|  |
| --- |
| 國立臺北科技大學  2023 Spring物件導向程式實習  專題報告 |
| Crypt of the NecroDancer |
| C:\Users\110590021\AppData\Local\Temp\98\vmware-110590021\VMwareDnD\horizon_13948-4babc8f7\mainmenu.png |
| 第四組 |

# 目錄

[目錄 1](#_Toc137767998)

[一、 簡介 2](#_Toc137767999)

[一、 動機 2](#_Toc137768000)

[二、 分工 3](#_Toc137768001)

[二、 遊戲介紹 4](#_Toc137768002)

[1. 規則 4](#_Toc137768003)

[2. 遊戲圖形 4](#_Toc137768004)

[3. 遊戲音效 6](#_Toc137768005)

[三、 遊戲設計 7](#_Toc137768006)

[1. 程式架構 7](#_Toc137768007)

[2. 程式類別 9](#_Toc137768008)

[3. 程式技術 9](#_Toc137768009)

[ Camera 9](#_Toc137768010)

[ Game\_map 10](#_Toc137768011)

[ Interface 10](#_Toc137768012)

[ Monster 10](#_Toc137768013)

[ Player 11](#_Toc137768014)

[ Tempo 11](#_Toc137768015)

[四、 結語 12](#_Toc137768016)

[1. 問題及解決方法 12](#_Toc137768017)

[2. 時間表 12](#_Toc137768018)

[3. 貢獻比例 13](#_Toc137768019)

[4. 自我檢核 13](#_Toc137768020)

[5. 收穫 14](#_Toc137768021)

[6. 心得或感想 14](#_Toc137768022)

[7. 想說的話 15](#_Toc137768023)

# 簡介

## 動機

會讓節奏地牢這款遊戲如此吸引我們的原因，除了在動作闖關中加入了音樂節拍來讓戰鬥更增添如同回合制的策略感以及音樂的戲的打擊感，更有大量有趣的角色扮演元素。

包含了豐富的技能招式、武器、道具，地圖場景上的陷阱、路線設計也很精巧。還可以挖掘牆壁打通密道發現隱藏的房間！

它本身就是一款主打高難度的類Rogue遊戲，迷宮地圖都是隨機生成的，每一輪會獲得的道具、敵人會出現的位置都無法預測，玩家們一開始會很容易就被怪物給打死又重頭來過，可是您在遊戲中收集到的鑽石，都可以用來永久強化自身的能力，所以在不斷重來的過程中，玩家的技術會變好，人物角色逐漸變強。

## 分工

張匯吾:主要的圖片以及音樂音效處裡，介面產生和節奏邏輯的實現。

林兆玄:地圖建構以及主要的畫面渲染邏輯，怪物和玩家的行為。

# 遊戲介紹

## 規則

遊戲主要是RPG類型，主要為打怪過關的方式，打倒最後一關BOSS之後就結束遊戲。當進入到一個關卡時，必須隨著節奏按下方向鍵，否則角色將無法移動，腳色可以攻擊怪物、挖牆壁、換武器等等。遊戲主要分成三關，其中有寶箱隱藏在密室裡面，第一關為基礎關，第二關則有進階的武器和怪物，第三關即為BOSS關，討伐完BOSS遊戲就結束。

## 遊戲圖形

|  |
| --- |
| https://cdn.discordapp.com/attachments/1091238116209197127/1118851872170717184/image.png |
| https://cdn.discordapp.com/attachments/1091238116209197127/1118852096012324874/image.png |
| https://cdn.discordapp.com/attachments/1091238116209197127/1118852400682385489/image.png |
| https://cdn.discordapp.com/attachments/1091238116209197127/1118852650222493726/image.png |

## 遊戲音效

|  |  |
| --- | --- |
| 音效類別 | 音效檔案 |
| 背景音樂 | Zone1\_1.wav |
| 挖土 | dig\_dirt.wav |
| 挖基岩 | Dig\_fail.wav |
| 玩家砍擊 | hit.wav |
| 玩家受傷 | hurt.wav |
| 開門 | door\_open.wav |
| 開寶箱 | chest\_open.wav |

