# 組合語言與系統程式期末專題報告

第二組 A1005504戴佳燊

簡介：專題內容簡介與成員分工。

簡介：本專題為模擬時下最熱門的兩種消除型遊戲：神魔之塔與龍族拼圖(以下簡稱為轉珠遊戲)的模擬練習器。

成員分工：

戴佳燊：SOLO

設計方法：所使用的演算法、特殊的資料格式、例外狀況之分析與處理等。

使用的演算法：nested loop, base index。

特殊的資料格式：

boarddata byte 6\*5dup(?) ;遊戲空間

eliminatedata byte 6\*5dup(?) ;消除flag

SelX byte ;玩家座標

SelY byte

例外狀況之分析與處理：呼叫msgbox產生錯誤訊息。

PROC列表：

main:主程式進入點

clearboarddata:清除6\*5的遊戲空間(0)

cleareliminatedata:清除6\*5的消除flag(0FFh)

printboarddata:測試用，以writeint顯示遊戲空間

printeliminatedata:測試用，以writehexb顯示消除flag

randboarddata: 把遊戲空間的0換成1~6的隨機整數

findelimination:尋找初始化時需要消除的珠子

horizontalfind: findelimination的橫向版子程序

verticalfind: findelimination的縱向版子程序

eliminate:將遊戲空間與消除flag做AND運算，並清除消除flag

initialize:初始化遊戲盤面

checkstable:檢查消除flag是否有存在0

printgamepad:ASCII ART版本的printboarddata

printcell index:dword, celltype:byte:

printgamepad子程序，只印出一顆珠

printlocation:以ASCII ART印出玩家座標

game:遊戲進入點

kbprocess:處理鍵盤輸入

earaselocation: printlocation的黑底白字版本

swap direction:byte:將玩家座標的珠子與傳入方向的相鄰珠子做互換

score:計算combo進入點

findscorecells:由左至右由上到下尋找第一個可消除的珠子

horsearch:橫向搜尋(使用玩家座標作為index)

versearch:縱向搜尋(使用玩家座標作為index)

markdetonate:遞迴標記消除flag(一整團)

markthiscell:標記消除flag(一個cell)

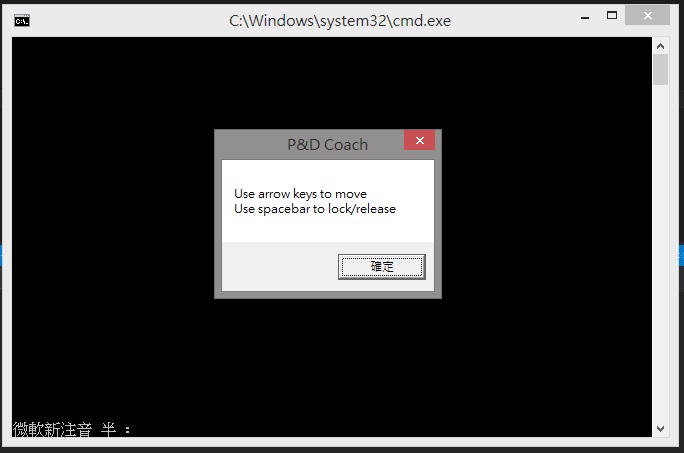
checkunknown:檢查盤面是否存在0(未知區塊)

stackboarddata:將盤面的0向上swap(頂端的0略過)

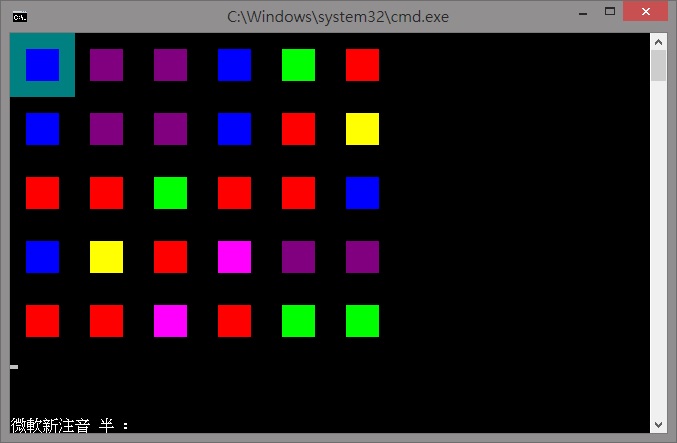
checkstack:檢查盤面是否每一行所有的0都在上方

成果：最後成果的主要功能與特色。

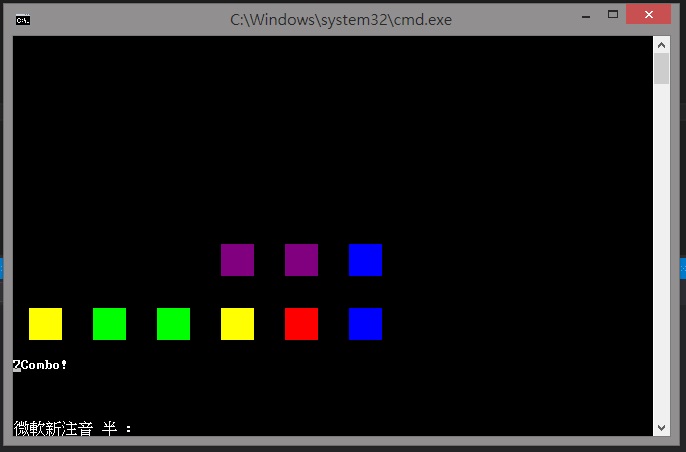
成果功能：使用者可以由方向鍵及空白鍵練習4方向的轉珠。



^模擬器進入畫面



^遊戲介面



^combo時截圖

程式特色：在消除時程式可以逐一判別待消除珠團，並以delay函式模擬轉珠遊戲時逐一消除的效果。

結論：專題製作心得、遭遇困難及解決經過、成果未來改進或延伸方向等。

專題製作心得：這是我的重修課，我在想當初被當掉會不會是因為我的期末DEMO太過簡陋。這次鐵了心寫了1372行(去年DEMO的兩倍)(可能有bad coding practice)。希望這次可以一鼓作氣ALLPASS。

遭遇困難：這次開發最困難的屬於遞迴判定該消除的cell。多虧有網路資料：cloning puzzle and dragons detecting matches and refilling the board

(附錄1)才讓我有了一點頭緒。

成果未來改進或延伸方向：這次程式沒有模擬到的一點，就是轉珠遊戲所謂的斜轉(使用8方向進行轉珠)，在未來有時間修改的時候，修改方向會式改swap跟kbprocess，利用鍵盤右邊的九宮格數字鍵實做斜轉功能。

附錄：

1.參考網址

http://www.200monkeys.com/index.php/2012/12/23/cloning-puzzle-and-dragons-detecting-matches-and-refilling-the-board/