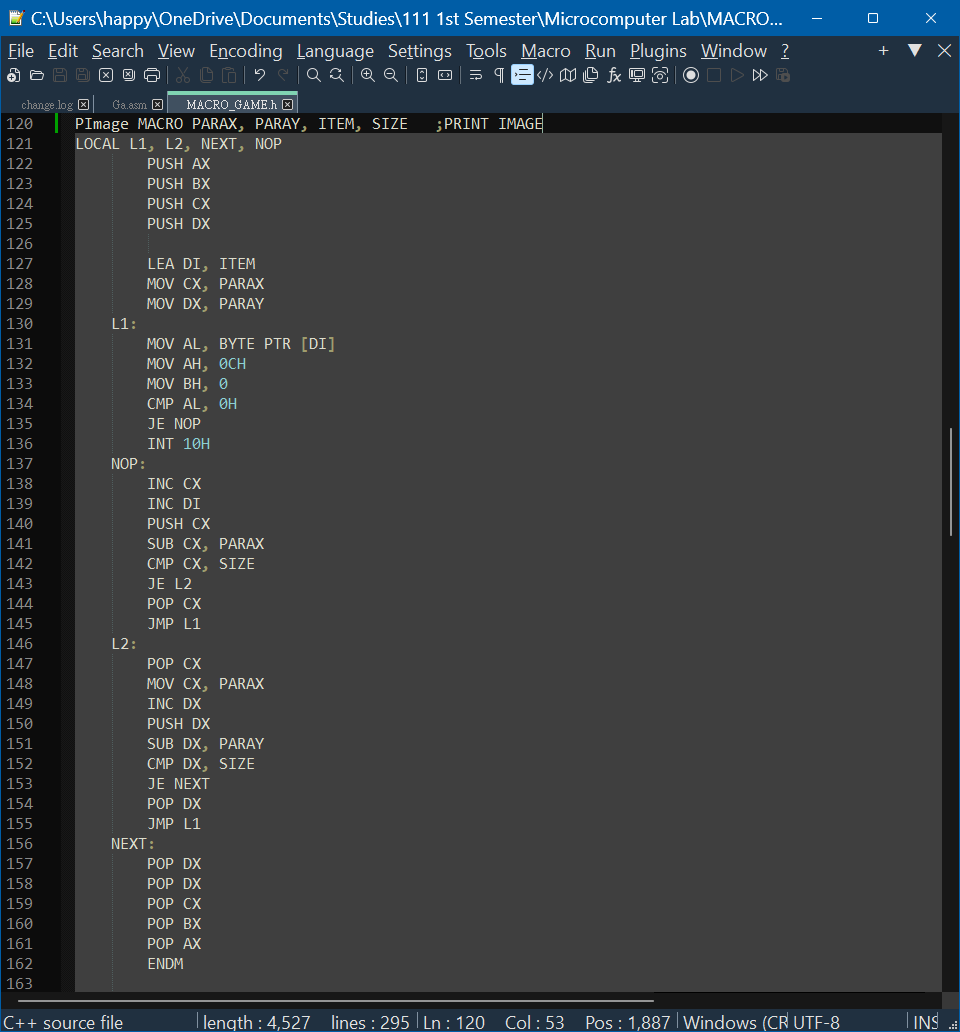
這是將資料從十六進制轉為十進制的巨集，在遊戲中僅使用在轉換生命值

1. PRINTSTRING 印出有顏色的字串



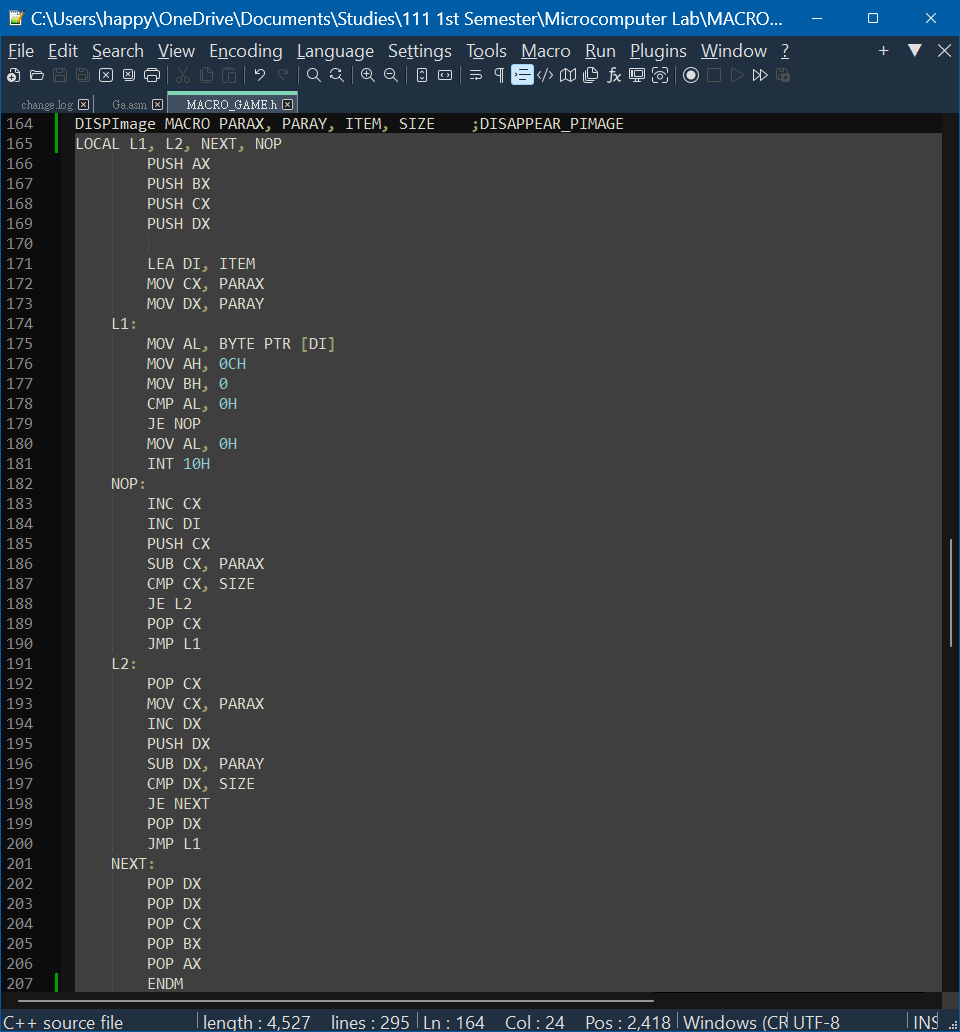
這邊使用到了INT 10H的0EH功能，能輸入有顏色的字串

1. PImage 印出圖案



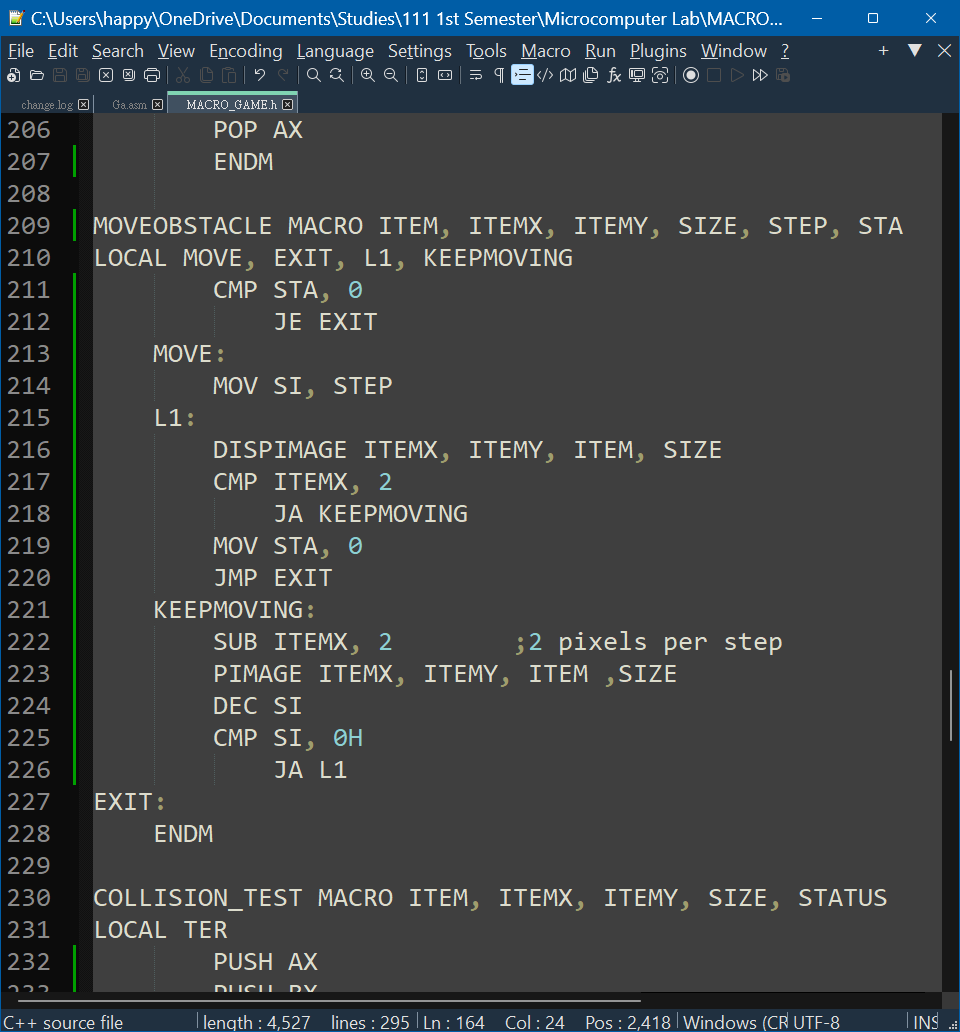
此巨集將圖案印出，印出前會檢查欲印之色塊，若為黑色（玩家或障礙物的邊角，背景色的地方）則不執行中斷，以此印出乾淨的圖形