# 遊戲設計

## 程式架構

在這次實習，我們主要將類別分成鏡頭、地圖、角色(怪物、主角)、節奏和介面四大類，每個類別都各司其職扮演重要角色。

|  |
| --- |
| https://cdn.discordapp.com/attachments/911613080247623701/1118909369548021770/fb326c172ff55653.jpg |
| https://cdn.discordapp.com/attachments/911613080247623701/1118914775733260398/f171c87b1329dc43.jpg |

## 程式類別

|  |  |  |  |
| --- | --- | --- | --- |
| 類別名稱 | 標頭檔(.h) | 原始碼(.cpp) | 說明 |
| camera | 20 | 160 | 實作畫面渲染 |
| game\_map | 41 | 119 | 存放地圖資訊 |
| Img\_path | 361 | 0 | 圖片路徑存放 |
| interface | 26 | 107 | 實作遊戲介面 |
| monster | 192 | 689 | 實作怪物行為 |
| player | 65 | 225 | 實作玩家行為 |
| tempo | 21 | 92 | 實作節奏邏輯 |
| mygame | 126 | 0 | 定義物件 |
| mygame\_initialize | 0 | 56 | 實作遊戲開始畫面 |
| mygame\_over | 0 | 52 | 實作遊戲結束畫面 |
| mygame\_run | 0 | 329 | 實作遊戲基礎運動 |

## 程式技術

### Camera

人物視角會在這邊渲染，其原理是player必定生成在畫面中間，所以在game\_map中抓取角色與主角的相對位置就可以顯示在畫面上。在渲染的過程中，會先將地板和牆壁渲染完畢，之後在將腳色及物件覆蓋上去。在渲染完地圖及腳色後，將會判定是否要顯示攻擊、挖掘動畫以及怪物的血量。除此之外我們還引入了計算區分陰影直線的演算法(bresenham's line algorithm)，讓玩家看不見障礙物後的方格。

### Game\_map

地圖為24\*21的矩陣，其中包含了所有的方塊的ID並且使用bin檔來儲存地圖資訊。

### Interface

介面包含了武器圖示，鏟子圖示以及玩家血量。

### Monster

儲存怪物的圖片以及實作怪物與玩家的互動。當玩家在怪物將要移動的位置時，將會由攻擊取代移動。

#### 史萊姆

每兩拍節奏移動一次並且只會左右移動。

#### 蝙蝠

每兩拍節奏向玩家移動一次。

#### 牛頭怪

每一拍節奏都會向玩家移動。當與玩家呈同一直線時，將會停滯一拍並進入衝撞模式。在衝撞模式中，牛頭怪只會向同個方向移動，如果撞擊到玩家將會造成大量傷害，而撞到牆壁時，將牆壁挖空並且進入暈眩狀態。

#### Deep Blue 和小怪(BOSS)

以西洋棋設計的Boss，Boss與小怪們的移動方式與對應的兵種相同。Boss在小怪殺完前都會逃離玩家，直到小怪們全部死亡他將會著朝著玩家移動。

### Player

在此寫入玩家的行為，包含攻擊、移動、挖掘以及動畫。

### Tempo

實作節奏的判斷以及動畫。我們計算整首曲子的時間，去控制音符移動到心臟的速率，再藉由音符的位置去定義按鍵按下之後能操作的區間，以此來實現按照節奏移動的玩法。

# 結語

## 問題及解決方法

在原作中，地圖的渲染是預先生成好的，人物在移動的時候僅僅是移動視角的位置。而這個framework是以bitmap為主純平面的渲染畫面且不能渲染畫面外的物件。所以我們將物件ID儲存在二維陣列裡，利用主角一定在中間的特性，將實際位置轉成在畫面中的位置，以此來實時渲染畫面。

