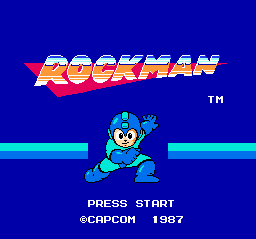
國立臺北科技大學

2023 Spring 資工系物件導向程式實習期末報告

Rockman



第 18 組

**目錄**

**一、 簡介**

[1. 動機 1](#_gjdgxs)

[**二、 遊戲介紹**](#_30j0zll)

1. [規則 1](#_規則)
2. [遊戲圖形 3](#_遊戲圖形)
3. [遊戲音效 11](#_遊戲音效)

[**三、 程式設計**](#_tyjcwt)

1. [程式架構 12](#_程式架構)
2. [程式類別 13](#_程式類別)
3. [程式技術 14](#_程式技術)

(一). 有大局觀的程式架構布局 14

(二). 人物與地圖的地形碰撞 14

(三). 第二關Boss的攻擊(fireman) 15

[**四、 結語**](#_2s8eyo1)

1. [問題及解決方法 16](#_問題及解決方法)
2. [時間表 17](#_時間表)
3. [貢獻比例 18](#_貢獻比例)
4. [自我檢核表 19](#_自我檢核表)

[5. 收穫 19](#_收穫)

1. [心得、感想 20](#_心得、感想)
2. [對於本課程的建議 20](#_7._對於本課程的建議)

# 一、 簡介

## 動機

Rockman為一款2D橫向捲軸遊戲，這是我從小第一次接觸紅白機模擬器，所玩的第一款遊戲，在當初我第一次接觸它時就被它的音效和動作流暢以及角色設定而深受吸引，就算在當時的我的不久後接觸到瑪莉歐，我依然認為這是我未來如果要復刻懷舊遊戲的首選。

當時的我也發現洛克人也有許多二創版本，而且我認為這遊戲雖然有名卻鮮少人知，所以覺得極有可能不撞題，如果不撞題就順便介紹這款遊戲給大家得知，然後這個是紅白機上首次登場的元祖洛克人1代，就是最一開始的版本，有經典的6大BOSS加上大BOSS威利博士。

然後課程一開始是我向林君曆介紹這款遊戲後，他也覺得看起來蠻有趣的，就決定了此遊戲作為復刻目標。

## 分組

110810006林君曆: 角色跑跳、關卡實作、所有敵人的實作、角色敵人互動

110AC1005陳文晟: 音效、角色爬梯、角色開火、跳躍優化、動畫實作、框架技術

# 二、 遊戲介紹

## 規則

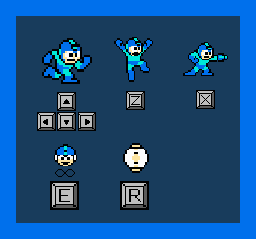
破關方式為打倒Boss，在一關當中主角有3條命，通關失敗可以選擇繼續挑戰同一關，或是回到主選單重新選關。

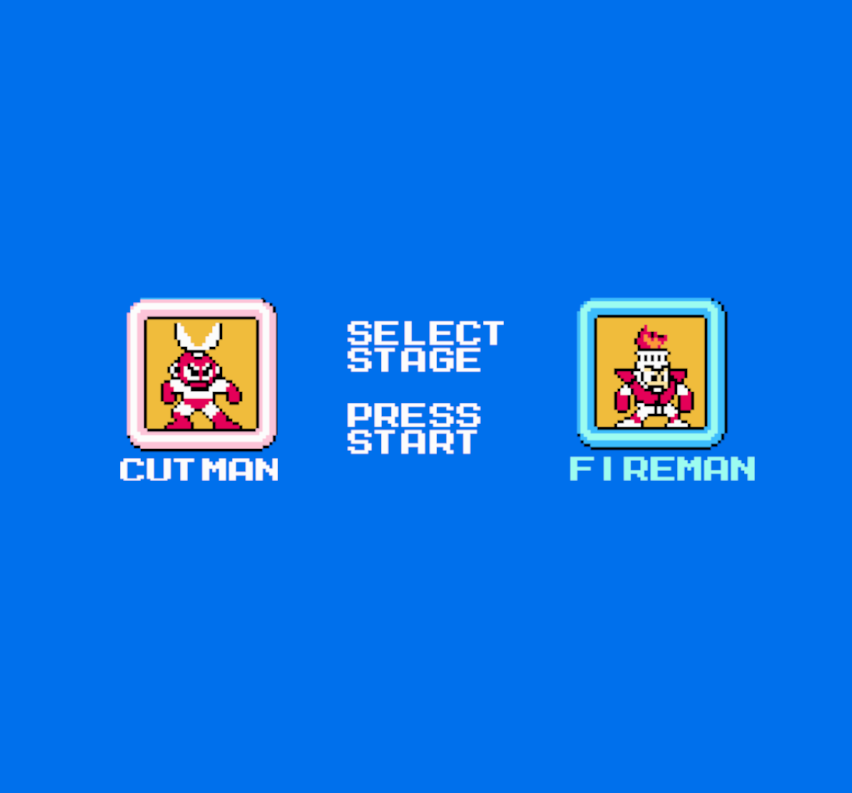
當進到一個關卡時，主角可以在空間中自由移動、攻擊敵人，掉出地圖外或是生命歸零就失去一條命，使用Z鍵來跳躍，使用X鍵發射子彈(max: 3)，方向左右鍵控制角色的左右移動，方向上下鍵爬梯子，在梯子上使用Z鍵來跳下梯子，按下H鍵可以開關顯示剩餘生命數。