1. DISPImage 印出黑色以清除圖案



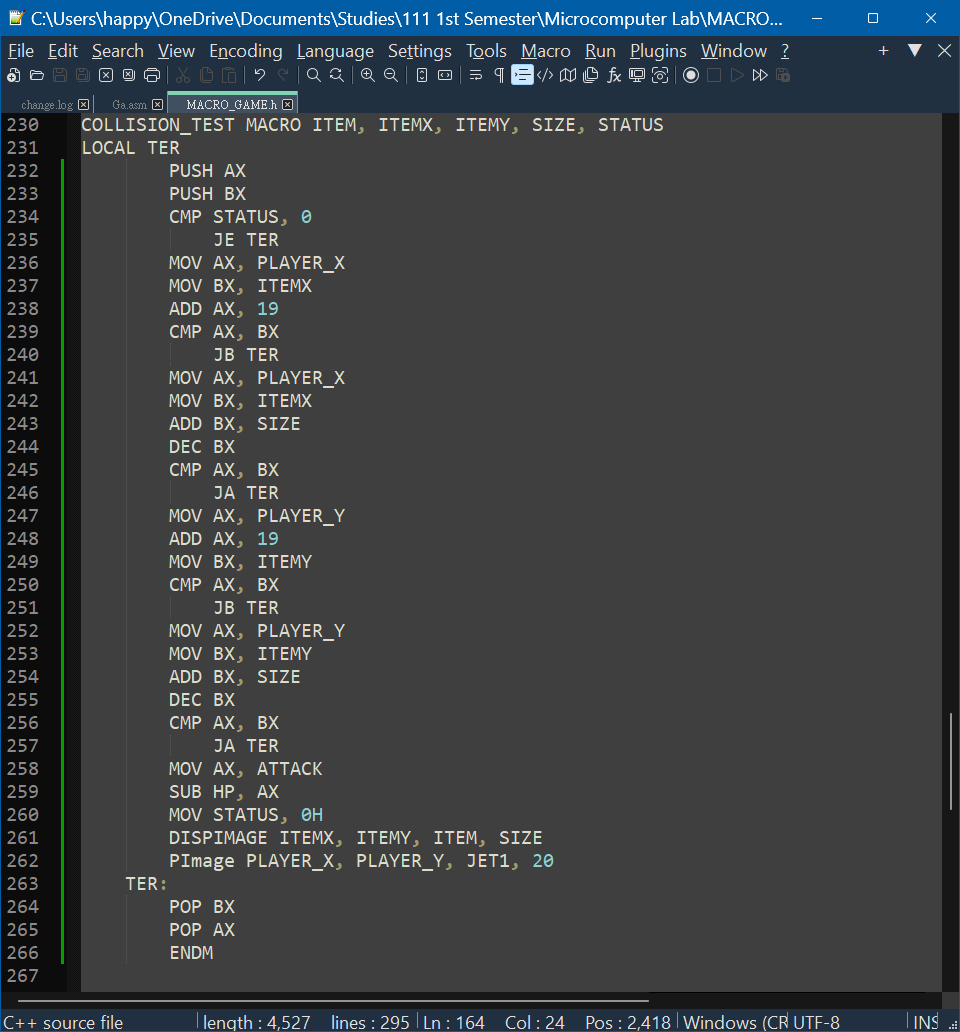
在圖案移動前，須將原圖案清除，原本我們以20\*20 pixels2的黑色方格用於清除，進行改良後，就多了這個巨集，同時也微幅提升了程式的效率。此巨集與上一個印出圖案的巨集相似，差別僅在於印出的色塊只會是黑色，印在非黑色的位置，以蓋掉原先的圖案。

1. MOVEOBSTACLE 障礙物移動



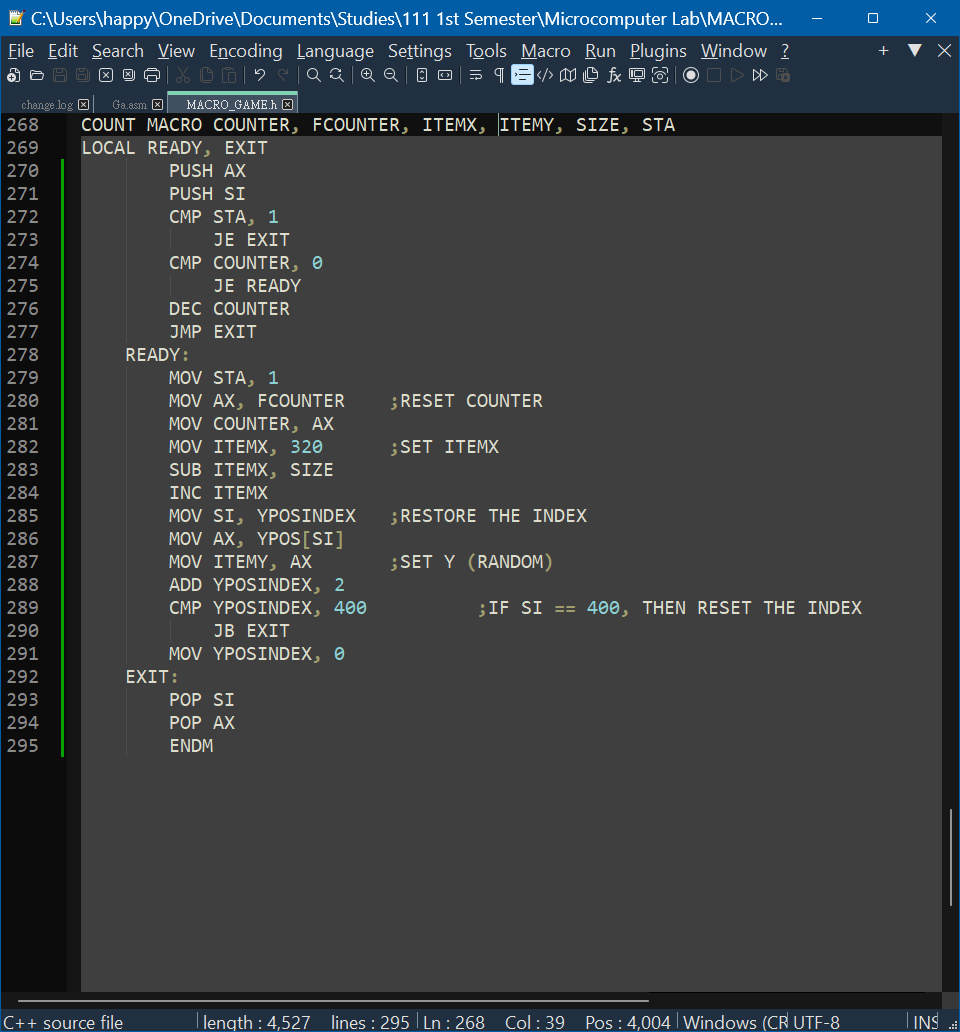
我們設置障礙物移動每步為2個像素，依照STEP輸入去做出移動速度。當障礙物不存在時，直接跳出巨集，不執行任何指令。當障礙物跑到畫面邊界時，就將障礙物之狀態設置為0並跳出巨集，因為在清除圖案後沒有再印上移動過後的的圖案，所以障礙物到邊界時就會消失在畫面上。

1. COLLISION\_TEST 碰撞判定



碰撞的產生源自於圖案重疊，那麼在一個個判斷的篩選後，仍沒被篩掉（跳出巨集）的情況，就是碰撞產生的情況了。原本我們對玩家的三個點（形狀近似於三角形）進行是否碰撞的判定，實作完成後我們發覺有太多重複且多餘的部分，因此我們進行改良，僅留下原本1/3長的程式碼，提升了程式效率，在權衡之下也犧牲了一些精確度。

1. COUNT 障礙物出現之延遲



若在消失或碰撞後（障礙物不存在）又馬上出現新的障礙物，會讓節奏太緊張，因此我們加入了COUNT巨集，讓程式在障礙物不存在時開始倒數，等到COUNTER為零時才允許其出現，並賦予其初始位置、重設COUNTER，以達到延遲的效果。

