**資料結構期末專案**

**第九組**

**黑白棋**

**(Reversi)**

**黃靖涵、黃欣惠、陳姸君**

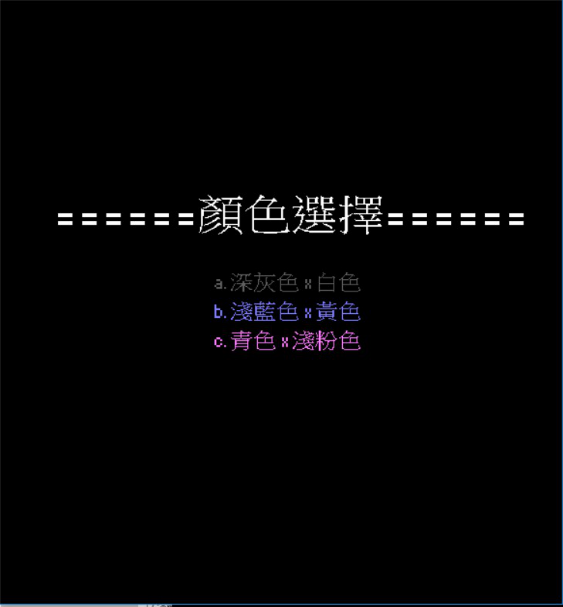
**李欣倪、盧乙嫻**

遊戲功能介紹

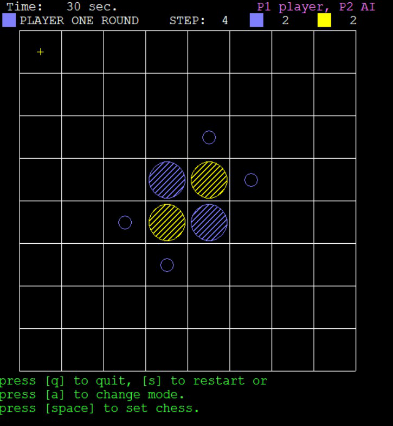
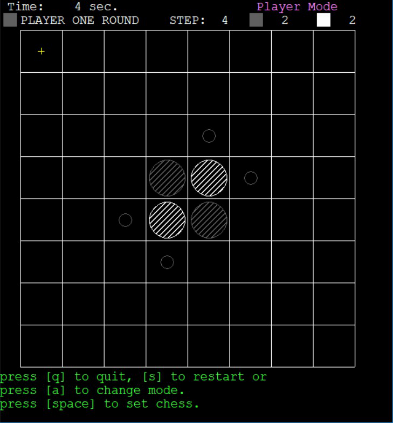
1. 主畫面：



2. 進入遊戲：



(a)進入顏色選擇頁面，其中有三種顏色組合可供使用者選擇，讓使用者選擇自己喜歡的顏色搭配。



(b)選擇完顏色之後，即直接進入遊戲。

3. 選擇模式：

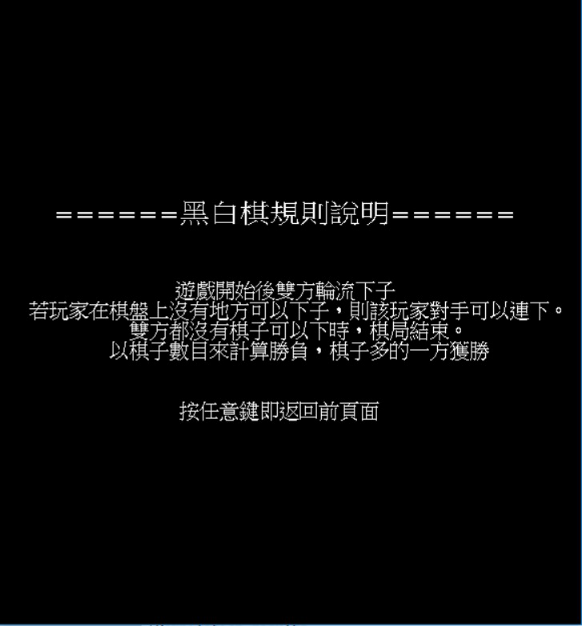
(a)在遊戲進行中，使用者可以切換三種不同模式，分別為玩家vs玩家、玩家vs AI、AI vs AI。