遊戲分為兩關，分別為Cutman跟Fireman，關卡內有三個重生點，主角通過下一個重生點後，重生點會更新。

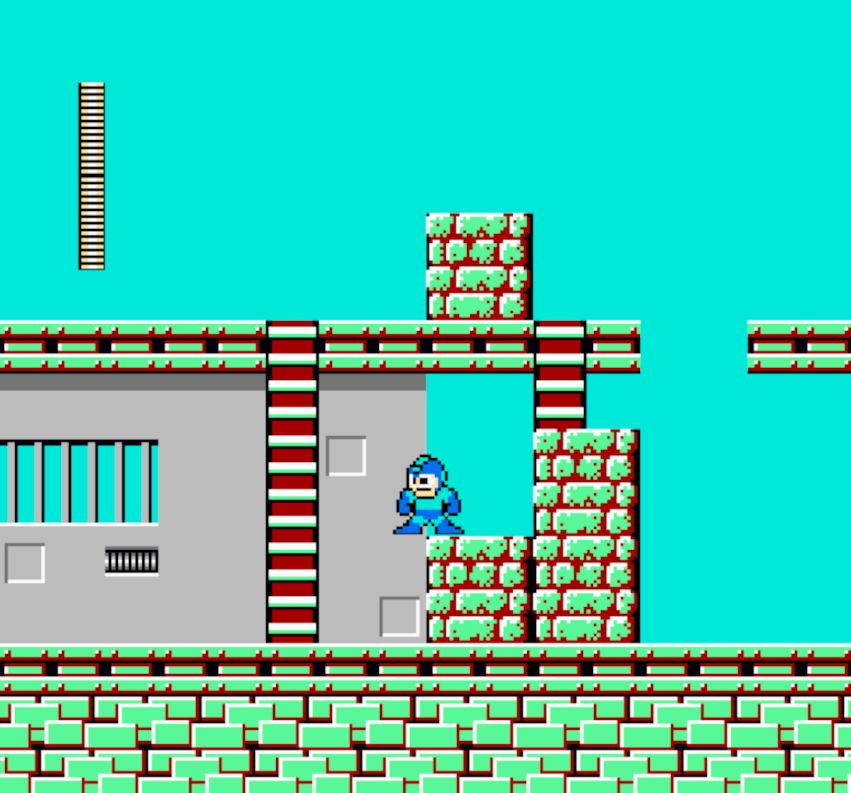
密技的部分，使用R鍵直接回滿血，使用E鍵來關閉/開啟敵人碰撞。

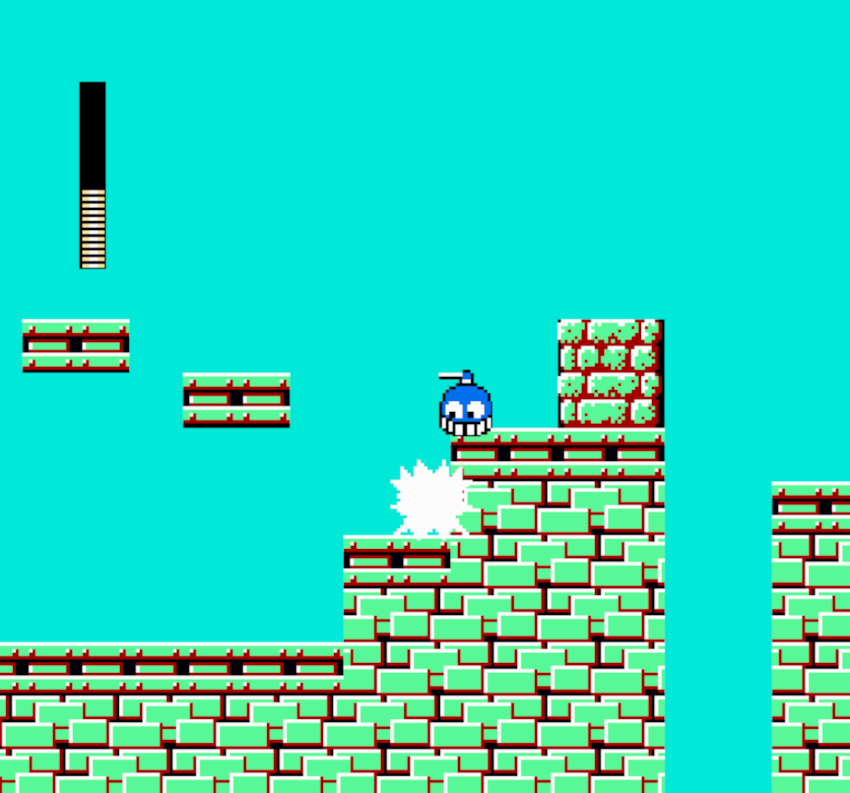
## 遊戲圖形

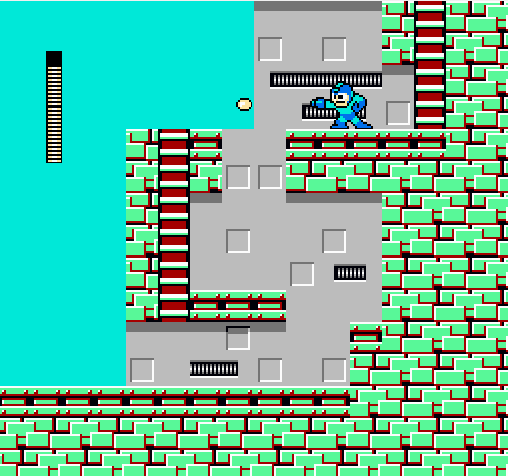
****

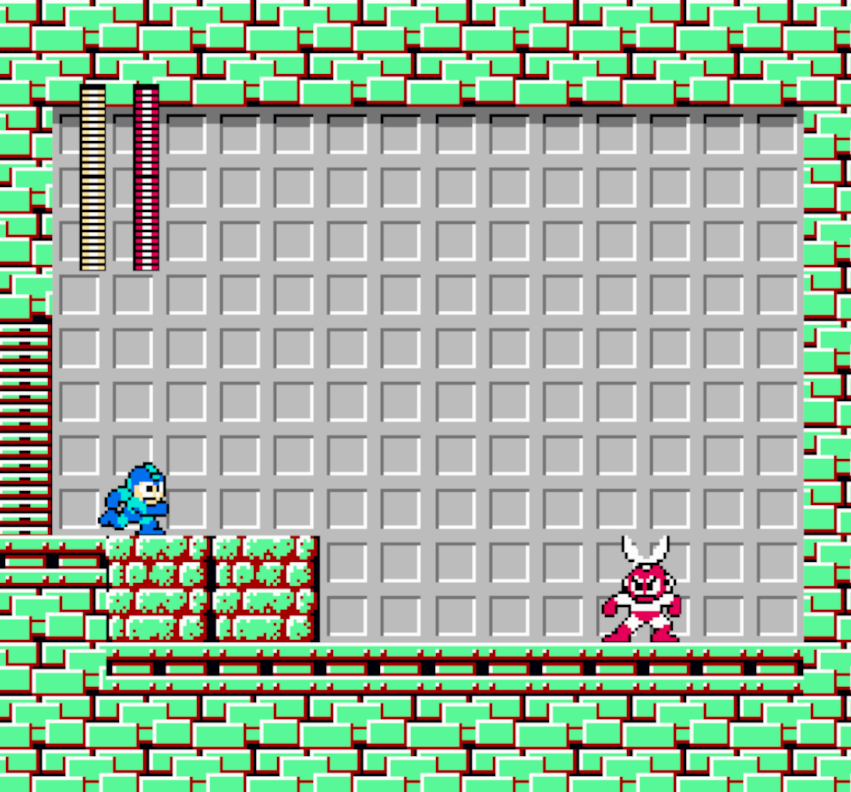
