

# 國立高雄大學

## EEB512 計算機程式

### 109 學年度第一學期期末報告

#### SUMA 遊戲設計



第四組：柯柏宇 B1085107（組長）

黃泓洧 B1085118

盧奕睿 B1085123

蘇浚緯 B1085151

指導老師：吳志宏教授

中華民國 110 年 1 月 18 日

# 目錄

設計理念-----	1
主要場景說明-----	2
系統架構-----	4
Allegro 補充函數-----	5
人員分工-----	6
組員心得-----	7
其    他-----	9

## 壹、設計理念

### 一、設計靈感：殭屍海嘯 zombie tsunami



### 二、原版遊戲內容：

玩家將操控殭屍，透過吃掉人類將人類化為同伴，途中需跳躍躲避地洞、炸彈、汽車以及飛機。增加的同伴可以做為玩家的血量，持續行走的距離將做為分數，而得到的金幣則能夠購買道具。

### 三、我們的遊戲：

玩家將操作主角，透過跳躍的方式避開洞窟以及障礙物，並在途中吸收同伴以及金幣，達到目標時間即可通關，而同伴將作為血量，金幣將作為分數。

### 四、相同之處

1. 吸收路人納為同伴，並以此為血量。
2. 能夠收集金幣
3. 有障礙物、坑洞等

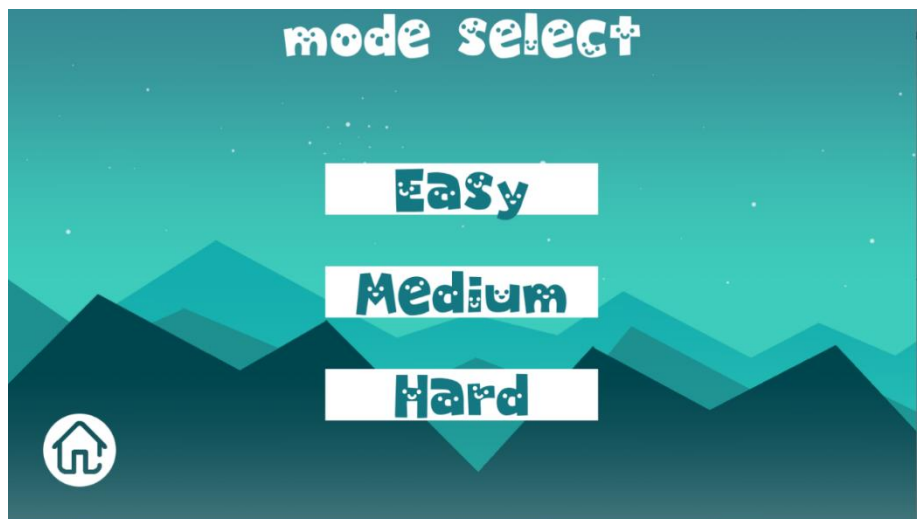
## 五、不同之處

1. 腳色能夠左右移動
2. 金幣作為分數依據
3. 有關卡分別
4. 有 boss 存在

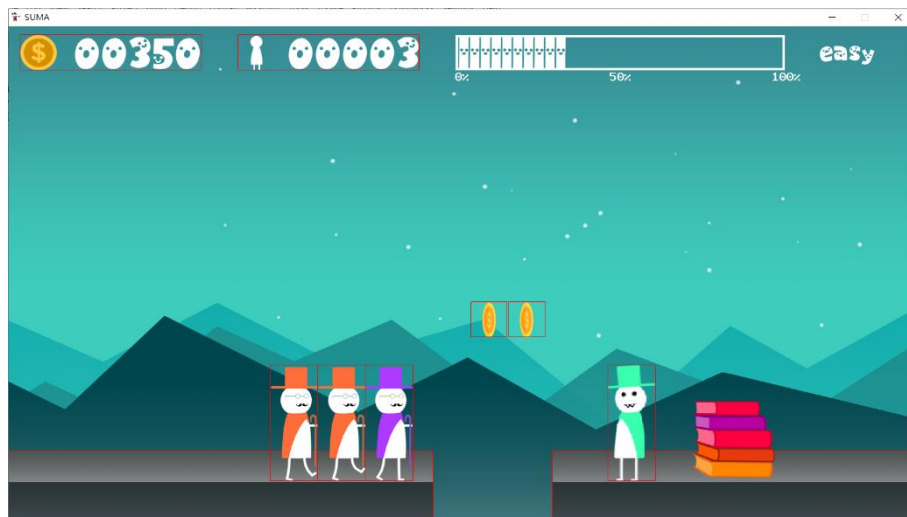
## 貳、主要場景說明



▲初始遊戲選單



▲難度調整



▲主要遊戲畫面 可以清楚看到遊戲大部分物件

## 一、角色以及物件說明



## 二、遊戲資訊欄說明



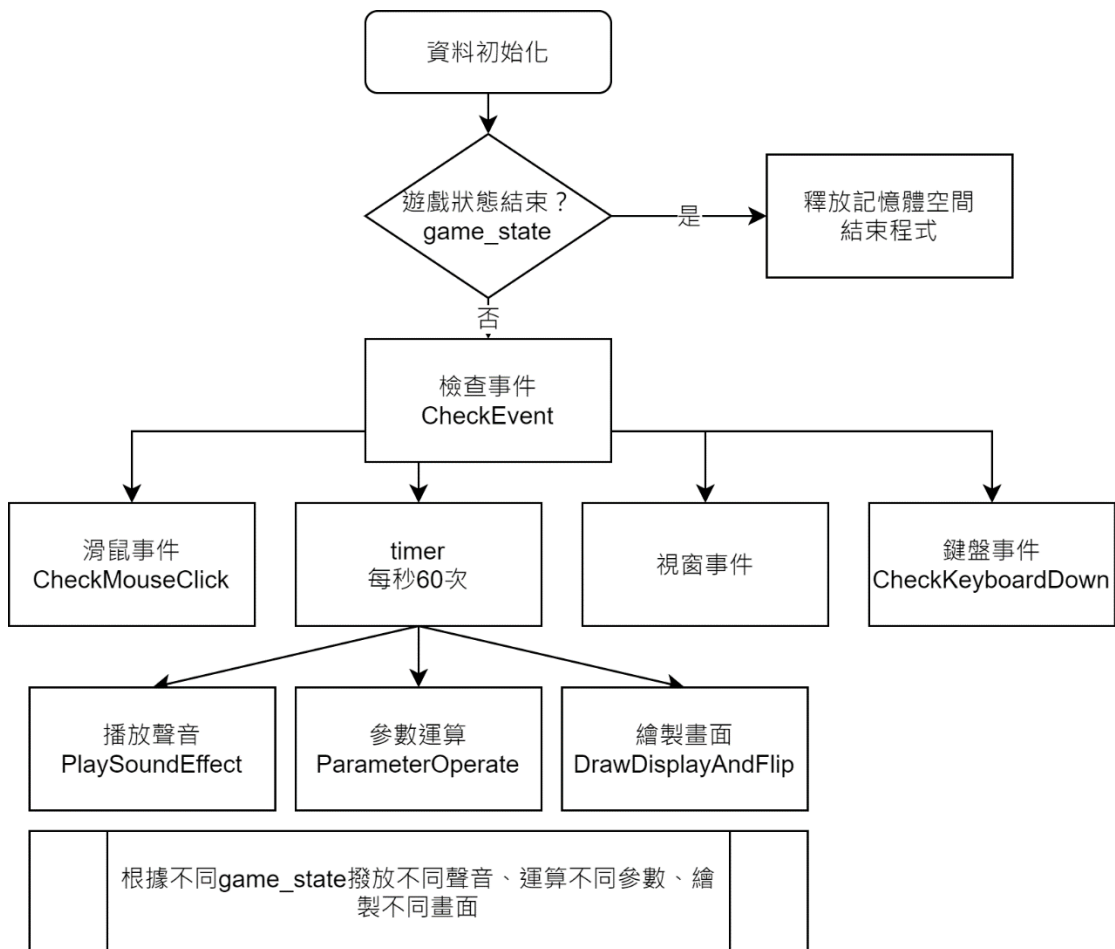
1. 左一：遊戲分數，透過獲得金幣取得，一枚加 50。
2. 左二：主角血量，透過接觸路人所加入的同伴獲得。
3. 右二：遊戲進度條，當遊戲進度推進至 100%時即完成遊戲。
4. 右一：遊戲模式，目前顯示遊戲模式(關卡一、二或三)。

