國立臺北科技大學

2022 Spring 資工系物件導向程式實習期末報告

TETR.IO

一張含有 圖形, 平面設計, 字型, 紅色 的圖片

自動產生的描述

第49組

**目錄**

**一、 簡介**

[1. 動機 1](#_TOC_250015)

[2. 分工 1](#_TOC_250015)

[二、 遊戲介紹](#_TOC_250014)

1. [規則 1](#_TOC_250013)
2. 遊戲圖形 2
3. [遊戲音效 5](#_TOC_250011)

[三、 程式設計](#_TOC_250010)

1. 程式架構 7
2. 程式類別 .8
3. 程式技術 9

[四、 結語](#_TOC_250006)

1. 問題及解決方法 9
2. 時間表 11
3. [貢獻比例 12](#_TOC_250003)
4. [自我檢核表 13](#_TOC_250002)

[5. 收穫 13](#_TOC_250001)

1. [心得、感想 13](#_TOC_250000)
2. 對於本課程的建議 14

# **一、 簡介**

## 動機

俄羅斯方塊是我小時候玩的遊戲，需要動腦的遊戲非常吸引我，所以在這次實習時就選了這款遊戲來復刻，在這一次的實驗中選的是TETR.IO這個遊戲，包含四個模式(40L BLITZ ZEN CUSTOM)，在設計時也玩了這款遊戲非常多次為了體驗不輸原版的體驗，所以也盡可能的完美複刻。

## 分工

## 組員一 呂凱達(素材，聲音，遊戲架構，遊戲功能實現)

## 組員二 陸嘉康(遊戲架構，遊戲功能實現)

# **二、 遊戲介紹**

## 遊戲說明

遊戲主要是一款經典的益智遊戲，玩家需要控制不同形狀的方塊（也稱為俄羅斯方塊）下落，並將它們放置在螢幕上形成完整的水平行。方塊可以沿著螢幕的垂直方向進行移動，並且可以旋轉以適應空間。遊戲的目標是通過放置方塊來創建完整的水平行，當一行被填滿時，該行將被清除並獲得分數，我們另設有作弊模式(按t進入，按a加快速度，按s減慢速度)。

**模式介紹**

40L:計算多久消除40L的時間

BLITE:在兩分鐘內，遊戲內設有Level，計算玩家可以達到多少 Level和用一個公式去計算玩家的分數，公式(硬降格數\*2 軟降格數\*1 消一:100 消二:300 消三:500 消四:800)

ZEN:休閒模式，可以保持玩家的遊玩紀錄

CUSTOM:可以客制化自己的桌布

## 遊戲圖形

**一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述**

**一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體軟體 的圖片

自動產生的描述**

一張含有 螢幕擷取畫面, 數位合成, 電腦遊戲, 3D 模型 的圖片

自動產生的描述

一張含有 螢幕擷取畫面, 像素, 藝術 的圖片

自動產生的描述

一張含有 space, 宇宙, 太空, 星星 的圖片

自動產生的描述

一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體軟體 的圖片

自動產生的描述

一張含有 螢幕擷取畫面, 多媒體軟體, 繪圖軟體, 天空 的圖片

自動產生的描述

**一張含有 螢幕擷取畫面, 文字, 多媒體軟體, 繪圖軟體 的圖片

自動產生的描述**

## 遊戲音效

以下陳列在遊戲中使用到的音效。

|  |  |
| --- | --- |
| **音效類別** | **音效檔案** |
| 背景音樂 | Arial\_City.wav  To\_The\_Limit.wav  The\_Great\_Eastern\_Expedition.wav  Morning\_Sun\_Kamoking.wav  In\_Sorrow\_And\_Pains\_Mirera.wav  Piercing\_Wind.wav  Success\_Story\_Akiko\_Shioyama.wav  Hyper\_Velocity.wav  Philosophyity.wav  Classy\_Cat.wav |
| 觸碰菜單 | Touch\_Menu.wav |
| 點擊菜單 | Click\_Menu.wav |
| 返回音效 | Back\_Menu.wav |
| 觸碰打勾型菜單 | Touch\_Check\_Menu.wav |
| 點擊打勾型菜單 | Click\_Check\_Menu.wav |
| 方塊旋轉 | Cube\_Rotate.wav |
| 方塊移動 | Cube\_Horizontal\_Move.wav |
| 方塊下降 | Cube\_Decline\_Move.wav |
| 方塊切換 | Cube\_Switch.wav |
| 方塊觸碰底部 | Cube\_Touch\_Bottom.wav |
| 方塊清除 | Cube\_Clear.wav |
| 方塊完全清空 | Cube\_Full\_Clear.wav |
| 遊戲失敗 | Game\_Over.wav |
| 遊戲完成 | Game\_Finish.wav |
| 遊戲結束端動畫音效 | Game\_Over\_Menu.wav |
| 升級 | Level\_Up.wav |
| 遊戲退出過程 | Exit\_Process\_Game.wav |
| 遊戲退出 | Exit\_Game.wav |
| 遊戲開始計數 | Game\_Start\_Count.wav |

# **三、 程式設計**

## 程式架構

## 在這次實習，我主要將類別分成俄羅絲方塊遊戲、按鈕、核心三大類。

一張含有 文字, 螢幕擷取畫面, 快閃記憶體 的圖片

自動產生的描述

## 程式類別

|  |  |  |  |
| --- | --- | --- | --- |
| 類別名稱 | 標頭檔（.h） | 原始碼（.cpp） | 說明 |
| TetrisGame | 104 | 464 | 遊戲架構 |
| Button | 41 | 44 | 按鈕 |
| Tromino | 51 | 159 | 玩家操縱的方塊 |
| 總行數 | 196 | 667 |  |