**關於SIZE**

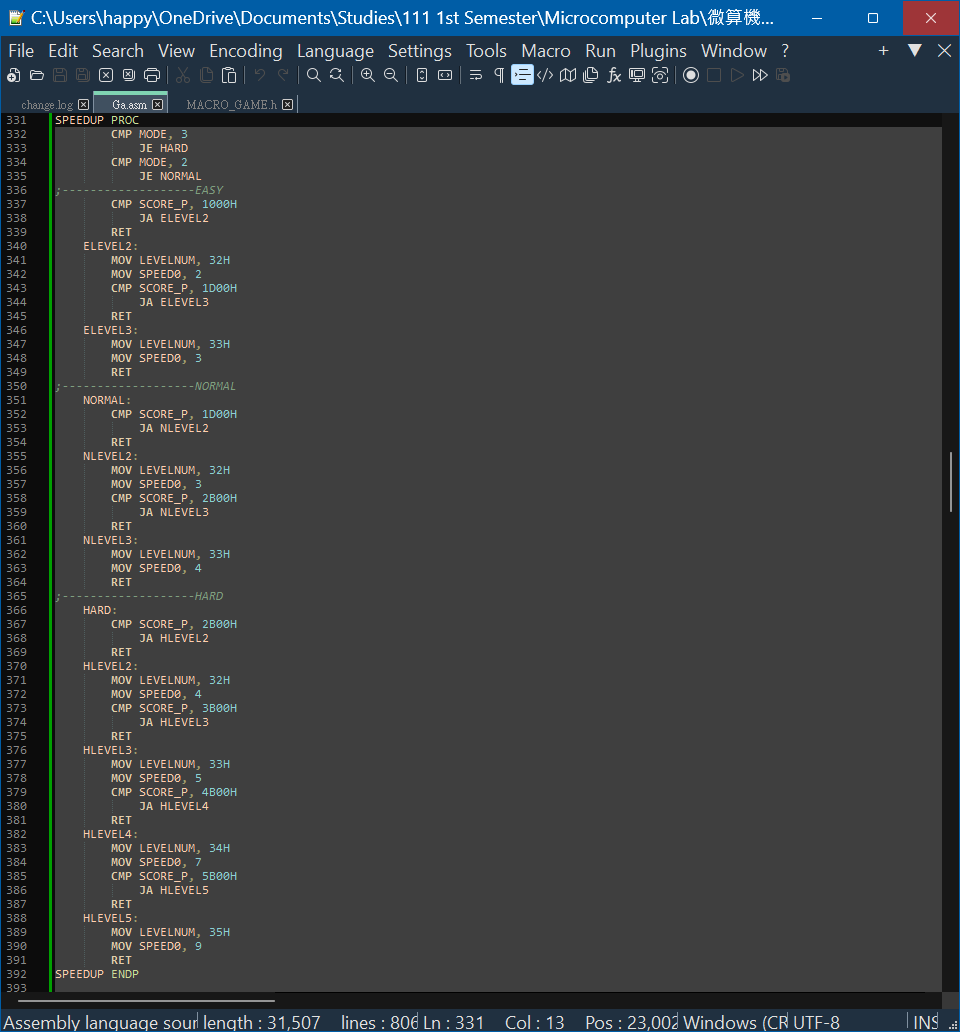
事實上，上面所看到的SIZE輸入本來是為了讓巨集能夠調整印出或判斷邊長為20或40的障礙物，以讓遊戲更豐富，但在測試時我們發現這在實現上有問題，因為印出四倍的色塊（1600 = 400\*4）實在太耗時間，導致移動速度受影響，變得很慢沒挑戰性，也讓球體看起來比較不完整（印很久），因此最終放棄了更大的障礙物。另外，本來我們想使多個障礙物出現於畫面上，不過也產生了同樣的問題，會有移動速度變慢、跑分變慢的問題，因此最終這兩項設計宣告失敗。

以下當時測試使用DOSBox執行遊戲的狀況

3000cycles是一般的執行速度，可以看到障礙物的移動上還頗尷尬的，也能感覺到沒有挑戰性。6000cycles則是兩倍的執行速度，很明顯的，被撞之後剩一顆的速度變得快許多，會打亂遊戲節奏，這些都是無法被克服的。