在計算節奏的時間時，我們參考了文件上show和move迴圈執行的速度，發現可以改config去控制迴圈執行的速率，可是經過測試之後我們發現那個速率並不準確，沒辦法準確控制迴圈的速率這對於節奏遊戲是一個硬傷，後來我們也只能使用體感去改參數來盡量完成。除此之外的都算是小問題。

## 時間表

|  |  |  |  |
| --- | --- | --- | --- |
| 週次 | 張匯吾 | 林兆玄 | 說明 |
| 1 | 4小時 | 4小時 | 熟悉framework |
| 2 | 4小時 | 4小時 | 熟悉framework |
| 3 | 12小時 | 4小時 | 處理素材 |
| 4 | 4小時 | 16小時 | 嘗試在畫面上生成地圖 |
| 5 | 6小時 | 10小時 | 開始製作怪物factory |
| 6 | 10小時 | 16小時 | Camera渲染邏輯 |
| 7 | 16小時 | 6小時 | 製作角色動畫 |
| 8 | DEMO1 | DEMO1 | DEMO1 |
| 9 | 16小時 | 6小時 | 節奏邏輯實作 |
| 10 | 16小時 | 6小時 | 處理素材 |
| 11 | 10小時 | 10小時 | 物件之間的互動 |
| 12 | DEMO2 | DEMO2 | DEMO2 |
| 13 | 4小時 | 16小時 | 加入視線演算法 |
| 14 | 6小時 | 6小時 | 修BUG |
| 15 | 16小時 | 16小時 | 音樂與節奏的同步 |
| 16 | 20小時 | 30小時 | 導入音效和修BUG |
| 17 | DEMO2 | DEMO2 | DEMO2 |
| Total Time | 144 | 150 |  |

## 貢獻比例

張匯吾:50%

林兆玄:50%

## 自我檢核

|  |  |  |  |
| --- | --- | --- | --- |
|  | 項目 | 是否完成 | 為何無法完成 |
| 1 | 解決Memory Leak的問題 | 否 | 在很後期才想到這個問題，想要處裡已經來不及了。 |
| 2 | 自訂遊戲Icon | 是 |  |
| 3 | About 畫面 | 是 |  |
| 4 | 遊戲中說明如何運作 | 否 | 遊戲本體並沒有教學 |
| 5 | 發布(Release)版本 | 是 |  |
| 6 | 承上，該版本可正常運作 | 是 |  |
| 7 | 報告字型、點數、對齊、行距、頁碼等格式正確 | 是 |  |

## 收穫

這學期做遊戲，整體來說還是蠻有趣的，更重要的是我們學到很多東西，包括更熟悉C++，也更實際應用了物件導向的概念，有種自己已經是一個開發者的感覺，另外我們也學到時間分配的重要性，一定要訂好進度，不然就會懶惰到期末才在趕feature。還有我們還更熟悉了Github的專案模式，兩個人一起做專案真的學到很多，規劃好分工的話真的可以事半功倍。

## 心得或感想

我覺得實際做遊戲真的跟在玩的時候完全不一樣，我要去思考很多東西，遊戲的某些運作邏輯需要靠一些演算法，實際去了解之後我們學到很多東西的，而自己在做專案的同時，也會注意到其他組別也會遇到一些問題，我都會很好奇地去瞭解並嘗試幫忙，從而學到一些東西，也擴展了自己的思考模式。其他組別的遊戲或許我們不會做，但也可以嘗試去想看看，這邊的邏輯是什麼，如果是我們要怎麼做，這次OOP用做遊戲的方式加強自己的程式能力，真的讓我們受益良多。

## 想說的話

助教很用心，不過可能是評量很多組的關係或助教對遊戲不熟悉，所以使用關卡數來評量遊戲的完成度，不過我們覺得每個遊戲本來就不一樣，用關卡衡量可能會讓有些組別照著關卡做，所以就覺得蠻可惜的。還有希望之後可以增加timer讓相關功能可以更方便的製作出來。