

國立臺北科技大學

2024 資工系物件導向程式設計實習

期末報告

魔塔



第 13 組

黃漢軒

目錄

一、	專案簡介	1
1.	遊戲簡介.....	1
二、	遊戲介紹	1
1.	遊戲規則.....	1
2.	遊戲畫面.....	3
三、	程式設計	6
1.	程式架構.....	6
2.	程式技術.....	6
四、	結語	10
1.	問題及解決方法.....	10
2.	貢獻比例.....	10
3.	自我檢核表.....	11
4.	收穫.....	11
5.	心得、感想.....	11
6.	對於本課程的建議.....	11

一、 專案簡介

1. 遊戲簡介

魔塔是我小時候玩的 RPG 遊戲，簡單的畫風與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，在這一次的實習中選的是魔塔的完整版本，也就是地上 20 層魔塔與地下 25 層魔塔，以及 10 關隱藏關卡的遊戲，在設計時也玩了這款遊戲非常多次，為了保留最完整的劇情，所以盡可能的完美復刻這個遊戲。

除了遊戲的細膩劇情，也含有豐富的支線，供玩家進行解謎，甚至還有三種不同的隱藏劇情（Bad End、Normal End、True End），讓玩家有一玩再玩的想法。

2. 組別分工

本次專案是我自己一個人獨自完成，故沒有分組分配進度的部分。

二、 遊戲介紹

1. 遊戲規則

遊戲主要是 RPG 遊戲走向，主要為打怪過關的形式，只要長頸鹿的血量不要歸零，遊戲就會繼續下去，打到最後一關遇到公主就會結束遊戲。當進到一個關卡時，長頸鹿可以開門、拿東西、跟 NPC 聊天對話、打怪 或者上樓，可以使用特殊按鍵來使用功能（例如任意門、心鏡等等），使用 L 鍵來查詢功能列表，在獲得特殊物件後可以使用哪個按鍵來操作。