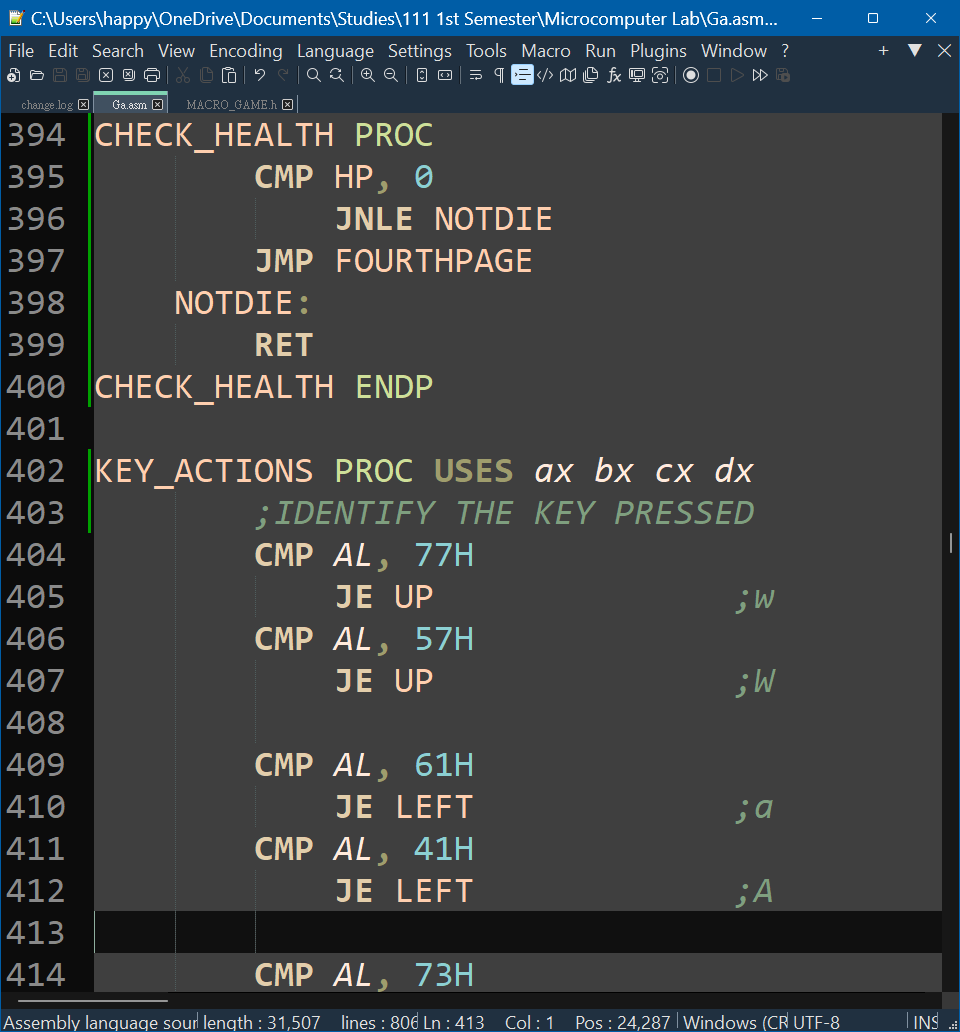
副程式：

* + 1. SPEEDUP 加速



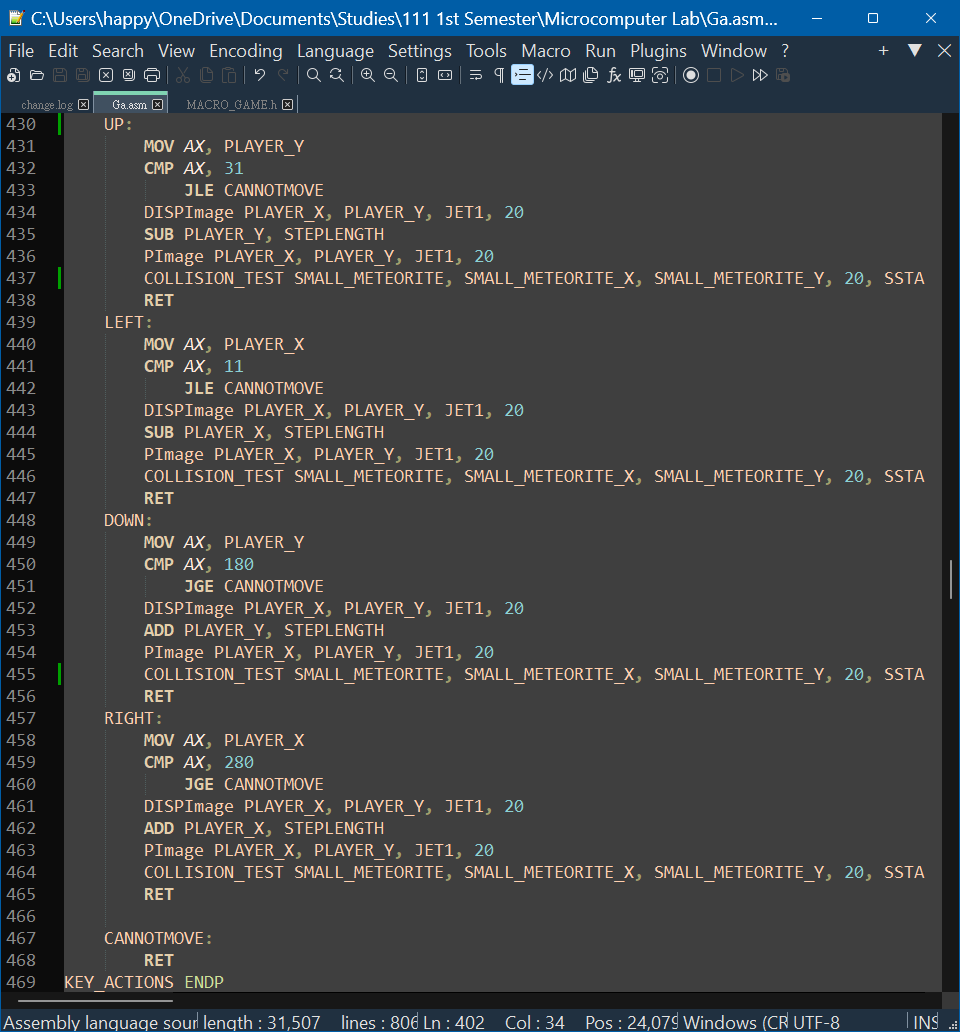
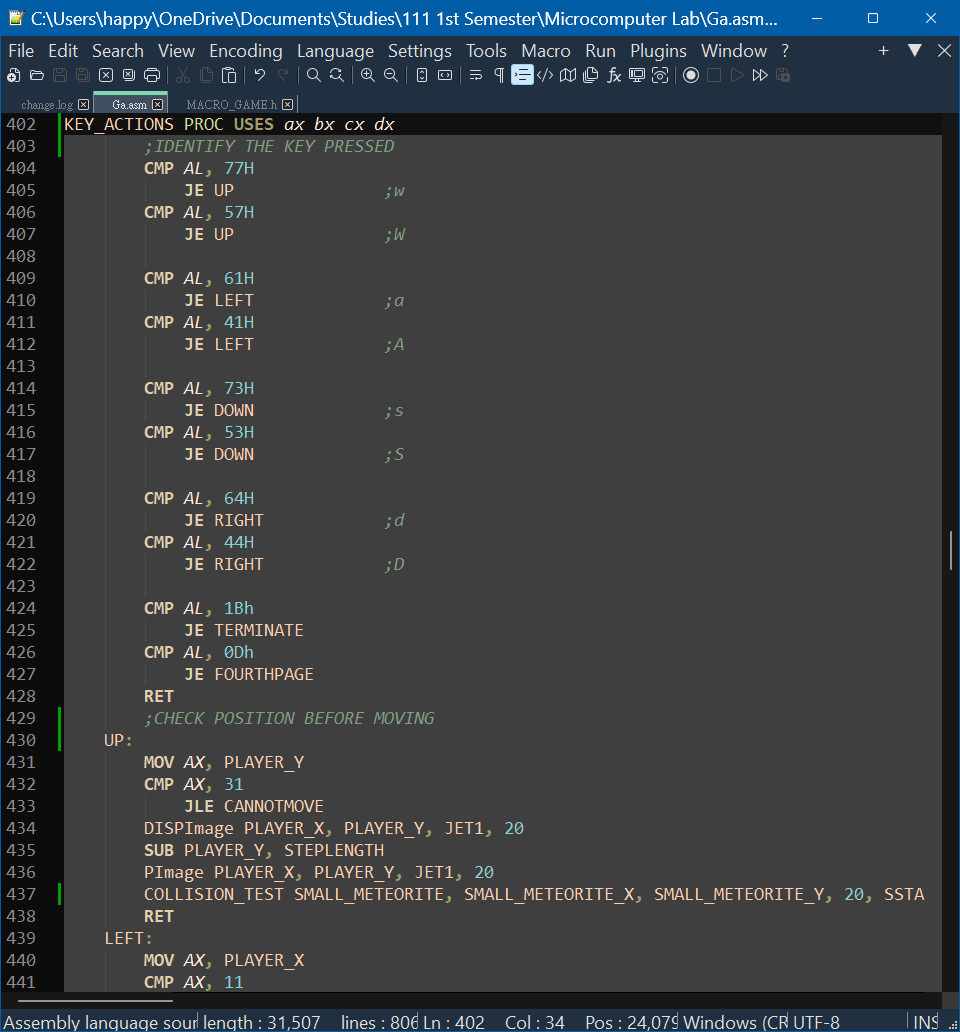
這個巨集會判斷玩家的積分以決定是否加速。簡單模式與普通模式的最高等級為3，最高速度分別為3與4，困難模式的最高等級為5，最高速度則高達9。積分界定的依據是測試與感覺，因此並非線性。

* + 1. CHECK\_HEALTH 確認生命值



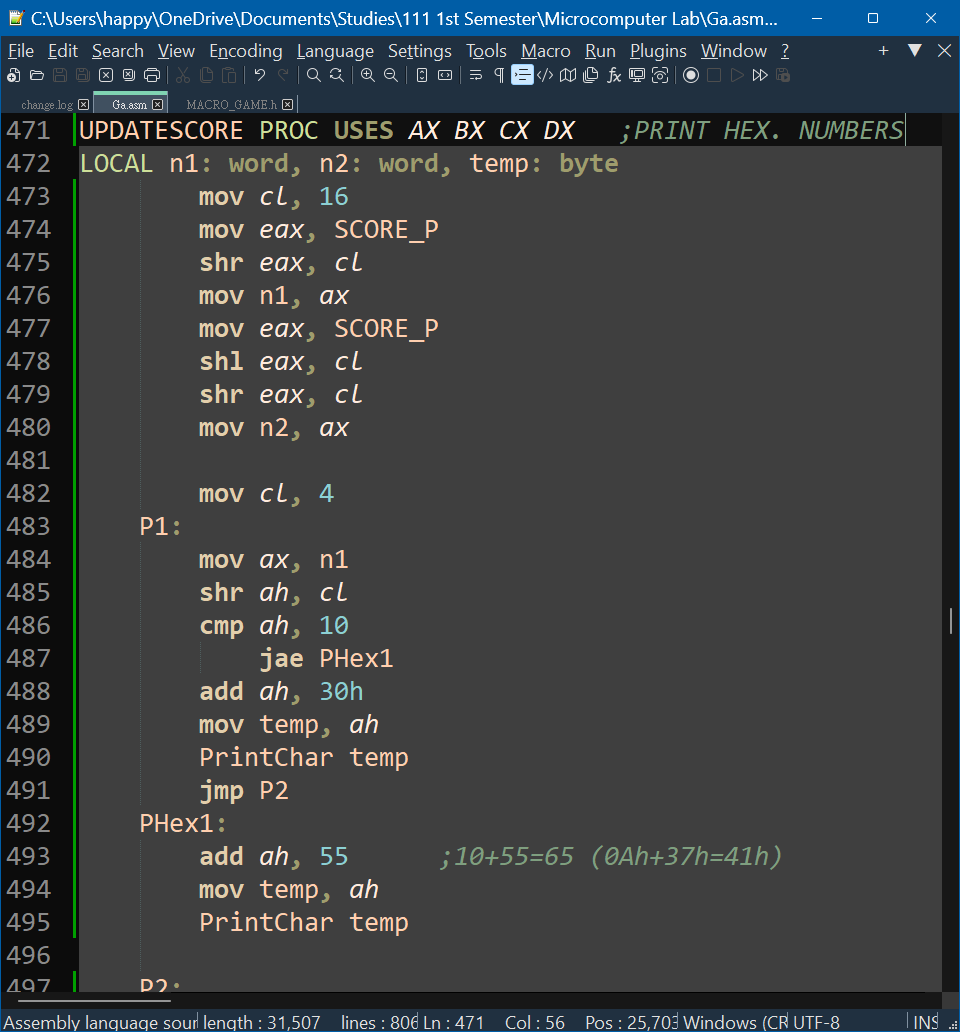
一簡單的判斷，若玩家生命值非小於等於0時，則繼續遊戲，否則進入第四頁（遊戲結束）。

