

視窗程式設計 期末專題

Qt-晃方世界

組別：第5組

4 1 2 4 3 1 0 1 伍翊瑄

4 1 2 4 3 1 0 3 林采儀

4 1 2 4 3 1 0 8 鄭又瑄

1. 主題與動機

- **主題:** 基於 Qt 的俄羅斯方塊遊戲——「晃方世界」。
- **動機:** 高中時，經常和同學一起玩 TETRIS 的線上俄羅斯方塊 PK，但總是最早被淘汰，心有不甘。為了提升自己的遊戲技巧，決定開發一款俄羅斯方塊遊戲，這樣可以調整遊戲內部數據（例如掉落速度），並進行針對性的練習。同時這也是一個學習如何應用 Qt 框架進行遊戲設計與開發的好機會。

2. 系統需求

- **硬體需求:** 開發環境(筆電)。
- **軟體需求:** Qt 開發環境版本(6.7.2)。

3. 系統設計與實作

□ 功能規劃

- **核心功能:** 俄羅斯方塊遊戲基本玩法。
- **衍生功能:** 背景音樂、難度調節等。

3. 系統設計與實作

□ Qt 框架應用

- **遊戲主視窗和界面佈局:** 使用 QWidget 或 QMainWindow 類來創建。
- **遊戲面板繪製:** 使用 QPainter 類，它允許在窗口上繪製方塊、分數等動態內容。
- **音效控制:** 通過 QCheckBox 和 QSoundEffect，管理音樂開關。

□ 信號與槽機制應用

- **按鈕點擊:** 如開始、暫停、開關音樂等操作，當玩家點擊按鈕時，會發送 clicked 信號，並觸發對應的槽函數。
- **鍵盤事件:** 透過 keyPressEvent 來捕獲鍵盤按鍵，控制方塊的移動和旋轉。

3. 系統設計與實作

□ 遊戲邏輯設計

- **方塊定義:** generateNewPiece() 函式，七種基本方塊形狀 (O, I, L, J, Z, S, T)，每個形狀由四個 QPoint 定義其相對位置。
- **方塊生成:** 使用 std::rand() 隨機生成一個索引 (範圍為 0 到 6)，以此來實現隨機生成七種形狀。
- **下一個方塊的預覽:** 使用 QPixmap 創建預覽區域，調用 generateNewPiece() 繪製下一個方塊的圖像，並透過 mainWindow 的方法更新顯示。
- **方塊消除:** 檢查當行是否填滿的變數 isFull，若變為 true，則消除該行並將上方的方塊下移。