https://lh6.googleusercontent.com/4jvKLHekj0C7zAnzbxUoF8oGRDFS89FnOxMMcGaOZiZ6L_G__2ilBKcV1dAzhkNUC2grzjDYYFHM0KjDvxbbqSTMu04HA9qhdhUPhHimsc6QVrJawKUuVyfl69d-i3uskPjJOrg5

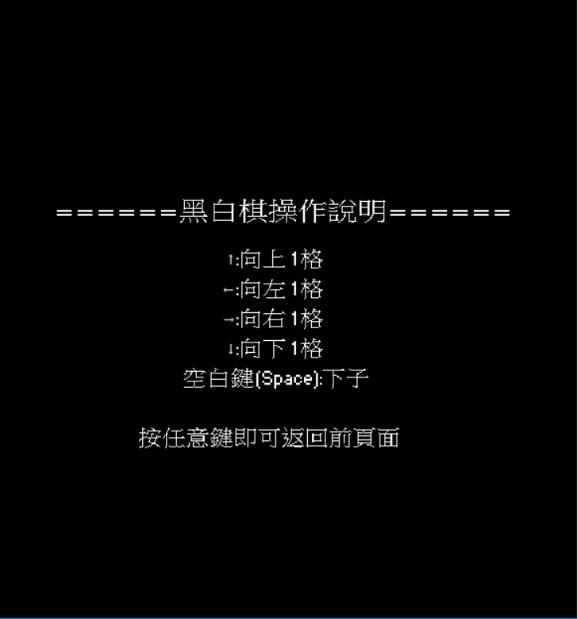
https://lh6.googleusercontent.com/MCKItmPKs0jzAficNzRb4aZ0sxsGmTDaDFrZoOr3v1B8_Ces7Tk9uhSFl994RfrW5hLfda2FzU8SjTraAnZDNOkor2V0Yu7haPiquBqOU48LEu6o7xBxdySU4rsNTD57eXFoge3X

https://lh3.googleusercontent.com/ozrMP3cfrbr_WAz4b4fss9wIj23aoD2Mz4mIOhsyPebF_Bhb3aq6Rr_6CPMLyxd8YDG-SpzFO3UzItX-PhU30LBHbDD7jqucb_r_BoAeqWrcOd88rLrZyHhwoMBEOs2RCfnqNeOA