* + 1. KEY\_ACTIONS 檢查按鍵



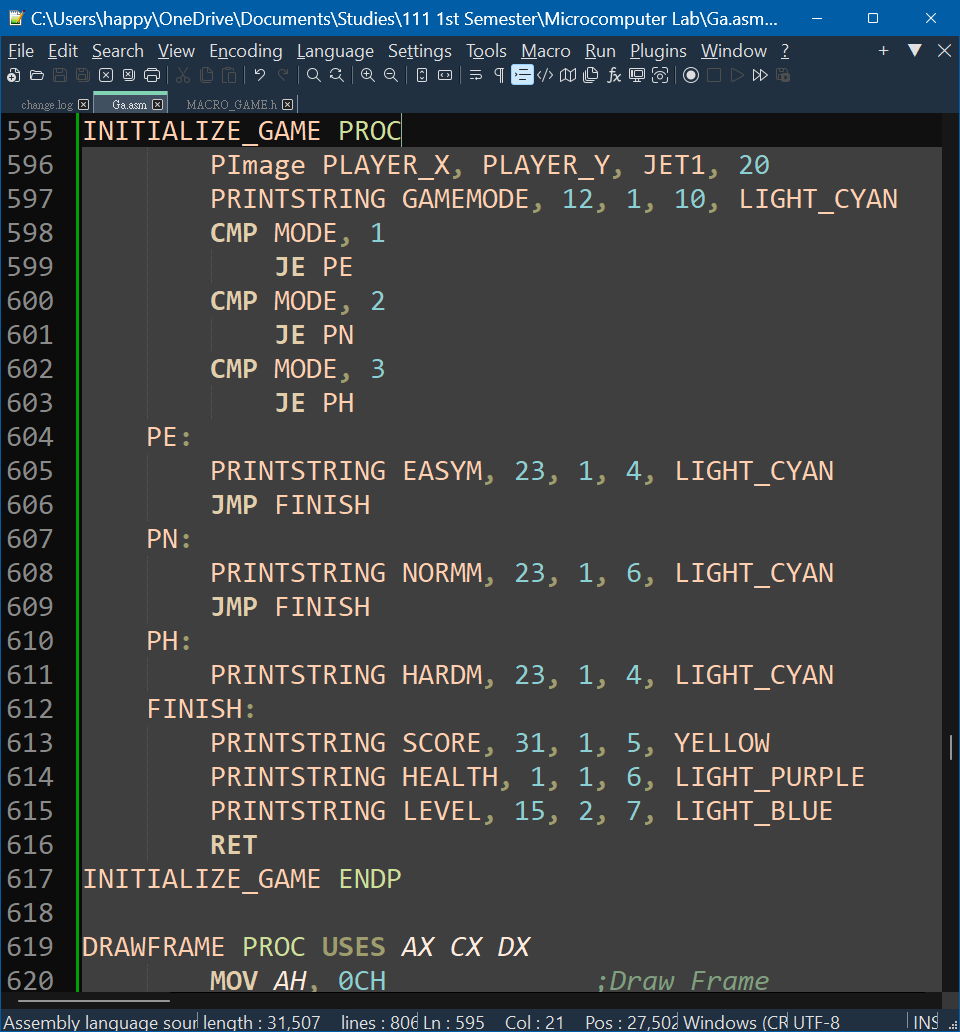
接收玩家輸入後，先辨別按鍵為何，再決定動作。移動前會確認玩家是否能繼續於遊戲畫面內移動而不超出範圍，若成功移動完則會再進行一次碰撞判定。

* + 1. UPDATESCORE 更新積分



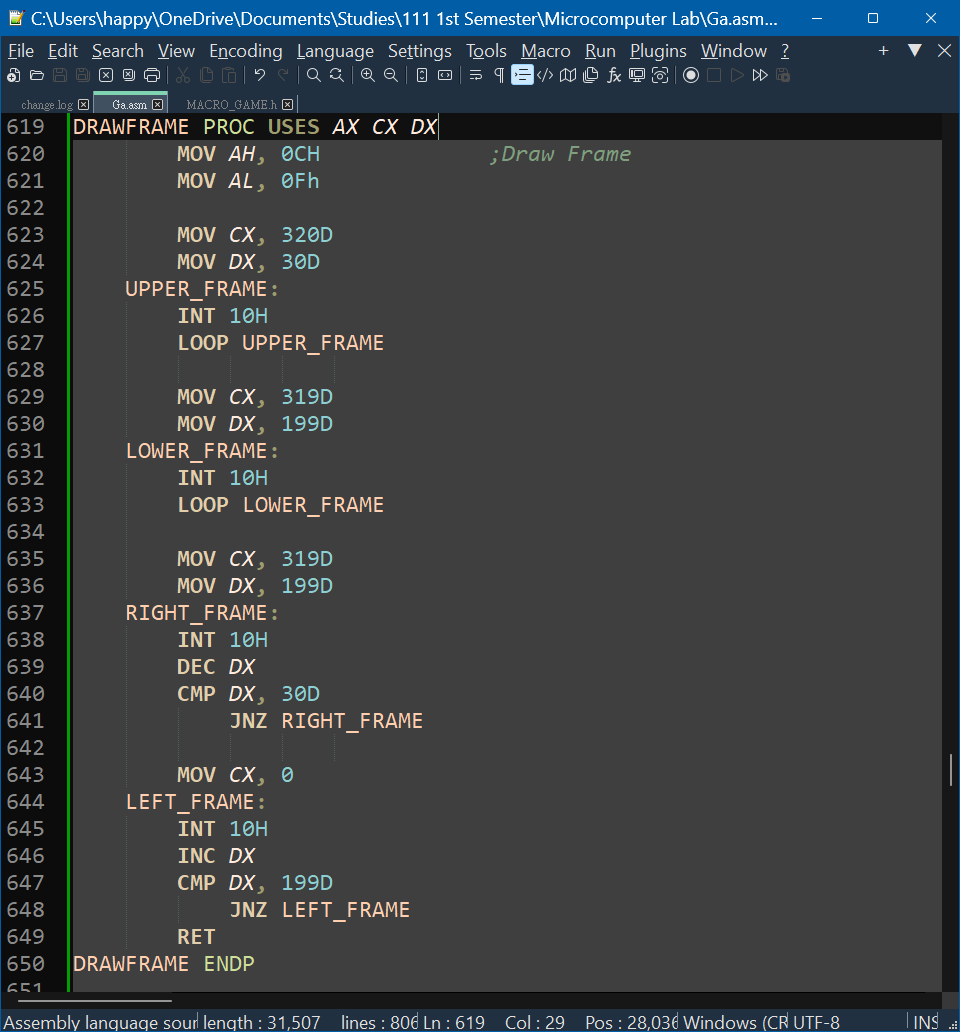
此副程式用於印出遊戲積分，十六進制數字共8位數，因程式碼重複性高，於此僅呈現一部分。