****

****

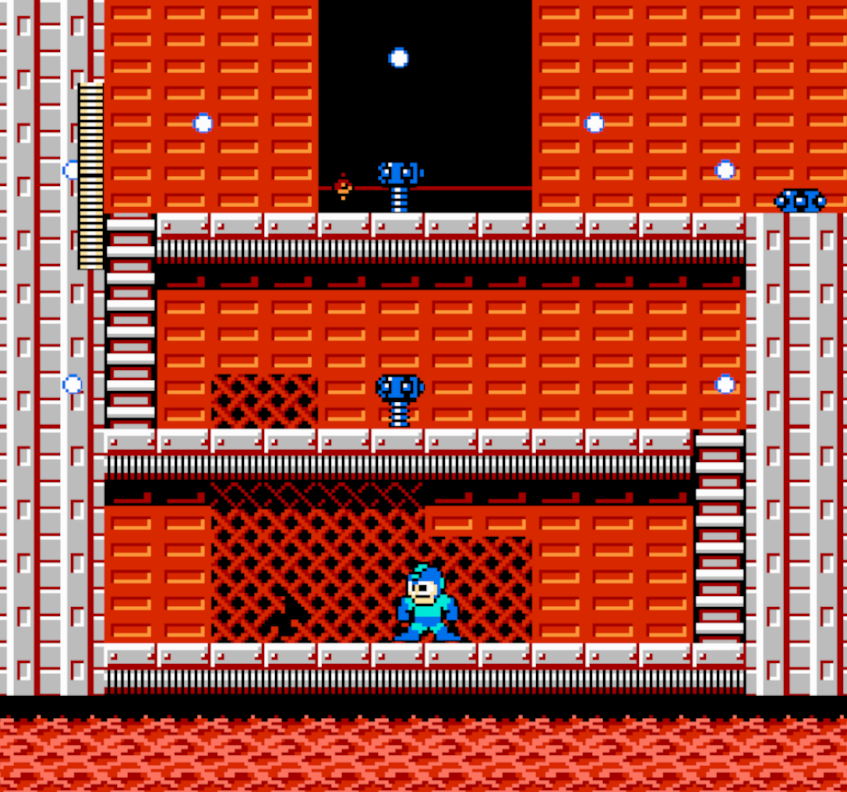
****

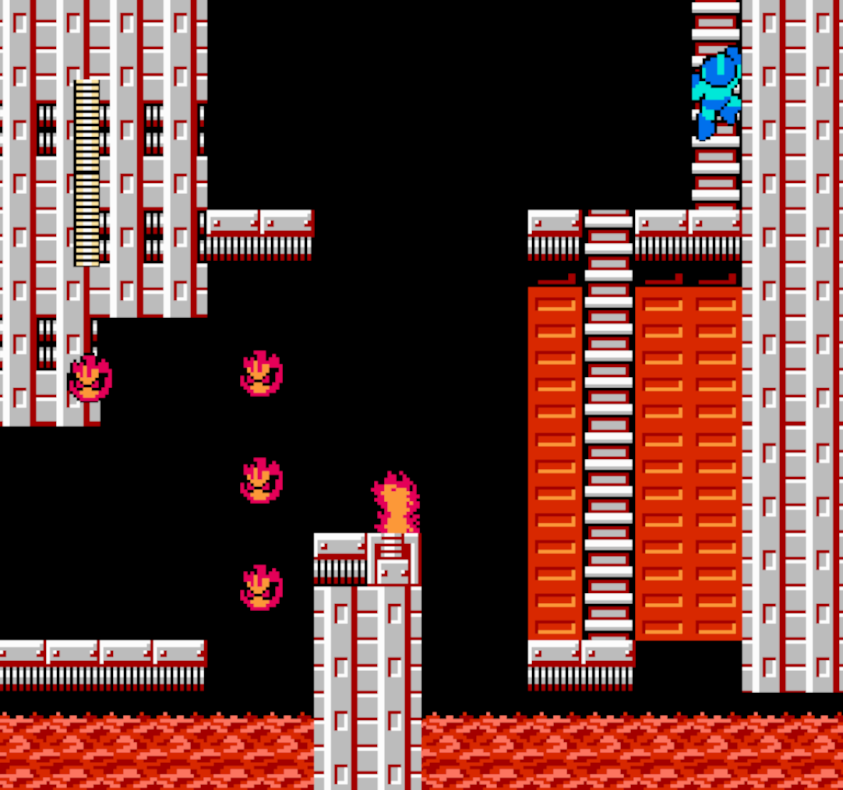


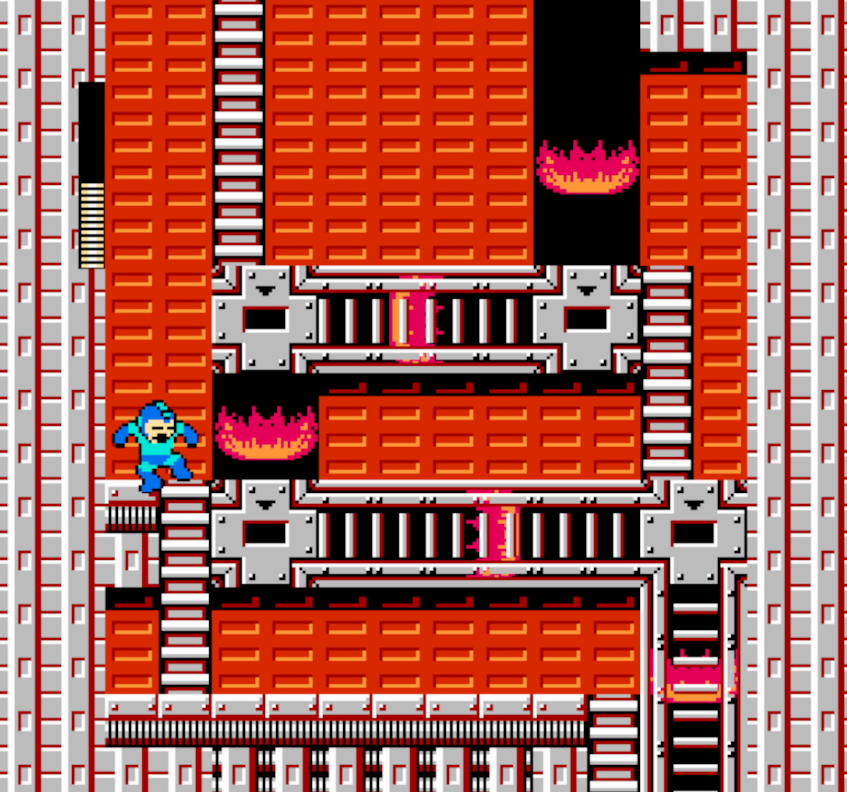
****

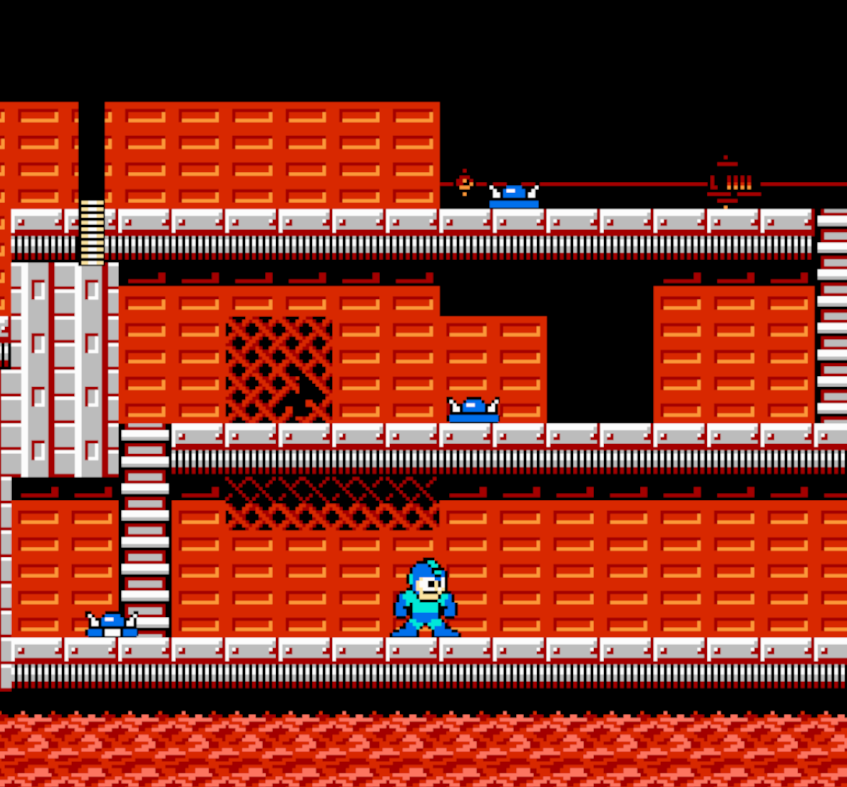
****

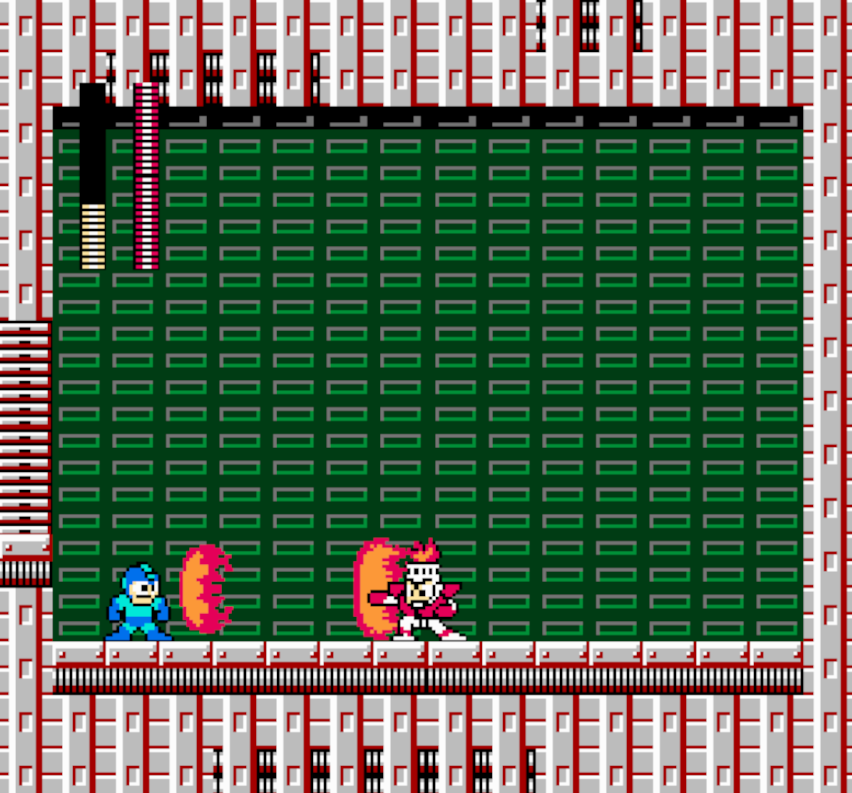