4. 遊戲介紹：

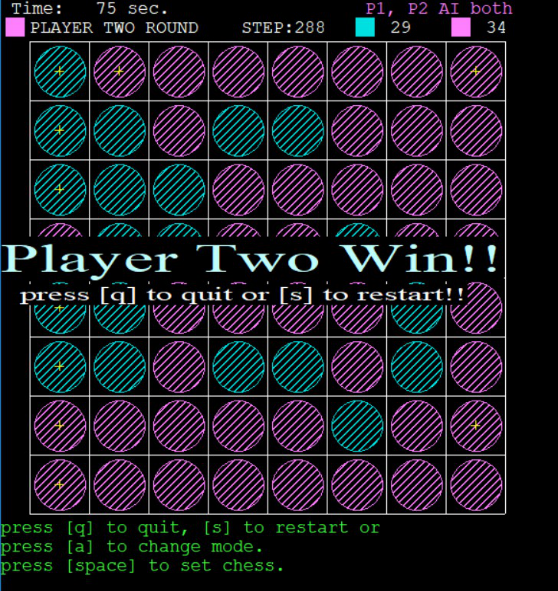


5. 操作說明：

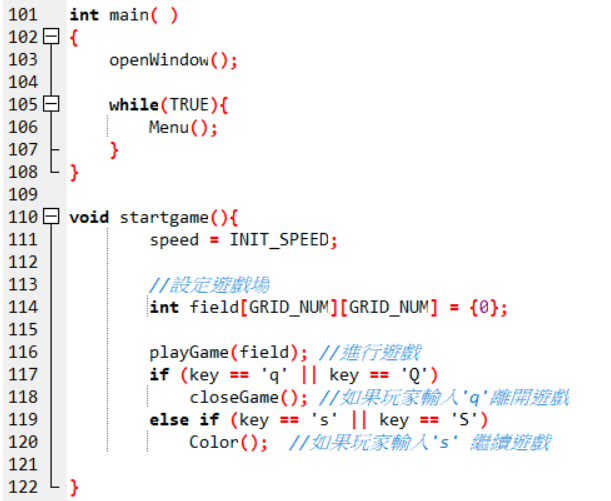


6. 遊戲結果：(a) 按q則可離開遊戲頁面。

            (b) 按s則會回到選擇顏色的頁面，繼續遊戲。



主頁面

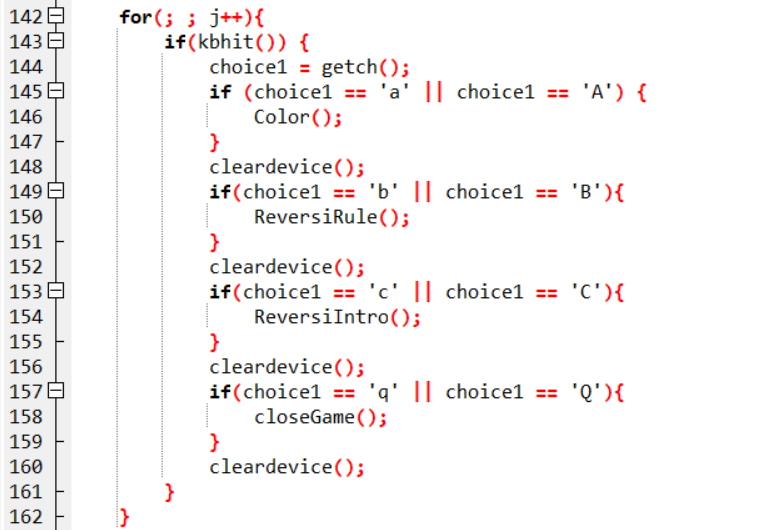


**函式Menu、startgame、Color執行順序：**

1. 開啟視窗後，會先進入Menu()函式。

2. 在Menu()函式執行完之後，接著再到startgame()函式。

3. 開始遊戲直到遊戲結束後，輸入q關閉遊戲，輸入s則返回選擇Color()函式，繼續遊戲。



**函式Menu：**

1. 若使用者輸入a/A，利用getch()讀入電腦，則連到Color()，選擇完Color之後，直接進入遊戲。

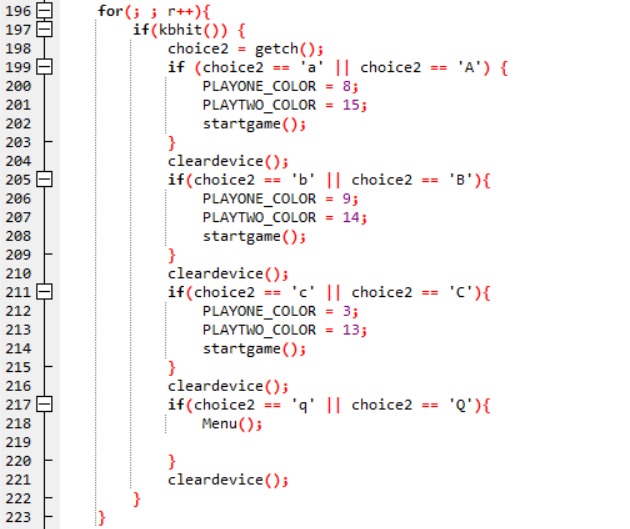
2. 若使用者輸入b/B，利用getch()讀入電腦，即進入遊戲說明頁面。

3. 若使用者輸入c/C，利用getch()讀入電腦，即進入操作說明頁面。

4. 若使用者輸入q/Q，利用getch()讀入電腦，則離開遊戲。

選擇顏色頁面

https://lh6.googleusercontent.com/V7rNWfsB7XOa3IkL-g1YjU_-X0j17CK66e6xZJBpk_qV-VZQK2yTVAFUa0QLPuzZvXYsB-seQSM7RrBYtlu7rV2eH1gtu36ylItwqzojlxtFRVnlMvwfR-_ofSlpGW0Tm79FQwnV