* + 1. INITIALIZE\_GAME 遊戲介面初始化



將遊戲畫面上方的狀態列印出

* + 1. DRAWFRAME 畫遊戲的白框



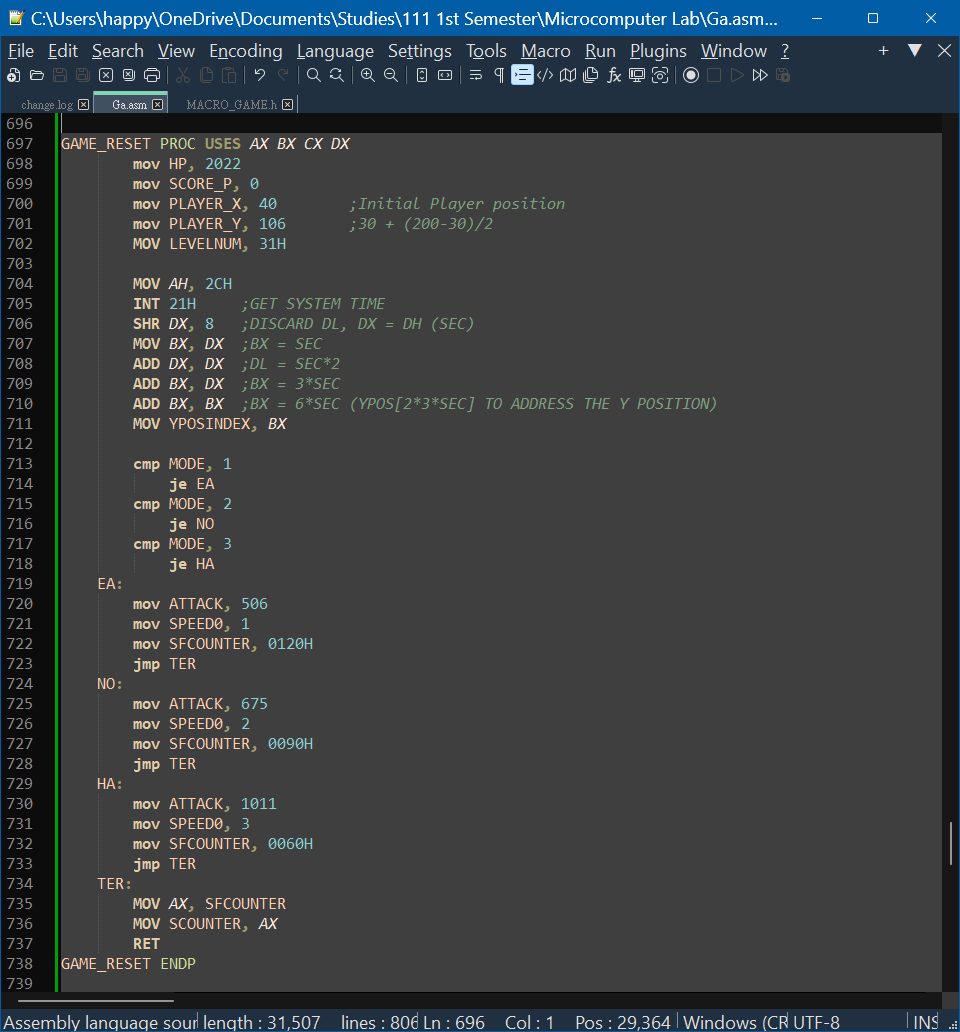
此白框將狀態列和玩家與障礙物的移動範圍區隔開

* + 1. ADDSCORE 加分



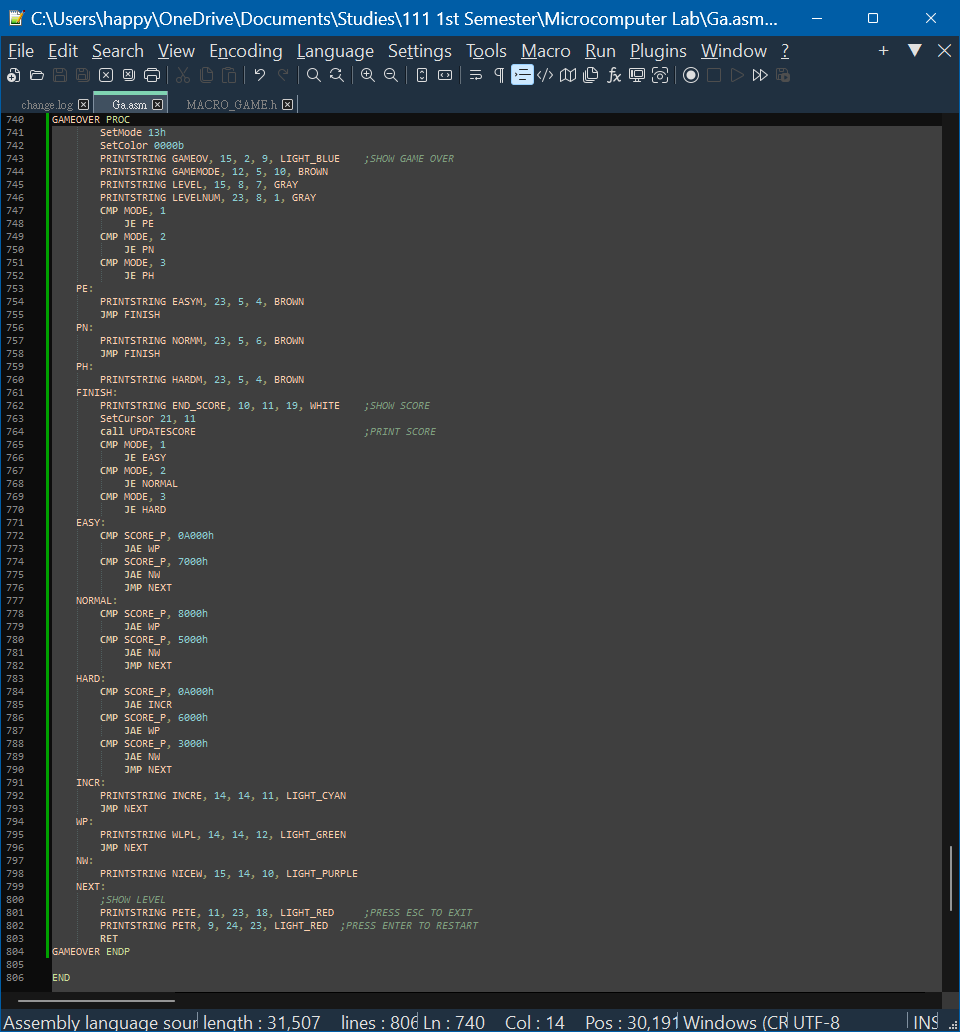
按照遊戲難度累加積分與判定玩家位置並給予獎勵分數

* + 1. GAME\_RESET 設定遊戲參數



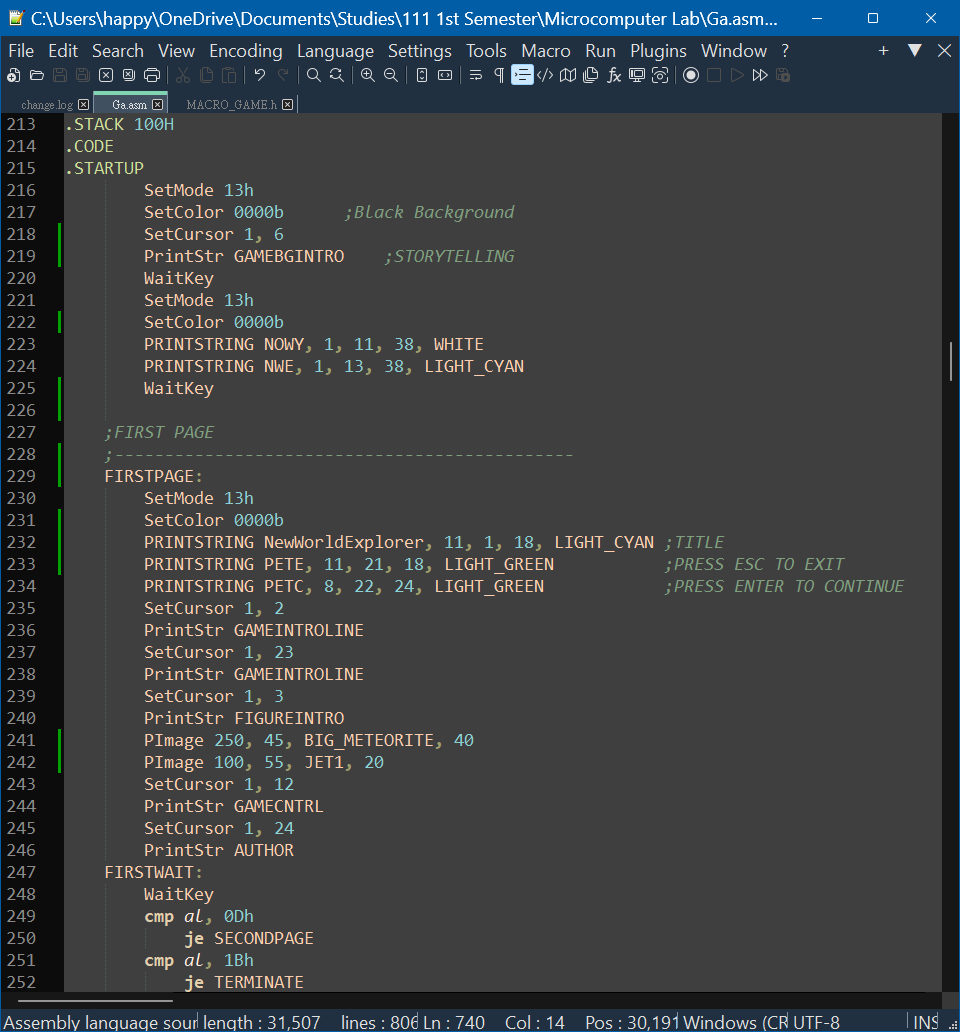
遊戲開始前將玩家位置、等級、分數、生命值等回歸初始設定，並調整遊戲難度。我們取系統時間（秒數），將其六倍（可為0~354中的一些數字）做為YPOSINDEX的值，即可使400bytes的YPOS中的某些數字作為障礙物的初始（遊戲開始後的第一個）Y值，以此便可做出隨機的效果。