****

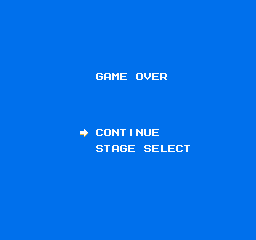
****

****

****

****

****

****

****

****

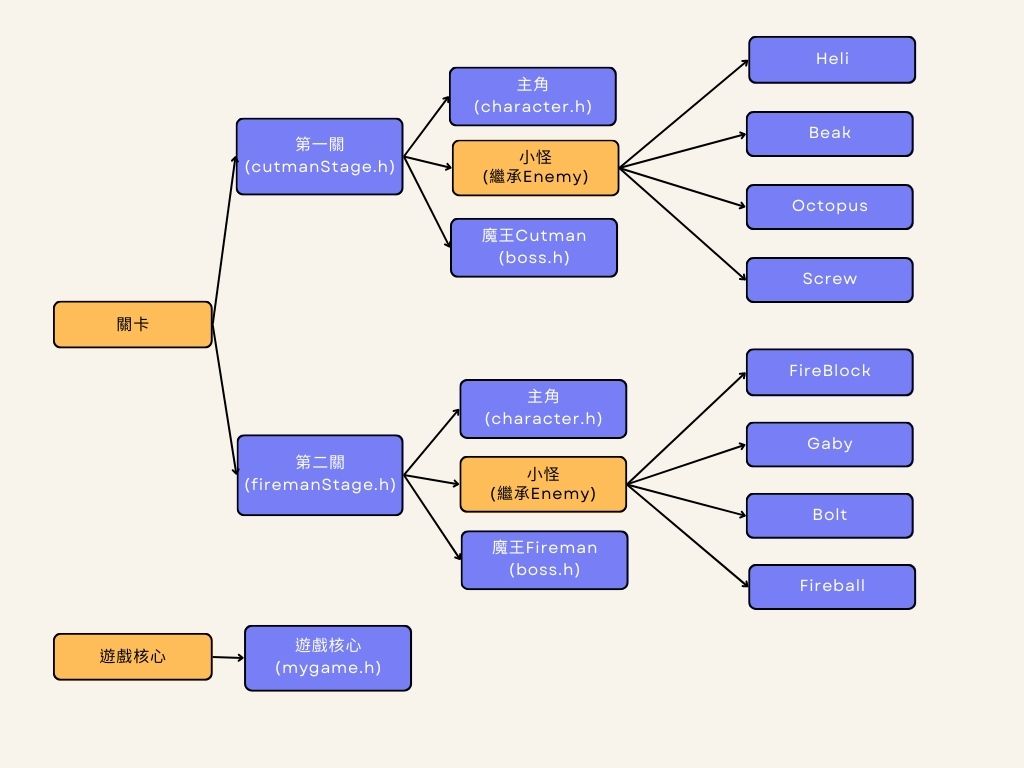
## 遊戲音效

以下陳列在遊戲中使用到的音效。

|  |  |  |
| --- | --- | --- |
| **音效類別** | | **音效檔案** |
| 主  角 | 攻擊 | MegaBuster.wav |
| 落地 | MegamanLand.wav |
| 被擊中 | MegamanDamage.wav |
| 死亡音效 | MegamanDefeat.wav |
| 小  怪 | 被擊中 | EnemyDamage.wav |
| 發射子彈 | EnemyShoot.wav |
| 第一關剪刀人 | 背景音樂 | CutManStage.wav |
| Boss攻擊音效 | RollingCutter.wav |
| Boss背景音樂 | BossBattle.wav |
| Boss血條音效 | EnergyFill3.wav |
| 第二關  火焰人 | 背景音樂 | FireManStage.wav |
| Boss攻擊音效 | FireStorm1.wav |
| Boss背景音樂 | 同上 |
| Boss血條音效 | EnergyFill3.wav |
| 選單 | 進入遊戲(選單) | GameStart.wav |
| 選單音樂 | MenuSelectTheme.wav |
| 選單選擇 | MenuSelect.wav |
| 片尾 | 片尾音樂 | EndingTheme.wav |