3. 系統設計與實作

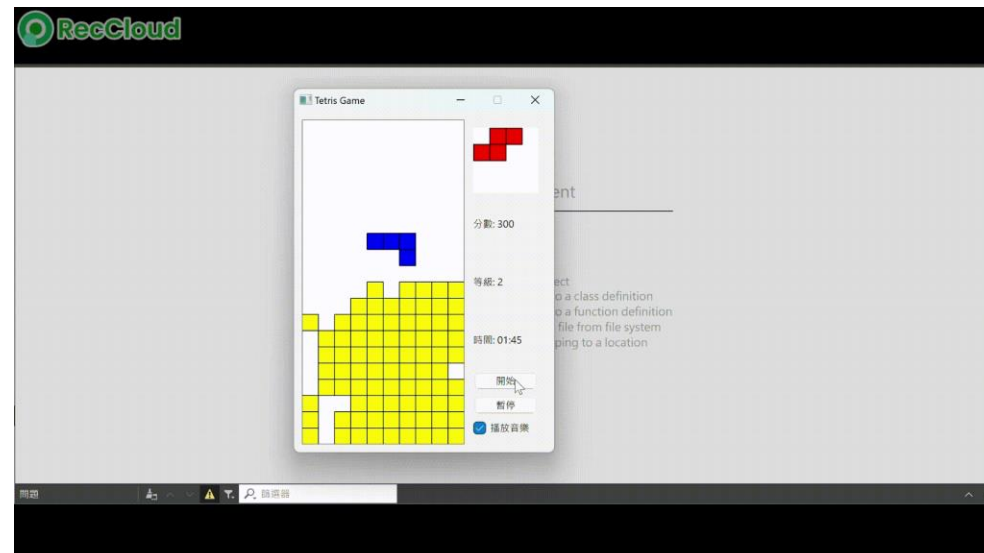
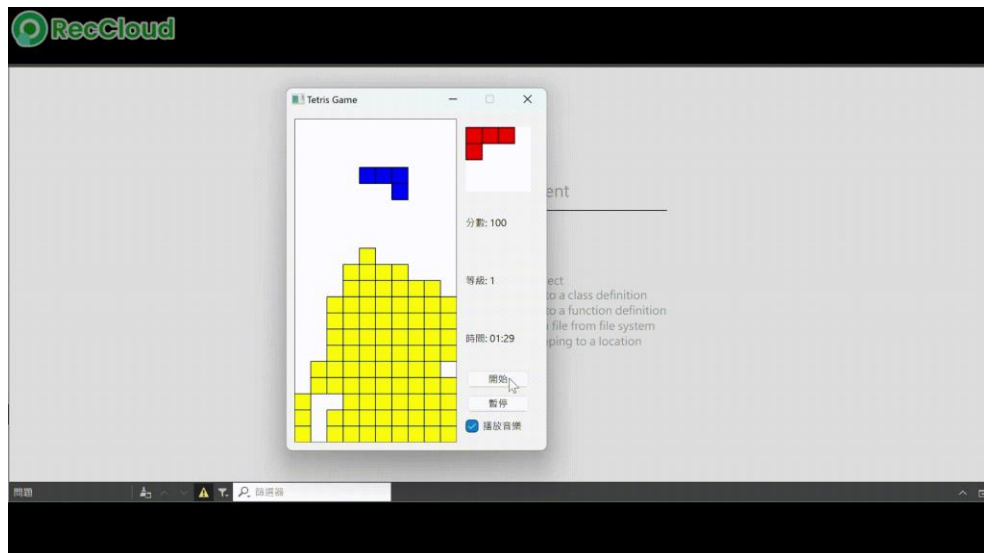
□ 遊戲邏輯設計

- **方塊移動(←↓→、Space):** 使用"左、下、右方向鍵"和"空白鍵"，分別觸發左移函式 `moveLeft()` 和下移函式 `dropStep()` 以及右移函式 `moveRight()`、快速下移函式 `moveToBottom()`。
- **方塊旋轉(↑):** 使用"上方向鍵"觸發旋轉函數 `rotate()`。通過矩陣的轉置和反轉實現 90 度旋轉，且避免超界與碰撞。**(先轉置，將其行列對調;後反轉，將每列的元素左右顛倒)**