### 三、關卡區別說明

1. 第一關：有地洞及障礙物，起始血量為 5。
2. 第二關：有地洞、障礙物、隕石及電磁學公式，速度加快，一旦碰到電磁學公式，則主角將有高機率不能再獲得同伴，起始血量為。
3. 第三關：有地洞、障礙物、電磁學公式及大魔王(BOSS)，速度加快，有一定機率無法獲得同伴，起始血量為 10。(2 個角色其中個為干擾用，另一個會吃掉同伴，吃掉的當下會扣血量)

### 參、遊戲架構

#### 一、程式架構



▲程式主要架構圖

說明：

在程式中，參數運算歸參數運算，算完後數據存至每個物件的 struct，再由繪製畫面做統一繪製。將功能分開可以有助於開發時的效率及除錯，並減少衝突之發生。

## 二、資料架構(重點想法)

1. typedef struct MainDataStut：內包含遊戲模式、難度、進度、參數…等。
2. typedef struct AllegroObjStut：內包含所有 allegro 相關物件(遊戲中會使用到的都在裡面，如：角色、地板、障礙物、聲音…等。
3. typedef struct ObjectStut：物件用，內含一指標指向下一物件(LinkedList 用法)，用於須產生不定數量物件或隨時會需要刪除某一特定物件時使用。

## 肆、Allegro 函數補充

1. 當遊戲有需要暫停功能時，可以使用下方方法將事件監聽器及 timer 暫停、恢復。

```
1. void al_pause_event_queue(ALLEGRO_EVENT_QUEUE *queue, true);
2. al_stop_timer(ALLEGRO_TIMER *timer);
3.
4. void al_pause_event_queue(ALLEGRO_EVENT_QUEUE *queue, false);
5. al_start_timer(ALLEGRO_TIMER *timer);
```

2. 當遊戲有需要進行碰撞時，可以使用此函數畫出物件邊框，方便除錯。

```
1. al_draw_rectangle(x1, y1, x2, y2, al_map_rgb(255, 255, 255), 0);
```

3. 設定程式視窗 icon。

```
1. al_set_display_icon(ALLEGRO_DISPLAY *display, ALLEGRO_BITMAP *icon)
```

4. 兩種不同撥放音效的方式，前者可以重疊撥放，後者不行(可適用於背景音樂)。

```
1. al_play_sample(ALLEGRO_SAMPLE *spl, float gain, float pan, float speed, ALLEGRO_PLAYMODE loop, ALLEGRO_SAMPLE_ID *ret_id)
2. al_set_sample_instance_playing(ALLEGRO_SAMPLE_INSTANCE *spl, bool val)
```

5. (非 allegro)第 3. 設定之 icon 只會顯示於是窗上，並非程式的 icon，若需改變程式 icon 可於 code::block 專案中加入一檔案：icon.rc，內容如下：

```
1. MAINICON ICON "icon.ico" //.exe file's icon
```

6. (非 allegro)隱藏 consloe 視窗方法，可於 main() 中加入：

```
1. HWND hwnd = FindWindow("ConsoleWindowClass", NULL);
2. if (hwnd) ShowWindow(hwnd, SW_HIDE);
```

