

Programming Design (II) Project 2 【2048】

report

資訊四 F74004046 劉慎之

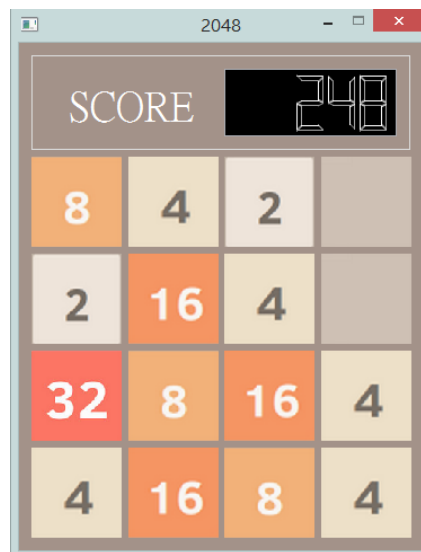
★ 【程式執行方式】：

利用 Qt creator 來執行建立出執行檔之後便可以執行遊戲

★ 【遊戲執行畫面】：



❶ 遊戲開始畫面



❷ 遊戲進行畫面



❸ 遊戲結束畫面

★ 【遊戲功能說明】：

- 1/ 遊戲開始執行即會出現開始畫面
 - 2/ 藉由上下左右移動的方式移動方塊，遇到相同數字會合併
 - 3/ 直到出現數字 2048 的方塊，遊戲即停止
 - 4/ 若遊戲失敗，會彈出視窗，決定要重新開始遊戲還是要離開
- 遊戲設置計分功能，可以記錄下分數

★ 【UML class diagram】：



Tile
<ul style="list-style-type: none"> - number:int - image:QPixmap*
<pre> <<constructor>> Tile(QWidget *parent = 0); + paintEvent(QPaintEvent *); + getNumber() + setNumber(num:int) + isEmpty() + upgrade() </pre>

★ 【程式設計邏輯】：

- 一開始執行時先執行 createboard 函數，先把 ui 界面中拉出來的 frame 指定給形態為 Tile 的 pointer，依序為 tile [0] [0] ~ tile [4] [4]

★ 相對應的坐標如下

```

+-----+-----+-----+-----+
|   |   |   |   |   |
| 0 · 0 | 0 · 1 | 0 · 2 | 0 · 3 |
|   |   |   |   |   |
+-----+-----+-----+-----+
|   |   |   |   |   |
| 1 · 0 | 1 · 1 | 1 · 2 | 1 · 3 |
|   |   |   |   |   |
+-----+-----+-----+-----+
|   |   |   |   |   |
| 2 · 0 | 2 · 1 | 2 · 2 | 2 · 3 |
|   |   |   |   |   |
+-----+-----+-----+-----+
|   |   |   |   |   |
| 3 · 0 | 3 · 1 | 3 · 2 | 3 · 3 |
|   |   |   |   |   |
+-----+-----+-----+-----+

```

而每一個 tile 中都有一個 number，初始為 0，代表是空的格子，顯示的圖示為 0.png

- 在程式執行時會依據 number 來替換圖片，

Ex： tile[1][1]的 number 為 2；則它顯示的圖片會是 2.png，若是 4 則是 4.png etc

3. 再來則執行 `startgame`，將隨機兩格 `number` 變成 2，然後使用者開始使用上下左右移動方塊，若是按上，則會先用 `movepossible` 檢查是否能移動
判斷的規則是先判斷是否所有格子的 `numner` 大於 0，若沒滿代表可移動，若是滿了則檢查相同的行列，是否有 `number` 相同且兩兩相鄰的格子，如果有代表可移動。
4. 判斷可以移動則執行 `moveup`，會先執行 `emptytileMoveDown` 將空的格子移動到最下面，再來看往上時有沒有相同 `number` 的格子相鄰，如果有則下方的格子歸零，上方的格子執行 `upgard` 函數(`number*2`)
Ex：[0][0]和[1][0]`number` 都是 2，往上移動時[1][0]變為 0，[0][0]`upgrade` 為 4，再來把 `score` 加上 `upgrade` 後的數字，然後觸發 `changed` 函數，改變 `lcdnumber` 顯示的分數，
5. 之後如果 `upgrade` 之後有出現 2048 則 `endgame`，移動完之後；則執行 `randomtile`，抽出隨機兩個空格子並將他們 `number` 變為 2 或 4，按其他方向時以此類推最後出現 2048 或者沒出現 2048 而且也無法在移動則執行 `endgame`，會跳出一個視窗顯示 2048 或者 `gameover`，下方則有按鈕 `reset` 和 `exit`，按下 `reset` 則會觸發信號槽裡的 `reset` 函數，所有格子 `number` 歸 0 分數也歸 0，然後重新執行 `startgame`，按下 `exit` 則是關閉程式。