# 三、 程式設計

## 程式架構

在這次實習中，我們主要將類別分成遊戲核心跟關卡(stage)，關卡各自是一個世界，擁有自己的元素(主角、小怪、魔王、地圖等)，其中多個小怪的類別均繼承了Enemy的類別，每個小怪各自定義了獨有的攻擊移動邏輯。

## 程式類別

|  |  |  |
| --- | --- | --- |
| 類別名稱 | .h 行數 | 說明 |
| Character | 1427 | 實作玩家Rockman |
| CutmanStage | 608 | 實作第一關(處理所有關卡內的物件互動及呈現) |
| Cutman | 1301 | 實作第一關Boss |
| FiremanStage | 690 | 實作第二關(處理所有關卡內的物件互動及呈現) |
| Fireman | 526 | 實作第二關Boss |
| Enemy | 37 | 定義敵人的介面，為小怪的base類別 |
| Heli | 280 | 實作敵人一 |
| Beak | 344 | 實作敵人二 |
| Octopus | 246 | 實作敵人三 |
| Screw | 537 | 實作敵人四 |
| FireBlock | 157 | 實作敵人五 |
| Gaby | 191 | 實作敵人六 |
| Bolt | 126 | 實作敵人七 |
| Fireball | 136 | 實作敵人八 |

## 程式技術

在這次實習課中學習與研究到了許多技術，讓遊戲的開發更加順利，以下為幾個本次開發重要的技術。

(一). 有大局觀的程式架構布局

在剛開始開發前，我們討論出開發應該要達到的事情，1.遊戲核心中的程式碼要簡潔 2.如何有效率的協作開發，因此在開始撰寫程式碼前，我們將整個有遊戲該有的架構先想清楚，mygame中只需要判斷現在玩家是在玩哪個關卡，就呼叫該關卡的物件函式，一切的定義都交由關卡物件，而每個關卡又各自擁有敵人跟主角物件，就這樣一層層下去，可以確保在每個物件只需要處理自己的事就好，也可以很好的安排組員間的負責部分。在完成遊戲架構後，我們開始研究整個框架的檔案建置順序以做到標頭檔獨立，讓組員可以各自開發不同的.h。

(二).人物與地圖的地形碰撞

在遊戲中，地圖是由32\*32的不同方塊組成的， 我們將大張地圖中的每個方塊的ID建立成一個二維的vector，透過vector的讀取判斷哪個ID的方塊主角能有的行為(如跳躍、橫移等)，而地圖中元素的座標都是以地圖左上角為0的絕對座標做設定，並且減去關卡中Camera的座標，以相對位置設定TopLeft來呈現在畫面上。Camera在遊戲中扮演很重要的角色，根據現在主角的位置去做追蹤，並且在特定的位置觸發轉場，我們實作的方法為將Camera與主角的(y座標%512)去判斷上下轉場，橫向的則是碰到閘門後會進行轉場。

註:512為遊戲畫面的長寬

(三). 第二關Boss的攻擊(fireman)

因為Boss的攻擊是隨著主角的移動去判斷，所以在不確定Boss攻擊武器的數量下，利用先射出的攻擊會先消失這點，使用queue來實作，每當Boss攻擊，就push一個武器的CMovingBitmap進入queue，當queue最前端的武器超出畫面就會被pop掉，比起使用vector更加地貼近我們的實作需求。

# 四、 結語

## 問題及解決方法

在這次實作中遇到了很多問題與困難，最後都有成功解決。

第一個是建地圖的工作量問題，我們的地圖是由32\*32的方塊組成，因為我們的第一張地圖方格元素比較單純好辨識，因此在開發初期花費了3、4個小時，人工就能建好地圖檔案。不過在第二關的時候，我們發現花在建地圖上的時間可能會增多，整張圖會有21504個方塊，重複性的動作應該要避免，於是我們使用python的Pillow跟numpy函式庫來建立地圖檔，具體一點的作法為，讀取多個32\*32的地圖方格元素，將大張的地圖針對每個32\*32的方格與所有方格元素做比對，如果RGB的數值皆相同，就將該編號存入陣列當中，只需要在切出小格的元素，不到30分鐘就能建好地圖，省了不少開發時間。

第二個是主角與敵人碰撞判斷上的誤差，由於遊戲圖片資源的倍率較小，我們在遊戲中皆放大兩倍作呈現，程式碼也都以兩倍的長度去判斷，但框架提供的isOverlap函式卻是用原倍率的長寬去作判斷，導致會有誤差，在我們研究框架所提供的函式後，我們將其修改優化後，原本的問題就迎刃而解了。