* + 1. GAMEOVER 遊戲結束



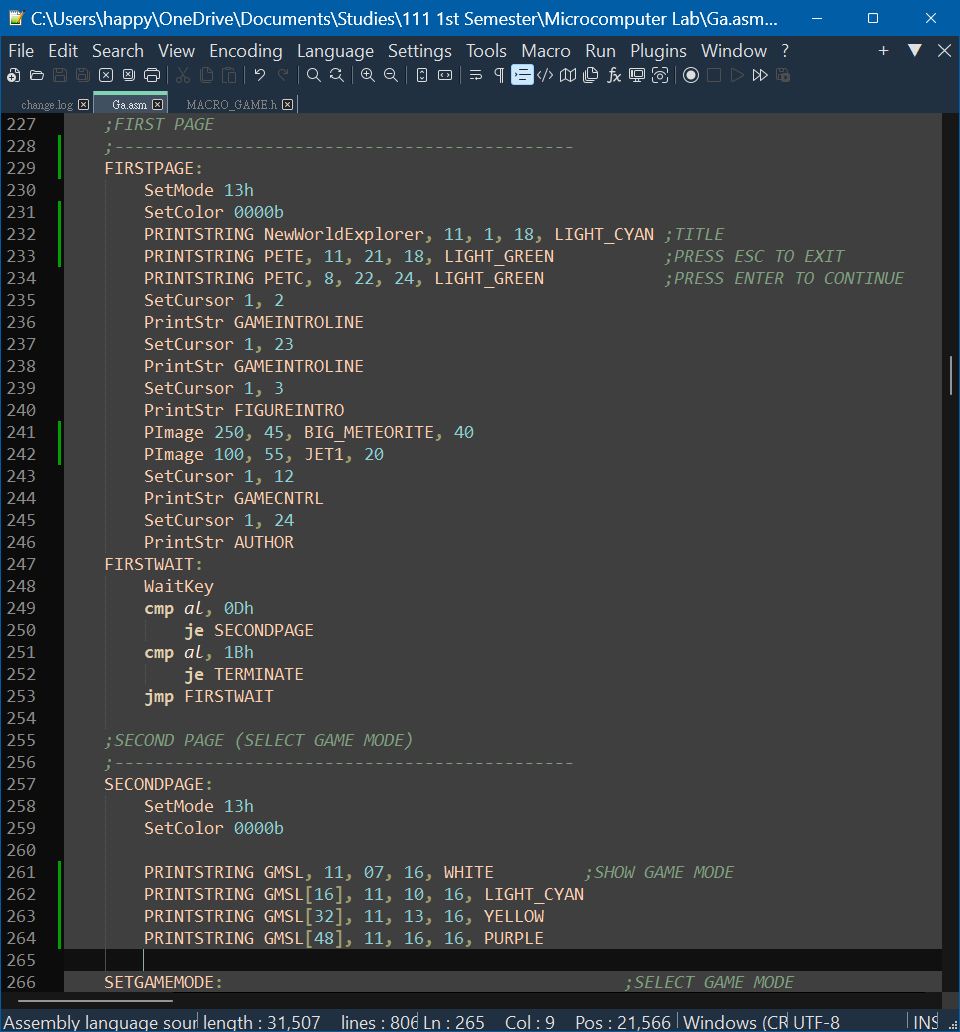
印出相關字串（如實習結果所示），並在每個遊戲難度下判斷該玩家的成績高低給予相對應的鼓勵話語

* + 主程式：
    1. 第零頁——故事



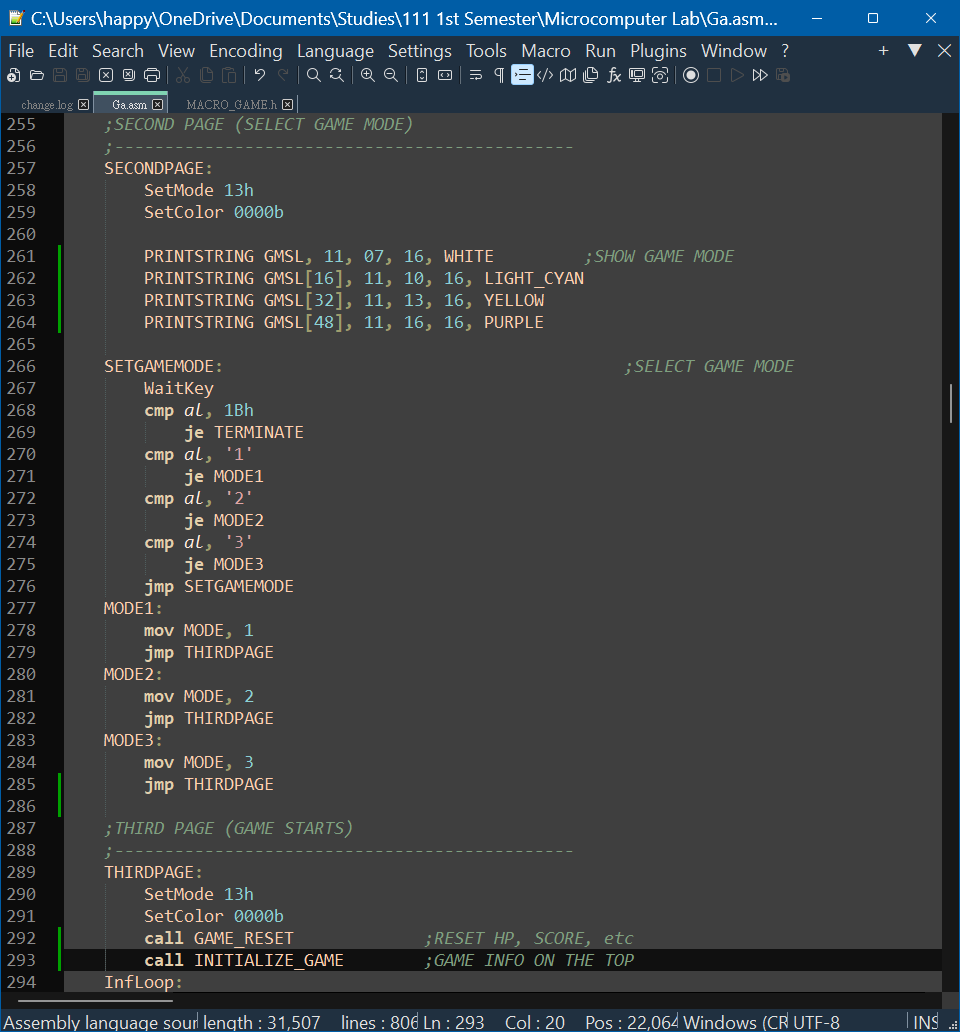
簡單的輸出字串，並讓使用者按下任意鍵繼續

* + 1. 第一頁——介紹



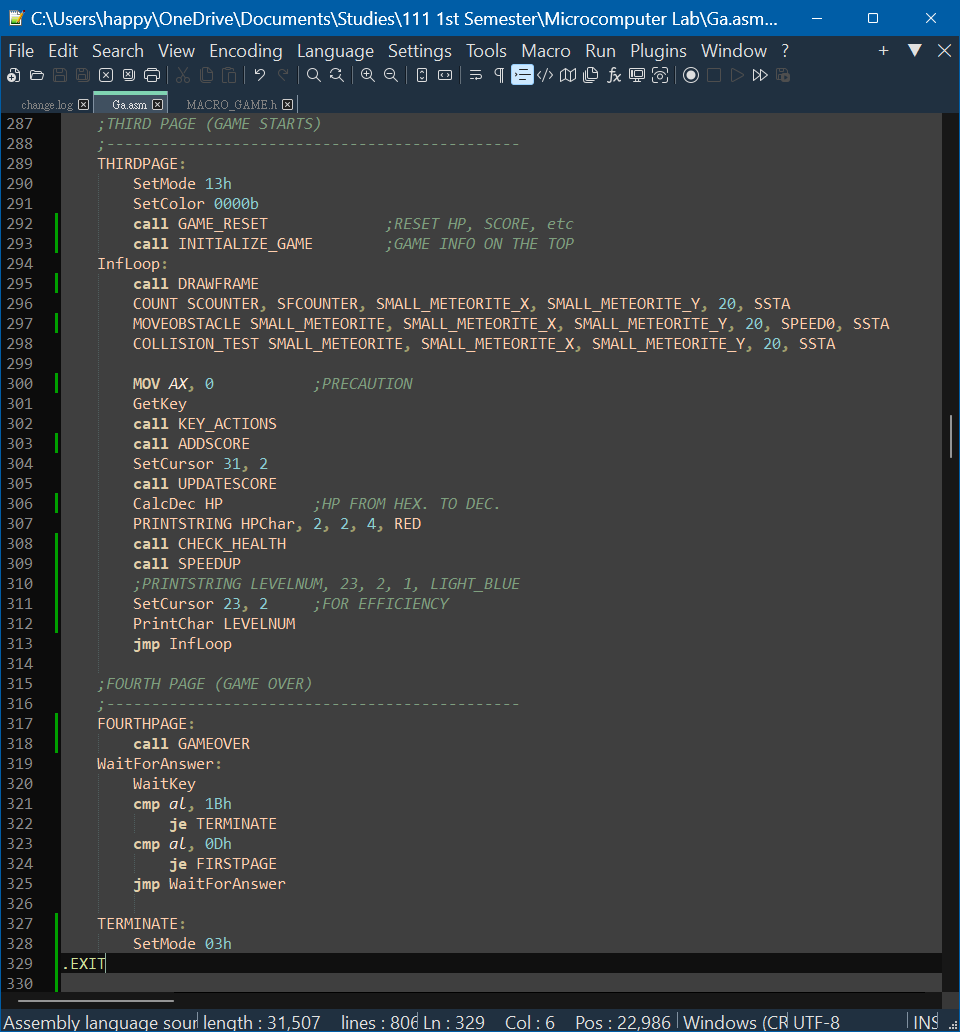
這邊給玩家簡單的遊戲控制與圖案介紹，在此畫面中僅能按下ENTER繼續或ESC退出