3. 系統設計與實作

□ 遊戲邏輯設計

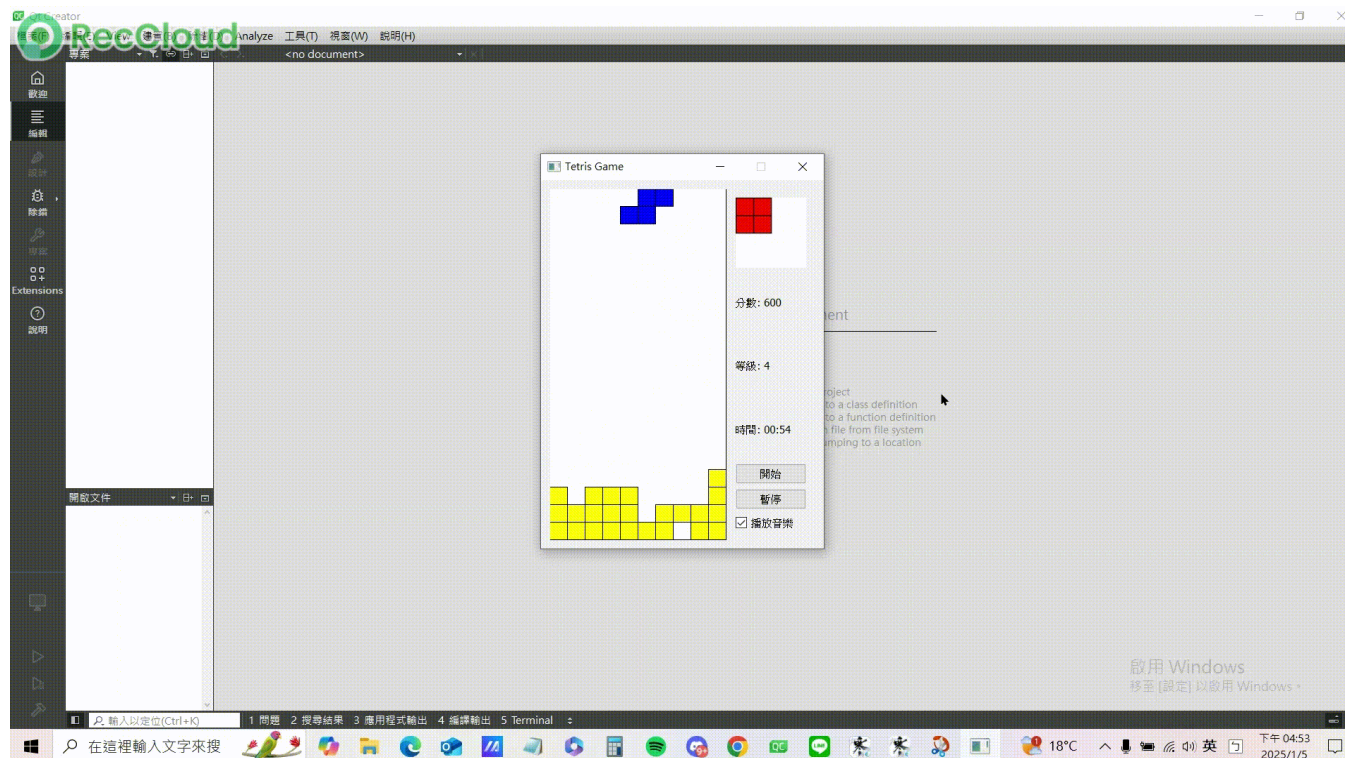
- **得分和等級:** 隨等級提高，遊戲加速並引入視窗晃動和浮動等視覺效果，提升遊戲難度。
- 第2級 (加入左右晃動)
- 第3級 (加入上下浮動)