第三個問題是在撰寫洛克人片尾名單時，背景幀數每次都不一樣，最後是直接把背景的秒數用文字顯示出來但放在畫面上看不到的地方就正常了，可能是有文字顯示和沒有文字顯示的效能不同導致的，所以我讓某個文字在畫面上固定顯示來確保我每次運行的動畫的時機點都是一致的。

## 時間表

在這一次開發總共耗費了許多時間，我們力求將遊戲最大程度的還原，花了許多時間在雕琢細節，以下是我們的時間表。

|  |  |  |  |
| --- | --- | --- | --- |
| **周** | 林君曆(小時) | 陳文晟(小時) | 說明 |
| 1 | 介紹 | 介紹 | 介紹 |
| 2 | 0 | 0 | 框架練習 |
| 3 | 3 | 3 | 尋找遊戲素材 |
| 4 | 2 | 1 | 實作關卡物件 |
| 5 | 13 | 0 | 建地圖、主角左右移動 |
| 5 | 0 | 48 | 框架技術、標頭檔分工實現 |
| 6 | 9 | 12 | 主角跳躍 |
| 7 | 0 | 16 | 主角爬梯 |
| 7 | 9 | 5 | 重建地圖、轉場 |
| 8 | 29 | 24 | DEMO1  轉場、敵人移動 |
| 9 | 11 | 0 | 敵人移動、敵人介面 |
| 9 | 0 | 16 | 主角開火 |
| 10 | 10 | 0 | 處理素材 |
| 11 | 15 | 0 | Boss1移動 |
| 12 | 13 | 0 | Boss1介面 |
| 13 | 21 | 12 | DEMO2  主角敵人碰撞、音效 |
| 14 | 11 | 0 | 死亡動畫 |
| 15 | 14 | 0 | 第二關、更多怪物 |
| 16 | 22 | 0 | 更多怪物、完成boss2 |
| 17 | 6 | 4 | 製作報告 |
| 18 | 7 | 15 | DEMO3  片尾動畫、選角動畫、音效補完、操作指示、生命數UI |
| 總計 | 195小時 | 156小時 |  |

## 貢獻比例

110810006林君曆55% 、110AC1005陳文晟45%

## 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(Release)版本 | V |
| 6 | 承上項， 該版本可以正常運作 | V |
| 7 | 報告字形、點數、對齊、行距、頁碼等  格式正確 | V |

## 收穫

110810006林君曆

這學期讓我們學到如何使用 C++ 來開發遊戲，在開發大型專案時也注意到需要先有完整的架構，後續的開發才會順利，同時也要注意自己程式碼的整潔度，並且利用不同工具或資料結構，讓整個開發過程更有效率，並且更加懂得如何跟組員搭配，對於git版本控制也有更進一步的認識，最後 也讓自己的作品集多了一個遊戲專案，收益良多。

110AC1005陳文晟

這學期的學習經驗為我帶來了新的視野，我得以掌握C++在遊戲開發上的實踐方法。我明白到，面對大型專案時，首要之事是要設定明確且全面的結構，這樣在後續的開發過程中才能如魚得水。我也更專注於保持程式碼的潔淨，並且活用各種工具及資料結構來提高開發的效率。

這次經驗讓我深入地理解了如何與團隊成員有效地合作，進一步體認到git版本控制的重要性。在這次的學習過程中，我獲益匪淺，並且成功地在我的作品集中新增了一項遊戲專案。此外，我的Unity經驗也在此次專案中發揮了不少的作用。由於我對Unity有著豐富的經驗，這使得我能在遊戲邏輯的設計上提供組員實質的幫助。我的這段經驗不僅提升了我們的團隊效率，更提高了我們的遊戲品質。

## 心得、感想

110810006林君曆

在這次的遊戲開發中，讓我成長了不少，從0到最後生出一個完整的遊戲，很有成就感，也確實應用了上學期OOP的上課所學，讓開發上非常順利，不過因為這學期的課太滿了，沒辦法挪出更多的時間來開發遊戲，是稍微可惜的地方，不過我還是對於我們學期尾聲完成的這個作品感到相當滿意!

110AC1005陳文晟

這次的遊戲開發讓我又重新接觸了一次遊戲製作，我又對純物件導向引擎的開發有了對應的經驗。我實際運用到了上學期學習的物件導向程式設計(OOP)知識，使得開發流程進行得蠻順暢。

然而，這學期的課業繁重，時間分配上略顯吃力，導致我無法投入更多的時間於遊戲開發上，這是我稍感遺憾的部分。但縱然如此，當我看到學期結束時我們所完成的遊戲作品，我仍然感到非常的欣慰和成就感。

## 7. 對於本課程的建議

沒有，謝謝老師。