遊戲主要分成上層 20 關、下層 25 關、以及密室 10 關，密室的觸發條件 需要經由 NPC 的對話得知，或者觀察地圖上不尋常的場景（例如被牆壁包

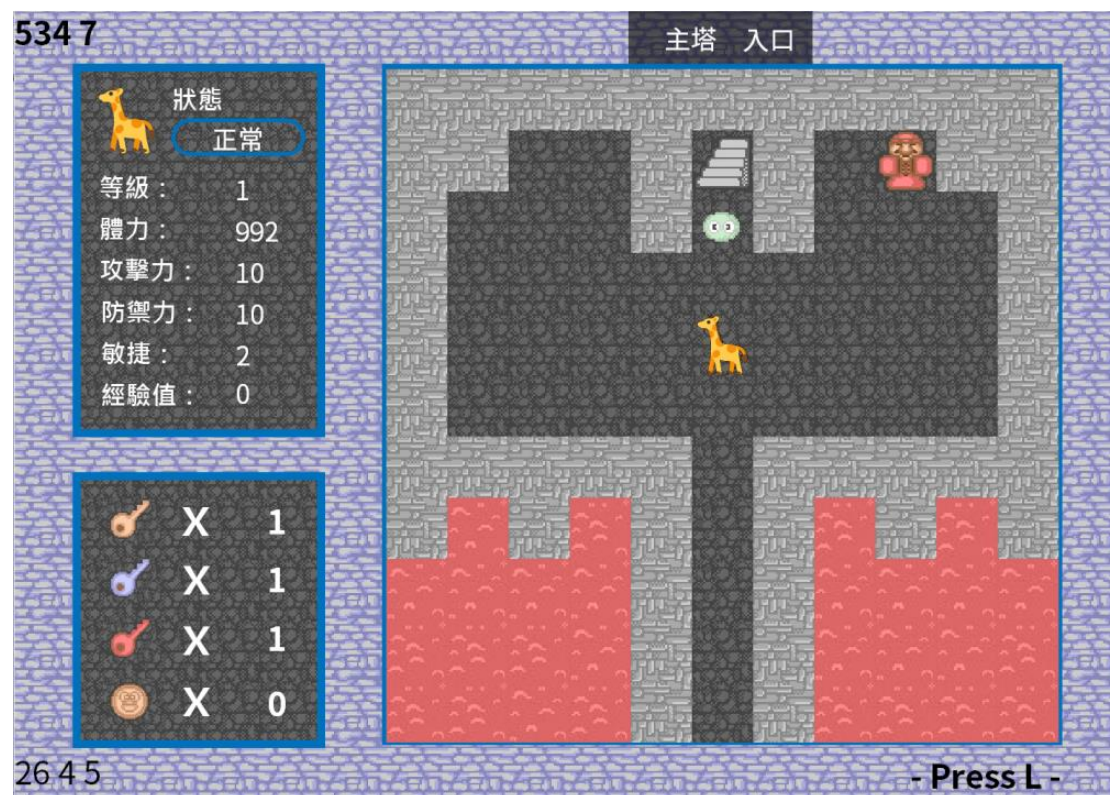
裏的物件，需使用任意門來打開牆壁）進入密室，會得到意想不到的關卡或寶藏。

密室其中一關會是觸發 True End 的關鍵，藉由 NPC 的對話可以促使玩家察覺到關卡不一樣的地方，利用這個關鍵，進而觸發進入 True End 關卡的隱藏條件。

密技的部分，請使用 menu 上的 debug 模式，裡面有許多密技可以使用。

2. 遊戲畫面

最多五張。



1114 801

主塔 入口


 狀態
 正常

等級： 1
 體力： 998244353
 攻擊力： 998244353
 防禦力： 998244353
 敏捷： 998244353
 經驗值： 998244353


 名稱 綠色史萊姆
 特殊技能 無

體力 40
 攻擊力 18
 防禦力 1
 敏捷 0
 攻擊次數 1
 Exp 1
 Gold 1

 X 23767
 X 23767
 X 23767
 X 998244353


- Quit(D) -

26 9 5

- Press L -

1245 208


主塔 2F





 狀態
 正常

等級： 1
 體力： 998244353
 攻擊力： 998244353
 防禦力： 998244353
 敏捷： 998244353
 經驗值： 998244361