3. 系統設計與實作

□ 遊戲邏輯設計

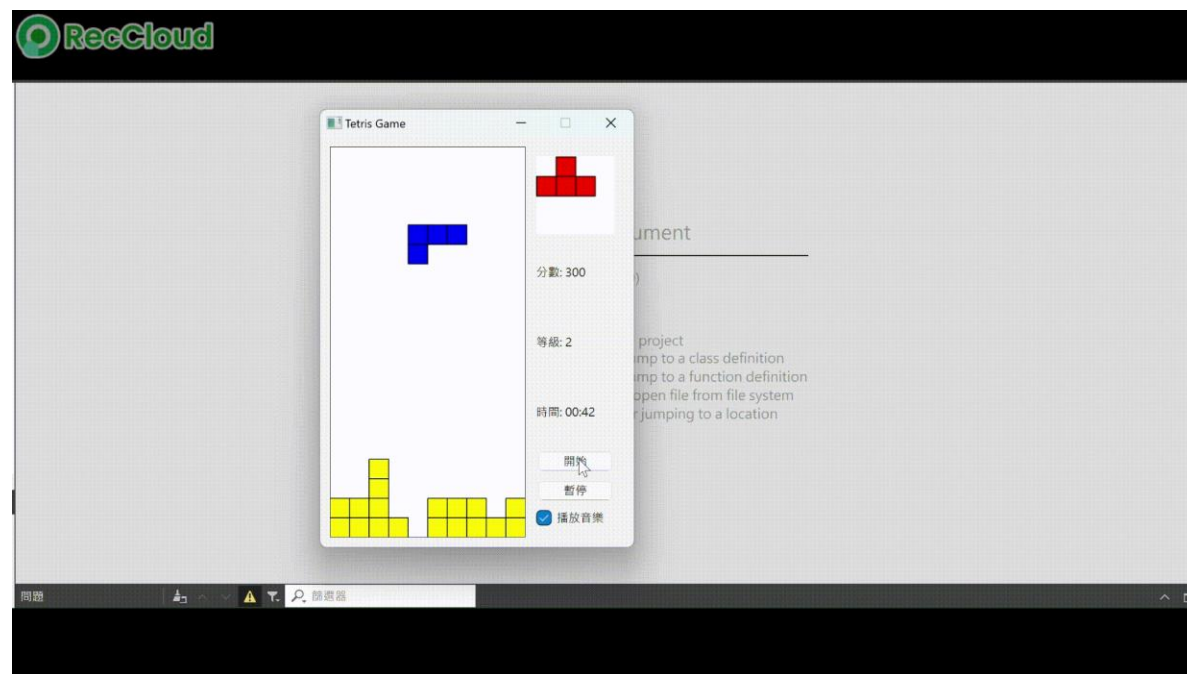
- **得分和等級:** 隨等級提高，遊戲加速並引入視窗晃動和浮動等視覺效果，提升遊戲難度。
- **第4級 (加入方塊左右移動)**



