內容

[一、簡介 2](#_Toc42611589)

[動機 3](#_Toc42611590)

[分工 4](#_Toc42611591)

[遊戲說明 5](#_Toc42611592)

[遊戲圖形 6](#_Toc42611593)

[遊戲音效 7](#_Toc42611594)

[程式設計 8](#_Toc42611595)

[程式架構 9](#_Toc42611596)

[程式類別 10](#_Toc42611597)

[程式技術 11](#_Toc42611598)

[結語 12](#_Toc42611599)

[間題及解決 13](#_Toc42611600)

[時間表(不含上課時間) 14](#_Toc42611601)

[貢獻比例 16](#_Toc42611602)

[自我檢核表 16](#_Toc42611603)

[收獲 17](#_Toc42611604)

[心得及感想 18](#_Toc42611605)

[附錄 19](#_Toc42611606)

# 一、簡介

## 動機

## 分工

吳軍樂：程式編寫、整體遊戲設計、程式架構設計、音效設計、素材準備

劉學逸：程式編寫、遊戲測試、程式漏洞修補、動畫設計、素材處理

## 遊戲說明

## 遊戲圖形

## 遊戲音效

|  |  |
| --- | --- |
| bgm1.mp3 | 主要遊戲背景音樂 |
| bgm2.mp3 | 打Boss時之遊戲背景音樂 |
| attack.mp3 | 攻擊怪物音效 |
| door.mp3 | 開啟黃、藍、紅、白門音效 |
| item.mp3 | 拾取物品或道具之音效 |
| gate.mp3 | 開啟鐵門之音效 |
| floor.mp3 | 走樓梯到下一層樓之音效 |
| tryOpen.mp3 | 嘗試開啟門，但失敗之音效 |

# 程式設計

## 程式架構

## 程式類別

## 程式技術

# 結語

## 間題及解決

1. 未使用及學習過JS
   * 在網路上自學JS的語法、架構和用法，再實作在專題裡
2. JS版本Gameframe的說明文件缺少
   * 一開始是參考範例程式，學習他的語法來實作。但到最後有些功能在範例程式中寫的方法有點亂或者沒有用到有關的功能，例如是BGM的宣告、動畫、按鈕功能等，所以直接看src裡的原始碼來自我理解和使用。
3. 遊戲人物和物品大小單位的定義
   * 遊戲的顯示像素為1920 x1080，而遊戲是以一格為單位，整個版面有26 x 13格。為了固定物件的大小，我們有定義一個(objectX, object)，這樣就能定義了物件的大小。所以我們整個遊戲都以「格」來做單位，而不是以像素來定義
4. JS版本的Gameframe裡缺少或者找不能相關的功能來做想做的功能
   * 例如我們的遊戲的人物能在23層地圖中自由穿梭，遊處會保存23層地圖的所有物件。在Gameframe裡有. goToLevel的功能，但這個功能會new出一個新的Level出來，而導致上一層的東西沒有保存，不符合我們的遊戲架構。所以我們使用在同一個level裡，以重新畫圖的方法來處理多層地圖的穿梭，把地圖裡的所有物件放一個陣列裡，然後可以隨時提取使用。但這個做法也引起了另一問題。
5. 程式面對記憶體不足
   * 上面提及說我們把23張地圖裡的所有物件存在一個陣列裡。一張地圖有26 x 13個物件，這樣導致我們的遊戲有嚴重的記憶體不足問題。所以我們選擇把物件以整數來編碼，然後一個陣列存了23張地圖的編碼，一個陣列是存當前層數的物件。這樣就可以大幅節省記憶體存儲空間。但這樣也再引發了另一問題。
6. JS版本記憶體釋放有缺陷，導致遊戲有機會記憶體溢出問題。
   * JS的記憶體釋放不同於C++。據這次實驗的經驗發現，在JS的環境下，是沒有一個明文的記憶體釋放的寫法。Google Chrome的環境下，當一個變數或物件，它的值是null或undefined時，在一段時間裡沒有用到當個變數或物件時，而且是在一個不規則的時間，不規定的情況下，Google Chrome才會把這個記憶體釋放出來。所以這樣JS的程式必然會有記憶體溢出問題。釋放記憶體的最有效方法是把Google Chrome給關掉，記憶體就會釋放。因此最佳的解決方法是做好記憶體管理，減少浪費使用不必要的記憶體，把程式優化是目前最佳的解決方法。
7. 記憶體管理
   * 有以上的經驗後，除了做了第5項，避免一次new太多的物件之外，我們的程式更要盡量減少new新的物件：第一、在每一次畫新地圖時，都會檢查有沒有相同的物件，從而重用上一層地圖共有的物件。第二、在必要時在物件上宣告其他物件都有的東西。例如地板、怪物都是一個物件，當打死怪物後顯示地板。如果用正常的物件時，當打死怪物後會把怪物的物件delete，再new地板物件。但在JS的話這樣new了一個新的物件，就會有記憶體不足或溢出問題。所以我們會打地板也宣告在怪物裡，這樣可以只改圖片，不改物件。然後等換地圖時，再統一考慮要不要把怪物的物件null掉，new成地板的物件。否則下一地圖也有用到同一個怪物的物件，就會重用物件，以記憶體舒緩記憶體的問題。