**函式Color：**

1. 先將PLAYONE\_COLOR、PLAYTWO\_COLOR設為integer的全域變數。

2. 當玩家選擇a、b、c任一顏色組合之後，利用getch()讀入電腦，便會使玩家一與玩家二的棋子顏色皆呈現該顏色組合。

3. 定義完雙方玩家之顏色後，即進入startgame函式，開始遊戲。

遊戲介紹頁面



**函式ReversiRule：**

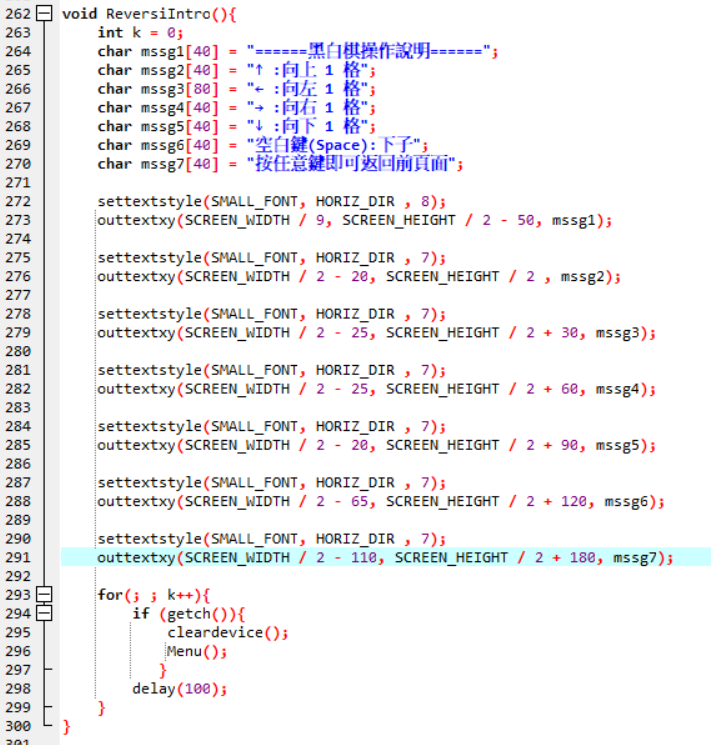
1. 將欲呈現的字放進char裡面

2. 用settextstyle設定字體、字體方向及字體大小

3. 用outtextty設定字體位置以及要呈現的字

4. 最後，若getch到任一鍵則return

遊戲操作說明頁面



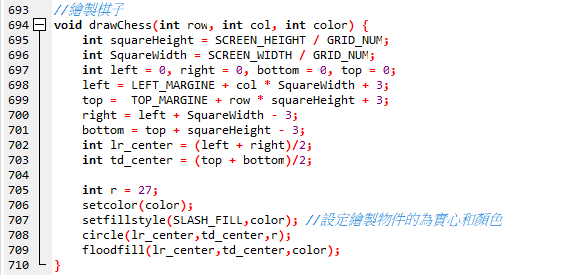
**函式ReversiIntro：**

1. 將欲呈現的字放進char裡面

2. 用settextstyle設定字體、字體方向及字體大小

3. 用outtextty設定字體位置以及要呈現的字

4. 最後，若getch到任一鍵則return

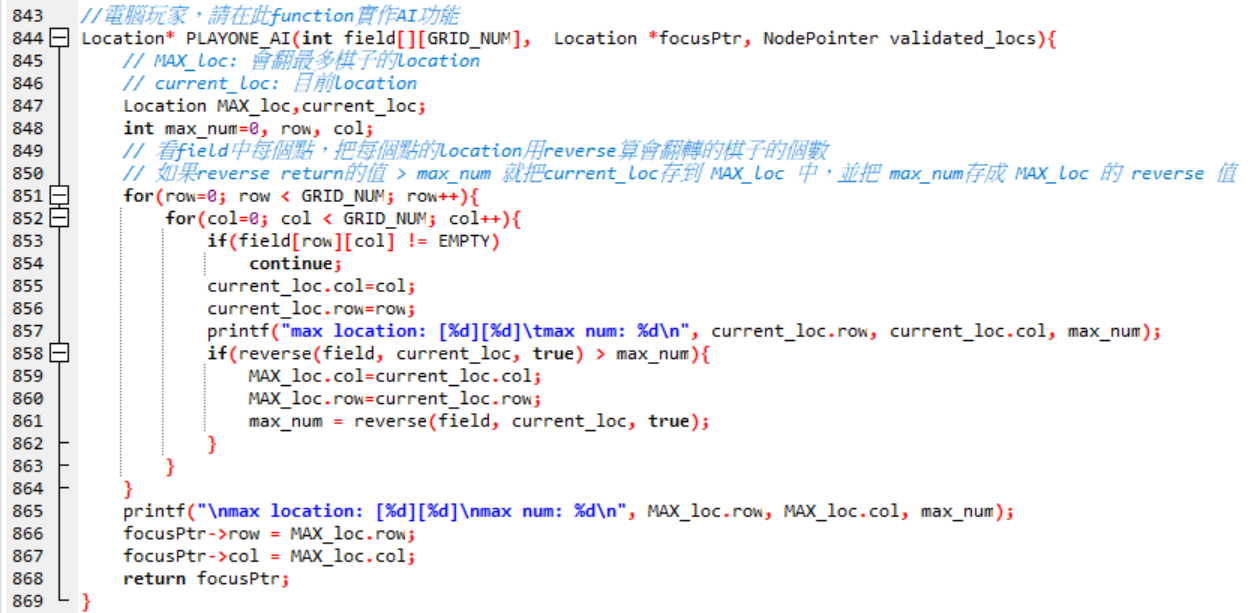


**函式drawChess：**

1.用setfillstyle宣告顏色以及填色的方式

2.接著用circle函數，在座標(lr\_center,td\_center)畫圓形（棋子），設半徑為r

