

專案名稱：deTetris

組員：楊子承 B12902064

#### 專案介紹：

在玩俄羅斯方塊的時候，常常會發生「想到的方塊一直不來」的時候吧。在這個遊戲中，我強化了這個概念，製作出來「絕對會給你不需要的方塊」的遊戲。

#### 製作歷程：

一開始先創造出一個正常的俄羅斯方塊遊戲，這部份除了語法不熟以外沒有什麼大問題。

之後在製作方塊生成 AI 的時候遇到了記憶體不足的問題，礙於記憶體大小和運算效能的限制，我不能製作太深的遞迴，因此我只做了一層的枚舉，再對每一個枚舉去判別盤面的好壞，而生產優先選擇不能消排、再選擇在最好情況下會造成最高高度的方塊。

在做完之後，我才發現在最理想的情況（也就是全部出 O 和 S 的情況），這個遊戲可以達到絕對消不了排的情況，但是這樣就沒有可玩性了，於是我決定降低 AI 的難度，因此我開了一個 boolean 陣列來記錄目前出過的方塊，並且限制 AI 只能從尚未選擇過的方塊中生成，而 boolean 陣列則會定期地進行初始化。

但是從頭到尾都是同一個難度的 AI 感覺還是有點無趣，因此我又寫了一個 limit 函數，由玩家的目前成績和目前的盤面狀況來控制 boolean 陣列的初始化與否，當玩家處於方塊堆得很高的時候（也就是即將 gameover 的時候），就會調降難度，當玩家的消排很多的時候，就會很頻繁地初始化，也因此麻煩的方塊的出現頻率就會非常高。

#### 改進：

1. 目前的演算法只是單純地暴力枚舉每一個方塊的每一個可能位置，因此在效能上很慢。另外我的演算法只有考慮到直接落下的狀況，沒有計算到插洞的情況，因此會有誤判。
2. 目前的狀況判定機制只有考慮到最高高度和消掉的行數，並沒有考慮到整體局面的可發展性(如盤面的整齊度或是洞的數量、大小等)

#### 心得：

在這次的專案製作之後，我學到了如何去製作一個敵人 AI，去實作盤面分析和去產生一個對自己最好的結果，希望未來如果有機會的話，可以製作更深的遞迴和更全面的判斷。