## 伍、人員分工

柯柏宇(組長)：隕石製作、角色移動以及跳動、BOSS 機制、砲彈製作。

黃泓洳：碰撞機制、金幣製作、待吃跟班(同伴)生成機制、障礙物製作、物件隨機生成機制。

蘇浚緯：Linked list 物件模板、記分板製作、遊戲選單製作、角色動畫(含圖檔)、待吃跟班製作、遊戲整合、跳躍優化

盧奕睿：全部美術製作(背景、主角、障礙物、BOSS)、音效觸發、遊戲進度條機制、地板製作



## 陸、組員心得

柯柏宇：

原本以為更多人一起寫程式以後，會變得更加輕鬆，然而事情果然沒有想像的那麼簡單。

分工以後雖然效率增加，相對的，複雜性也明顯增加，除了程式設計、邏輯的演算以外，成員之間的溝通也相當相當重要，如果是獨自撰寫，變數的命名、函式的功能就會相當清楚，但多人一起撰寫時，一切都豬羊變色，常常會看不懂其他組員的變數命名或是函式，因此整合時更是痛苦中的痛苦。不過也非常幸運，我們這一組有使用 github 來做版本控制，雖然需要花時間學習，但帶來的效益實在非常大，除了降低了整合的時間，也大大降低整合的痛苦。

過往都是獨自寫程式，這次的經驗實在非常寶貴，不僅能夠與組員交流彼此不同的想法，如何有效溝通、分工，增加分工的效率，也相當重要。通過這次的經驗，我在組員身上學到很多，不僅打開自己的視野，也發現自己不足的地方，相信可以在未來成為相當有幫助的養分。

黃泓涓：

這次的期末報告算是我第一次從無到有的報告，從一開始的主題尋找到評估可行性，更甚至到計畫反覆修改，這些報告的前置作業我之前不太碰過，我們甚至開了一場線上會議討論這些前置作業接近 2 小時；開始實際打程式時，也遇到了之前自己打程式不會遇到的程式互衝，一開始發現是 while 的問題，但漸漸地不只 while 有問題，即使我們用了 GitHub 都不能完全根治這問題，最後憑藉著組員高超的程式整合，將程式有系統的分類，才完全處理了這個問題；在最後一周，要做程式收尾時，發現了東少一點點稀少一點點部件，在裝上去之後又出現零星的 Bug，連續兩天一直奮鬥著；藉由這次

期末報告，我真的學到很多程式和程式以外的經驗，同時也很高興能與組員一起完成這項作業。

蘇浚緯：

在這學期的課堂中，我覺得令我感到相當欣慰的是我們這組的組員都很願意為了專題付出。在這次的期末專題中，我們不僅將課堂所學能結合應用，我認為最關鍵的收穫在於如何與小組進行分工，專案固然龐大，剛開始我們就在想如何有效率地進行開發，尤其要避免的是過於無效率的開發，因此我們前期花了非常多的時間在溝通彼此的想法、專題架構及運作流程。我覺得有一個好的專題架構及開發流程就像一棟地基紮實的建築，要有好的架構才能蓋得夠高、夠穩。此外，為了同時避免專案整合的困難，我們也透過這次專題的機會實作練習了 git 版本控制的應用，雖然在前期非常痛苦，不過到後來，真心感受到其方便性，尤其是當小組每個人在開發不同功能時，只有少部分需要進行人工解決衝突，也能加快開發的速度，最重要的是有復原的功能，可以調閱紀錄看那位組員是否因為誤改相關程式碼而導致程式異常。總而言之，這堂課無論是個人作業還是小組專題，都帶給學生一定的壓力，不過我覺得這是物超所值的，能順利完成這門課的課程內容，我也感到十分地開心。

盧奕睿：

這次的課程應該是自從進入大學到目前為止數一數二痛苦的，我現在在學二宿舍的自修室打這個心得，時間是一月 18 號上午三點，要用半個學期的時間寫一個遊戲實在是有相當的難度，尤其是要在兼顧其他電機必修的狀況之下，我敢說我們這一組除了我以外，程式的能力都是系上前幾的，在這種情況下我們還數度面臨危機，就知道到底有多困難了，非常感謝他們的

幫助，令我學習到了很多，真的，以 git 來說，雖然高一時參加電腦社的營隊就已經稍微的講解過了 git 的基本原理，也辦了 github 的帳號，不過到了大二的計算機程式，在浚緯的詳細教學之下，我才真正的了解到底要如何應用 git，這應該是我在這門課學到最有深獲的事了，除了我以外的每個人，不只浚緯，泓洳、柏宇也幫助了我許多，跟他們一組，除了學到程式的幾巧以外，我更學到了如何在專案中與其他人分工合作，我覺得這甚至比課堂本身的內容還要有用，除了老師的教導以外，也很幸運能夠跟一群學霸一組，我才能學到這麼多。