3. 系統設計與實作

遊戲邏輯設計

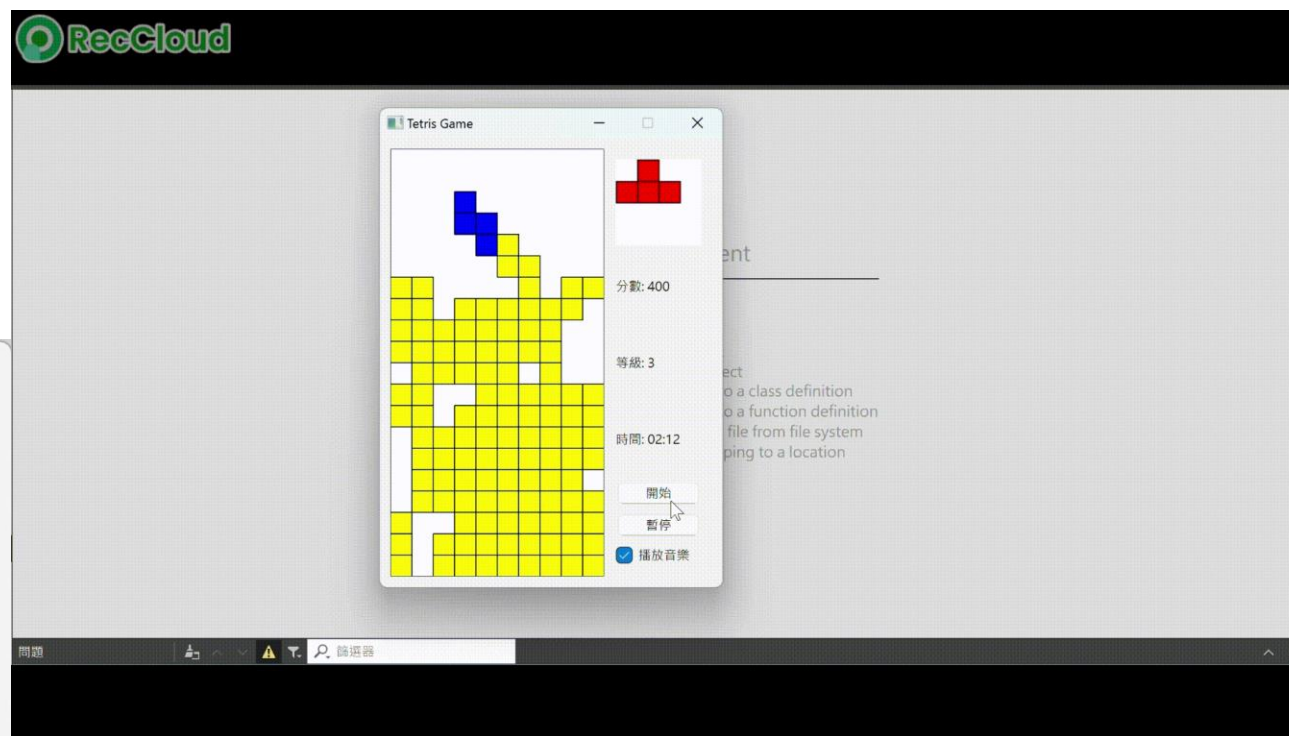
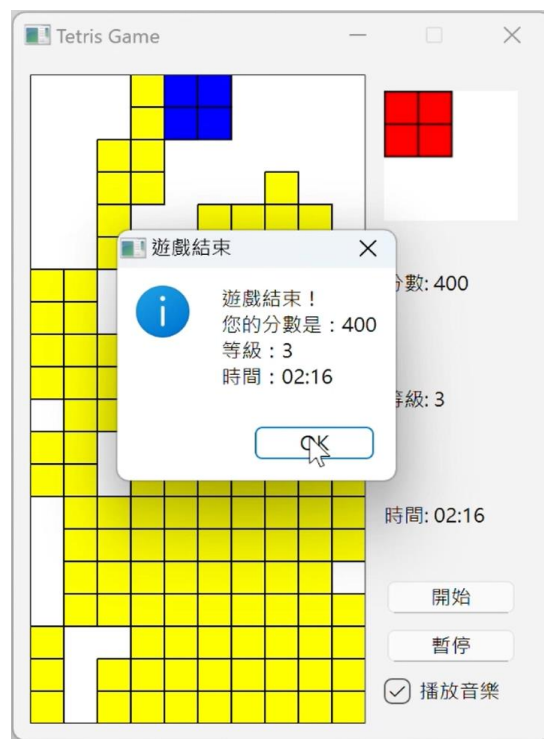
- 暫停機制的觸發條件
- 暫停時，應停止所有計時器、動畫或遊戲邏輯的更新。
- 暫停狀態的保存與恢復
- 在恢復時，確保遊戲邏輯和動畫能從中斷點繼續執行。
- 所有暫停前的動作（例如方塊的下落、計分等）應保存以便恢復。



3. 系統設計與實作

遊戲邏輯設計

遊戲中止條件與結束畫面: 無法合法放置新方塊則觸發遊戲結束, 並跳出結束視窗。



4. 測試與結果

□ 測試案例

- 生成的方塊形狀是否隨機，並覆蓋所有可能的形狀。
- 測試方塊生成、旋轉和消除及音效同步功能。
- 檢查視窗晃動是否影響遊戲操作。

□ 測試結果

- 方塊的隨機生成均符合預期，且每種形狀的出現概率平均分布。
- 核心功能運行穩定且同步。
- 功能表現穩定，音效開關能準確被控制，視窗晃動在多次測試中未出現卡頓現象。

5. 結論

□ 優勢

- 簡單直觀的操作方式，易於玩家上手。
- 遊戲的難度設計循序漸進，既適合初學者，也能滿足進階玩家的挑戰需求。
- 核心功能表現穩定且同步。

□ 學習收穫

- 團隊合作能力。
- 熟悉了界面設計和信號與槽機制的運用。
- 提升了 Qt 框架實踐能力。

6. 附錄

- ❑ 參考資料：<https://github.com/tashaxing/QtTetris.git>
- ❑ 遊戲影片連結：https://youtu.be/pa5jJAjJ_44

END