3.用floodfill在座標(lr\_center,td\_center)填色，顏色為color。

AI

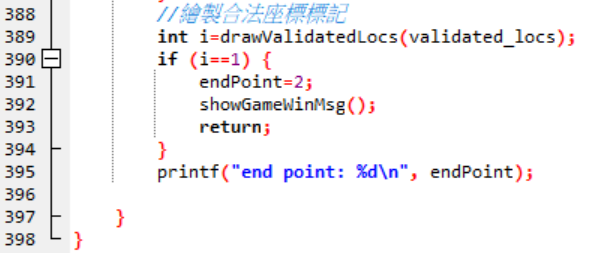
**函式PLAYONE\_AI：**

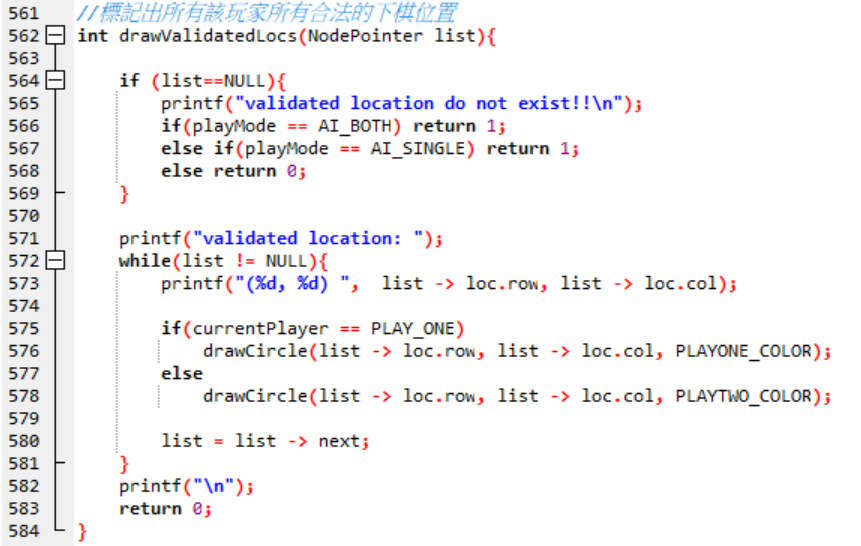
1. 宣告位置struct MAX\_loc以及current\_loc，MAX\_loc用來儲存當下搜尋到可翻轉棋子數最多的可下子位置資訊，而current\_loc則用來儲存目前的位置資訊。

2. 另外宣告整數變數max\_num，用來儲存當下搜尋到可翻轉棋子數最多的數量（初始值為零），由於此AI策略為「選擇當下可下的步數中，可翻轉對手棋子最多的一個」，所以後面每當搜尋到比max\_num大的數字，就會存進max\_num。

3. 用兩個for迴圈去檢查棋盤(field)中每個點，把每個點的location傳入reverse，去算該點會翻轉對手棋子的個數。如果該個數結果比max\_num大，就會存進max\_num，並用MAX\_loc去紀錄該點的位置資訊。

4. 等for迴圈跑完，就將MAX\_loc位置資訊複製到focusPtr，並回傳出去。

←在函式playGame中



**函式playGame及函式drawValidatedLocs判斷：**

1. 原本程式碼執行後，無法顯示遊戲結果(不能跑進函式showGameWinMsg()中)，所以在函式drawValidatedLocs中新增判斷式(556~568行)。