* + 1. 第二頁——難度選擇

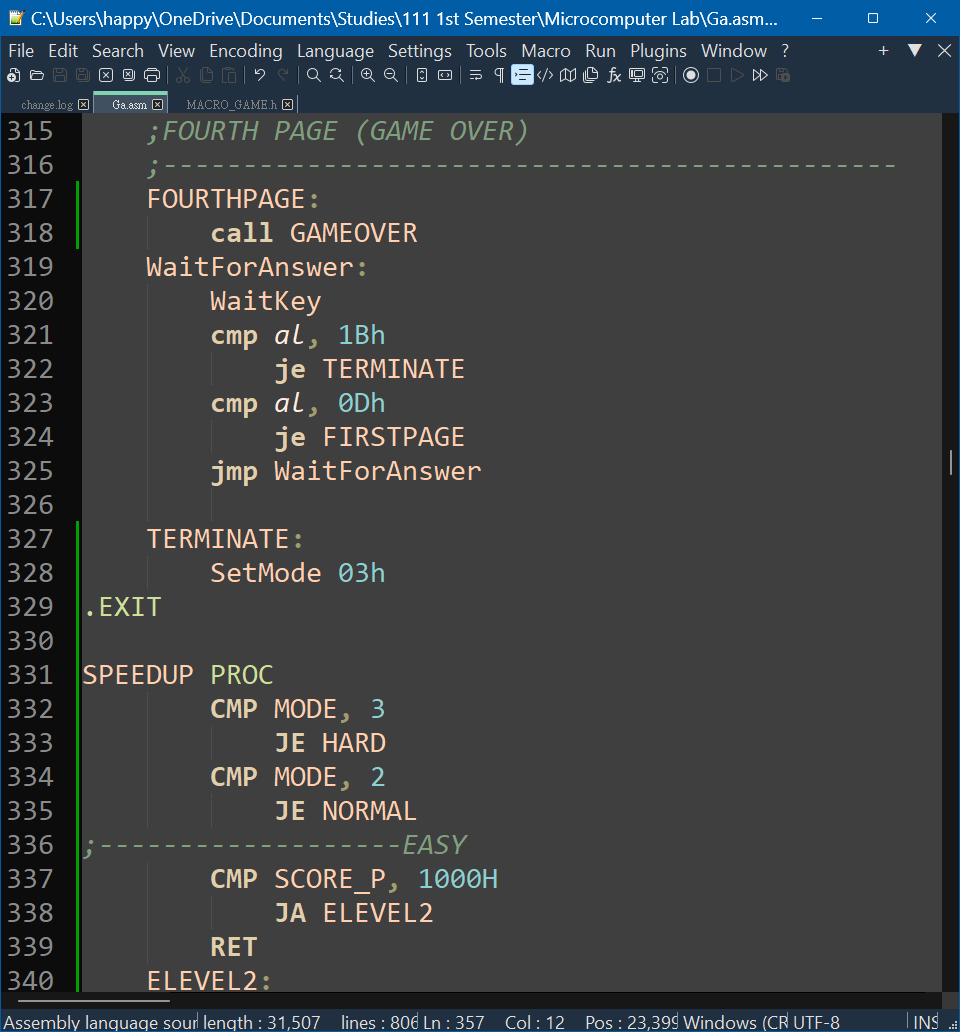


讓使用者輸入欲遊玩之難度或按下ESC退出

* + 1. 第三頁——遊戲主迴圈



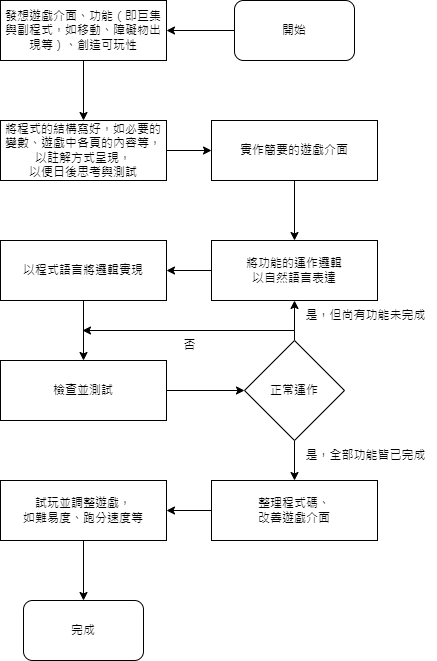
* + - 1. 遊戲實際上為一個無窮迴圈，只有當玩家按下ENTER鍵或ESC鍵或生命值小於等於零時才會執行之後的程式碼（第四頁或程式結束），所以在其他狀況下，遊戲是一直進行著的。
      2. 首先我們將遊戲中各參數設置好並畫好遊戲介面，之後才進入迴圈。與玩家移動相同，每當障礙物移動完成後，須檢查一次是否有碰撞到的情況產生，接著開放使用者輸入，程式會依照輸入做出相對應的動作。每次迴圈都會檢查目前的生命值與積分，進行結束遊戲或是加速的動作。
      3. 迴圈內的指令雖多，看起來也有先後之分，但實際上處理器運行十分迅速，因此可以將迴圈內程式碼視為同一時間點執行，各副程式或巨集在排序上並沒有太多限制，唯獨移動與碰撞判斷須留意。
    1. 第四頁——結束



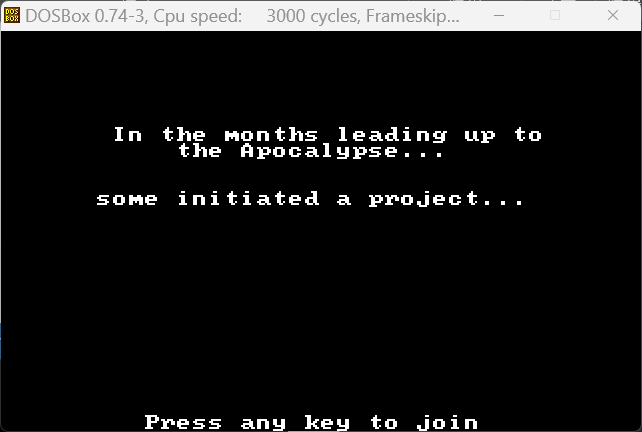
遊戲結束時會有獨立的一個畫面做總結，並提示玩家能夠重新開始或是結束遊戲

1. 流程圖

實習結果 (遊戲運作畫面)



1. 遊戲背景：

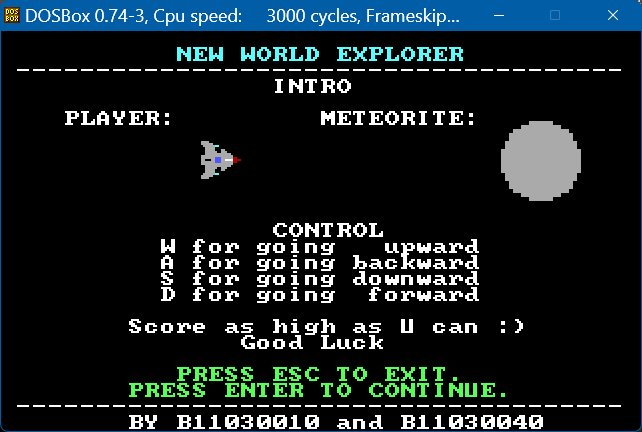


在末日來臨前幾個月……

有一些人發起了一個計畫……

現在你是新世界探索者的一員了！

1. 首頁：



W、A、S、D分別可以控制遊戲角色向上，後，下，前

1. 遊戲難度選擇畫面：



按下1、2、3，分別選擇簡單、普通、困難，於此畫面按下ESC則結束遊戲程式

1. 遊戲畫面：



1. 左上角顯示生命值，設計成2022為紀念用途。依照遊戲難度簡易、普通與困難分別給予四次、三次與二次機會，機會用完時遊戲結束。
2. 當玩家移至圖中所示位置或更右側時，生命值旁會出現BONUS字樣，代表玩家很勇，依據遊戲難度給予更多分數
3. 中間則有玩家選擇之遊戲難度與等級，等級隨遊玩時間增加而增加，每增一級，速度就增加一些，簡單與普通模式最高3級，困難模式最高5級
4. 右上角則是分數，遊戲開始後將不斷增加，並不依據等級增加而改變增量，僅按照遊戲難度與有BONUS與否而改變增量
5. 於此畫面按下ESC則結束遊戲程式，ENTER則結束遊戲進入下一頁
6. 結束畫面：



1. 生命值小於等於零時，遊戲會進入第四頁，顯示「遊戲結束」，並印出遊戲難度、等級、分數與鼓勵話語
2. 鼓勵話語依照分數與關卡決定給予“NICE WORK!”或“WELL-PLAYED!”或“INCREDIBLE!”，分數不夠高則不顯示。
3. 可按下ESC退出或按下ENTER重新開始遊戲
4. 心得 (同一組兩位組員要分開寫，附在同一份報告裡)
   * 陳○○:

（隱私保護）

* + 李○○:

（隱私保護）