1 F



 X 23767
 X 23767
 X 23767
 X 998244368

- Quit(F) -

- Enter -

28 5 7

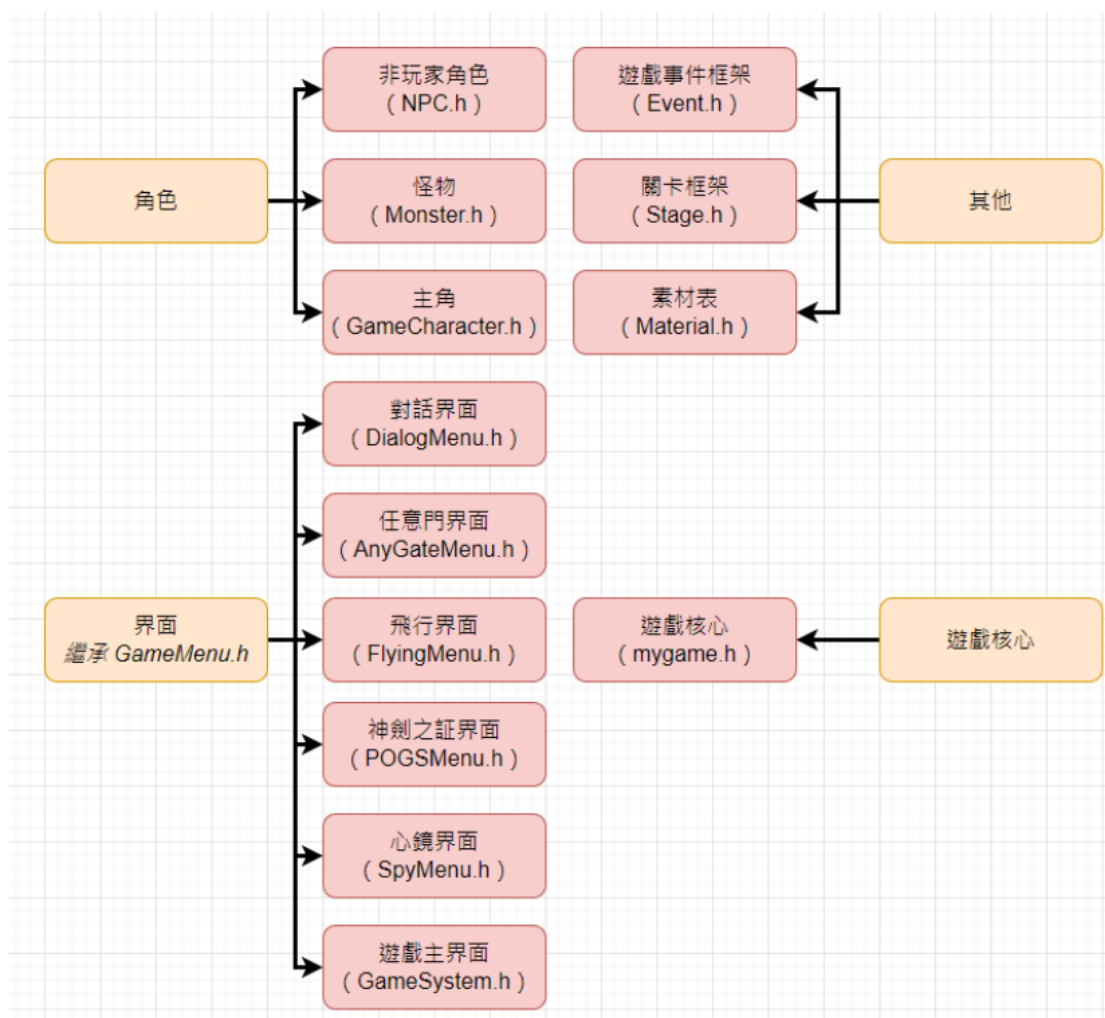
- Press L -



三、 程式設計

1. 程式架構

在這次實習，我主要將類別分成角色、介面、其他、遊戲核心四大類，其中介面的部分繼承了 GameMenu.h 來實作介面。



2. 程式技術

在這次實作中學習與研究到了許多技術，讓這個遊戲的開發更加順利，以下將會帶來三個比較重要的技術。

A. 遊戲物件註冊表

在做專題分析時，發現到遊戲的物件主要可以分成六類，分別是主角、互

動場景、互動物品、非玩家角色、怪物與單純場景物件。除了主角只有一個，用這樣的方式分類所有的物件會顯得非常有效率，**互動場景通常會有許多個動畫組成，互動物品、單純場景物件沒有動畫，只有一個靜態的圖片，而非玩家角色與怪物都由兩個不同的圖所組成的動畫。**

考慮到這樣的分割方式，我做了一個遊戲註冊表，分成五大類，流程如下：

1. 第一個數字 a 讀入有幾個互動場景。
2. 接下來有 a 行，每一行讀入一個字串與兩個數字，字串為該互動場景的圖片名稱，第一個數字為互動場景的 ID，第二個數字為該互動場景的動畫幀數。
3. 接下來有第二個數字 b 讀入有幾個互動物品。
4. 接下來有 b 行，每一行讀入一個字串與一個數字，代表該互動物品的圖片名稱與 ID。
5. 接下來有第三個數字 c 讀入有幾個怪物。
6. 接下來有 c 行，每一行讀入一個字串與數字，代表該怪物的圖片名稱與 ID。
7. 接下來有第四個數字 d 讀入有幾個非玩家角色。
8. 接下來有 d 行，每一行讀入一個字串，代表該非玩家角色的圖片名稱與 ID。
9. 接下來有第五個數字 e 讀入有幾個單純場景物件。
10. 接下來有 e 行，每一行讀入一個字串，代表該單純場景物件的圖片名稱與 ID。

如此，我們就能夠有效率的在主核心建立起 ID 對應圖片名稱的 map，只需要在同樣物件不同幀的圖片，檔名命名為 <圖片名稱><第幾個幀數>.bmp 即可，例如：一扇門可能有五個幀，所以我們命名為 door.bmp, door2.bmp, door3.bmp, door4.bmp, door5.bmp，如此一來，我們不需要特別為了讀取一個物

件新增 mygame.cpp 讀取素材的程式，只需要在設定檔與圖片名稱以命名規則來新增即可。

1. 針對互動場景，我們只需要使用框架中多個圖片讀取的功能，一次讀入多個 bmp 圖片（例如：./RES/door.bmp, ./RES/door2.bmp, ...），並且以 ID 為 key 寫入 map，預設這個物件的動畫為觸發後才進行。
2. 針對互動物品、單純場景物件，我們只需要根據圖片名稱讀入 1 個 bmp 圖片即可，並且以 ID 為 key 寫入 map。
3. 針對怪物與非玩家角色，我們只需要根據圖片名稱讀入 2 個 bmp 圖片即可（./RES/怪物名稱.bmp、./RES/怪物名稱 2.bmp），並且以 ID 為 key 寫入 map，並且預設動畫為永久進行。

如此一來，在應付 100 甚至 1000 個物件時，我們不須在主核心讀取遊戲素材的部分新增多餘的行數，也能應付物件的讀取，讓物件的註冊非常的快速且方便。

B. 動態遊戲物件變更

在遊戲中，容易遇到一個事件就必須要更動遊戲地圖的部分，例如在第 20 層時會遇到假公主，導致被困在只有假公主的空間，此時必須要將地圖 Material 的部分將路（road）變更成牆（wall）。