## 程式技術

在這次實作中，我們探索了許多重要的技術，這些技術為遊戲的開發帶來了巨大的幫助和順利進展，以下是我們所使用到的技術。

物件導向程式設計（Object-Oriented Programming）：我們使用物件導向的設計方式來組織和管理Tetris遊戲中的各個元素。透過建立不同的類別（例如方塊、遊戲場景、遊戲規則等），我們可以將相關的資料和行為封裝在一起，使程式碼更加模組化和易於維護。

遊戲邏輯和演算法：在Tetris遊戲中，我們需要實現方塊的下落、碰撞檢測、消除行等遊戲邏輯。我們使用的資料結構有optional(玩家操縱的方塊，預測方的方塊)、vector(遊戲畫布)、deque(存放下一個的方塊)、queue(存放七種不同的顏色)等資料構來處理這些邏輯，確保遊戲運行的正確性和效率性，值得提的一點，我們下一個方塊並不是下一個隨機，而是一次產生七種不同顏色的方塊(儲存在queue的資料結構)，這樣的話，可以避免我們一直遇到重複種類的方塊，使玩家感到遊玩困難。

# **四、 結語**

## 問題及解決方法

在這次實作中，我們碰到了一些困難，但我們也一一找到了解決方案。以下是我們所遇到的難題以及我們針對每個問題的因應之道。

第一個是素材取得的問題，由於網路上沒辦法直接拿到素材，導致我們

須一張一張的截，這花費我們很多時間，尤其一開始不規則狀的素材，我們

最終使用GIMP來解決圖片不規則去背的問題。

第二個是框架的使用，一開始我們想要一些參數想讓全部的檔案吃到，

但發現直接寫會直接crash，經過觀察後，我們把需要的參數寫成h檔，並用namespace裡宣告，然後再讓要使用的檔案include這個檔案。

第三個是遊戲的設計，我們那時候討論要直接用物體之間的碰撞來實作TetrisGame，還是要採用一個二維陣列方塊來時做畫布，也就是說畫布的每一格都是一個方塊，我們可以透過切換圖片，來讓畫布更新，最後是採用二維陣列方塊來實作我們的遊戲，因為我們發現到，如果我們採用前者的話，我們沒辦法做出旋轉。

## 時間表

在這一次開發總共耗費了許多時間，在刻畫遊戲架構的部分花上了比較多的時間，但還是有覺得充滿收穫，以下是我們的時間表。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 呂凱達 | | 陸嘉康 | |
| 週 | 時間 | 說明 | 時間 | 說明 |
| 1 | 介紹 | | | |
| 2 | 3小時 | 學習框架 | 2小時 | 學習C++、編譯器、框架 |
| 3 | 6小時 | 準備素材 | 22小時 | 以TypeScript 選寫遊戲邏輯原型 |
| 4 | 10小時 | 製作菜單 | 20小時 | 以TypeScript 選寫遊戲邏輯原型 |
| 5 | 13小時 | 製作第一關 | 22小時 | 轉換 (port） 遊戲邏輯至C++ |
| 6 | 5小時 | 製作第一關 | 20小時 | 轉換 (port） 遊戲邏輯至C++ |
| 7 | DEMO 1 | | | |
| 8 | 10小時 | 準備素材 | 10小時 | 優化遊戲邏輯，修正漏洞 |
| 9 | 8小時 | 新增特效 | 4小時 | 優化遊戲邏輯，修正漏洞 |
| 10 | 18小時 | 製作第二關 | 6小時 | 優化遊戲邏輯，修正漏洞 |
| 11 | 10小時 | 製作第三關 | 7小時 | 優化遊戲邏輯，修正漏洞 |
| 12 | DEMO 2 | | | |
| 13 | 12小時 | 程式碼重構 | 4小時 | 程式碼重構 |
| 14 | 20小時 | 製作第四關 | 5小時 | 程式碼重構 |
| 15 | 8小時 | 新增特效 | 4小時 | 程式碼重構 |
| 16 | 8小時 | 修正遊戲bug | 3小時 | 修正遊戲bug |
| 17 | DEMO 3 | | | |
| 18 | 端午節 | | | |
|  | 總計131小時 | | 總計129小時 | |

## 貢獻比例

組員一 呂凱達:60%

組員二 陸嘉康:40%

## 自我檢核表

|  |  |  |
| --- | --- | --- |
|  | 項目 | 完成 |
| 1 | 解決 Memory Leak | V |
| 2 | 自訂遊戲 Icon | V |
| 3 | 有 About 畫面 | V |
| 4 | 遊戲中說明如何操作 | V |
| 5 | 發佈(Rlease)版本 | V |
| 6 | 承上項，該版本可以正常操作 | V |
| 7 | 報告字形、點數、對齊、行距、頁碼等  格式正確 | V |

## 收穫

## 這學期的遊戲開發讓我學到了很多，我掌握了使用C++來開發遊戲的技巧。在開發過程中，我特別注重程式碼的整潔度，並尋找提升開發效率和遊戲效能的方法。我也遇到了一些挑戰和問題，我們努力解決了它們。這次的讓我更加熟練地運用C++來實現遊戲功能，並且對大型專案的開發有了更多的了解。我們期待在未來的遊戲開發中繼續學習和成長。

## 心得、感想

這一次開發遊戲，對於素材的準備感到心力交瘁，常常為了得到較好看的圖檔，剪了又剪、改了又改，才得到一張較能說服自己的圖，尤其不知道什麼原因，網路上png to bmp的功能，所產生的照片不能用，導致我們必須一張一張的使用小畫家來轉，進而加重我們獲取資源的困難。

我們在這次的開發更了解了何謂物件導向，更了解h檔、cpp檔之間的使用，期待未來我們能更加精進自己。

1. **對於本課程建議**

### 沒有任何問題，謝謝老師