2. 並讓函式可以return boolean(0代表遊戲尚未結束，1代表遊戲結束)。

3. 在函式playGame中將函式drawValidatedLocs return的值存入變數i中，如果i=1(遊戲結束)，就將endPoint=2後顯示遊戲結果，並跳出函式playGame。

心得感想

   這次的專案中，我們除了學習到許多有關<graphics.h>的應用外，也學習到如何看別人的程式碼。以前上程式設計時候寫的專案是沒有使用graphics函式庫的，整個程式幾乎都是手刻，所以畫出來的介面不僅很難對齊，也很難找出問題的原因，但是透過graphics函式庫，可以輕易地畫出直線、圓圈，也可以更有系統的寫程式。另外，因為在寫新的函式的同時也要與現有的函式結合，除了學習到不一樣函式庫的應用外，也要學習看懂別人的程式碼，將一個範例的程式碼理解之後，再加上自己的思考邏輯，因此在後續的應用上就會相當有效率，整個專案下來，也比較願意耐住性子去看別人的程式。

   另外，這次的過程中也很深刻的體會到團隊合作的重要性，初期剛拿到助教給的程式碼的時候因為才剛接觸新的函式庫所以撞牆很久，但大家都會耐心地搜尋資料並一直嘗試，因為大家互相幫忙、漸漸熟悉函式庫之後，最後才能做出一個完整的遊戲出來。除了透過不斷上網查詢別人的作法並從中學習外，也學習到如何將持續測試，找出問題所在，不斷嘗試修正直到問題解決，所以此次專案讓我們收穫許多，也學習到許多新的事物，非常開心能有此次的學習機會!

參考資料

<https://www.geeksforgeeks.org/setcolor-function-c/>