這時候如果地圖的建立是因為一格只能讀取一張圖片，就較難做到將一格更新成其他圖片的方法，因此在我寫的框架上，支援了讀取多張圖片，但同時可能會造成大量圖片讀取的問題（每一格讀取了近 100 張的物件），會讓遊戲的讀取稍微久一點，但同時讓遊戲地圖變得更靈活，能夠輕鬆的使用 3 維（第一維為第幾列，第二維為第幾欄，第三維為呈現的物件）來更新地圖物件。

C. 遊戲對話框架優化

考量到每一個 NPC 都是固定的對話模式，有些 NPC 可能會販賣東西，所以我們可以利用這個特徵來寫一個對話框架，並且使用 txt 來撰寫對話框架，在這次實作中我成功地寫出了這個對話框架，主要流程如下：

1. 第一行讀入一個數字 n ，代表有幾個對話事件（一個 NPC 可能有連續的對話框，所以這個是必要的）
2. 接下來有 n 筆對話，每一筆對話第一行有一個數字 m ，代表對話的行數，第二行有一個字元 c ，代表這個對話應該要用哪個對話框架呈現，分成 T、B、C，代表上方對話框架(NPC)、下方對話框架(自己)、中間對話框架(獲得東西)，接下來有 m 行文字，每一行即為對話。
3. 接下來有一個數字 k ，代表這個 npc 要販賣多少東西。
4. 接下來有 k 行，每一行文字代表販賣東西的選項名稱。
5. 接下來有一個數字 r ，代表販賣東西的子選項名稱。

使用這個框架之後，在設定每一個 NPC 的對話就會輕鬆很多，固定對話的部分不用特別寫死在遊戲上，而觸發特殊對話則會由 Event 處理，由於部分規則較複雜，所以不適合寫成框架，但我同時開發出了一個可以處理即時對話的部分，只需要讀入二維的文字 vector、一維的字元 vector，以及其他參數即可做出同等的功能。（見 `Dialog::setNPCTemporaryDialog`）。

四、結語

1. 問題及解決方法

在這次實作中遇到了很多問題，但也都成功一一排除了。

第一個是動畫同步的問題，由於動畫是隨著 `onShow()` 一幀一幀的顯示，要與遊戲的機關同步出現了一點問題，這個部分大概花了兩天的時間把同步的部分調整好了。

第二個是建地圖的困難問題，由於地圖是 11×11 的方格，這樣意味著如果要建出一張地圖，需要手打 242 格 ID，顯然在手打的部分會耗盡許多時間。因此我選擇使用機器學習來建模，透過線上學習 (Online Learning) 的方式，將機器判錯的圖片加上標籤丟回去資料庫再次學習，在短時間內就能形成數千筆測試資料，這個部分使用了 Python 的 Scikit Learn 的 Random forest classification 套件免除了需要手刻訓練模型的部分，單純處理完資料得到報表就能夠修正機器判錯的部分，這部分也開源在我的 Github 上。

第三個是現行的框架對我想要的功能支援不足，這部分我在原先的框架上新增了自己想要的功能，變成自己更新了這個框架，例如：移除 status bar 或 tool bar，修正了在不符合標準的解析度時依然能夠全螢幕的功能，以及新增讀取條的功能等，這部分也開源在我的 Github 上，希望可以供給未來修這一門課的同學有更加歡樂的開發體驗。

2. 貢獻比例

由於是自己一個人製作，所以是自己 100%。

3. 自我檢核表

	項目	完成
1	完成協議書上所描述的最小關卡數量。	V
2	完成專案權限改為 public。	V
3	具有 debug mode 的功能。	V
4	解決專案上所有 Memory Leak 的問題。	V
5	報告中沒有任何錯字，以及沒有任何一項遺漏。	V
6	報告至少保持基本的美感，人類可讀	V

4. 收穫

這學期讓我學到了如何使用 C++ 來開發遊戲，在開發大型專案時也注意了一下自己的程式碼的整潔度、或者有沒有什麼更好的方法可以提升開發效率、提升遊戲效能等等，在研究遊戲時也遇到了許多問題，但也都一一解決，也讓自己的作品集多了一個新的遊戲。

5. 心得、感想

這一次開發遊戲的過程非常開心，也因為自己寫的框架在遊戲上幫助了我不少，例如將動畫功能整合到 CMovingBitmap 上，讓遊戲在開發時更加的有效率，也讓我更著重於把遊戲做到完美無缺。

6. 對於本課程的建議

沒有，謝謝老師。