## 時間表(不含上課時間)

吳軍樂：

|  |  |
| --- | --- |
| 第零週 | 18小時 |
| 第一週 | 24小時 |
| 第二週 | 7小時 |
| 第三週 | 8小時 |
| 第四週 | 7小時 |
| 第五週 | 5小時 |
| 第六週 | 4小時 |
| 第七週 | 12小時 |
| 第八週 | 7小時 |
| 第九週 | 5小時 |
| 第十週 | 5小時 |
| 第十一週 | 5小時 |
| 第十二週 | 5小時 |
| 第十三週 |  |
| 第十四週 |  |
| 第十五週 |  |
| 總共 |  |

劉學逸：

|  |  |
| --- | --- |
| 第零週 | 7小時 |
| 第一週 | 4小時 |
| 第二週 | 6小時 |
| 第三週 | 5.5小時 |
| 第四週 | 8小時 |
| 第五週 | 3.5小時 |
| 第六週 | 4小時 |
| 第七週 | 7小時 |
| 第八週 | 4小時 |
| 第九週 | 6小時 |
| 第十週 | 4小時 |
| 第十一週 | 4小時 |
| 第十二週 | 5小時 |
| 第十三週 |  |
| 第十四週 |  |
| 第十五週 |  |
| 總共 |  |

## 貢獻比例

吳軍樂：50%

劉學逸：50%

共100%

## 自我檢核表

|  |  |  |  |
| --- | --- | --- | --- |
|  | 項目 | 完成否 | 無法完成的原因 |
| 1 | 解決Memory leak | ☑已完成 □未完成 |  |
| 2 | 自定遊戲 Icon | ☑已完成 □未完成 |  |
| 3 | 全螢幕啟動 | ☑已完成 □未完成 |  |
| 4 | 有About晝面 | ☑已完成 □未完成 |  |
| 5 | 初始畫面說明按鍵及滑鼠之用法與密技 | ☑已完成 □未完成 |  |
| 6 | 上傳setup / apk / source 檔 | ☑已完成 □未完成 |  |
| 7 | 報告字型、點數、對齊、行 距、頁碼等格式正確 | ☑已完成 □未完成 |  |
| 8 | 報告封面、側邊格式正確 | ☑已完成 □未完成 |  |
| 9 | 報告封面、側邊格式正確 | ☑已完成 □未完成 |  |
| 10 | 報告附錄程式格式正確 | ☑已完成 □未完成 |  |

## 收獲

## 心得及感想

附錄