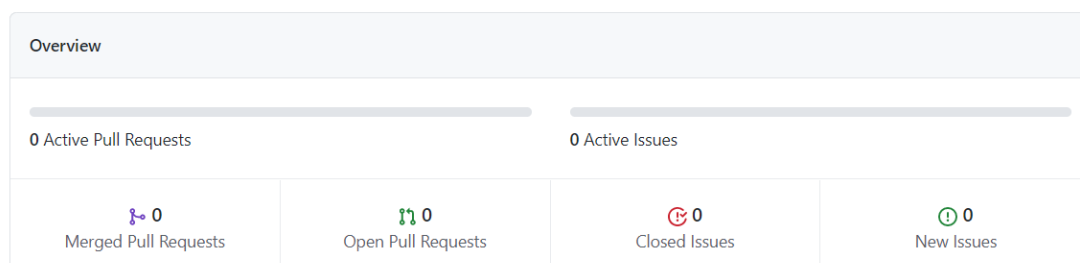
## 柒、其他

使用 GitHub 並發揮專案管理與多人分工之版本控制

Github 連結：<https://github.com/JyunWei-Su/EEB512T4>

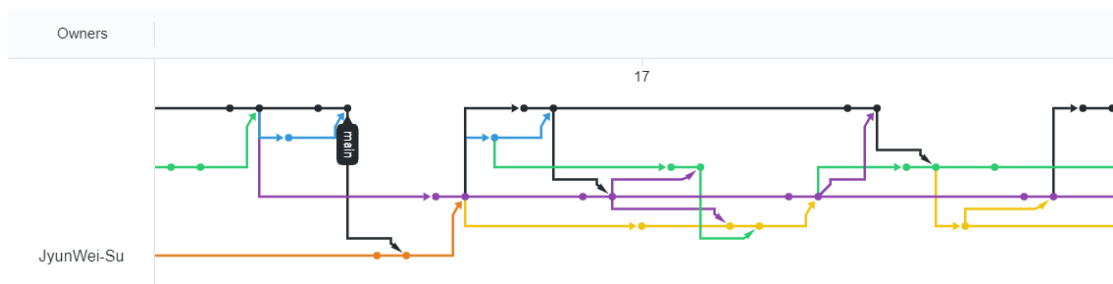
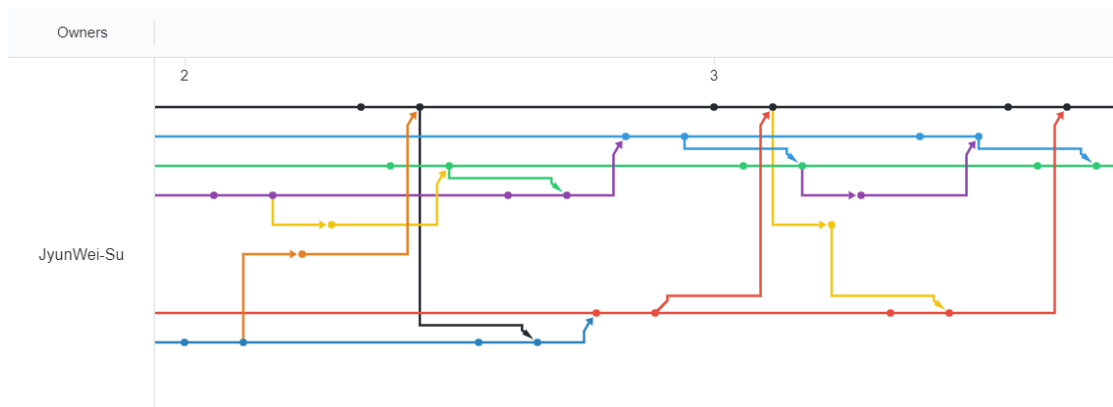
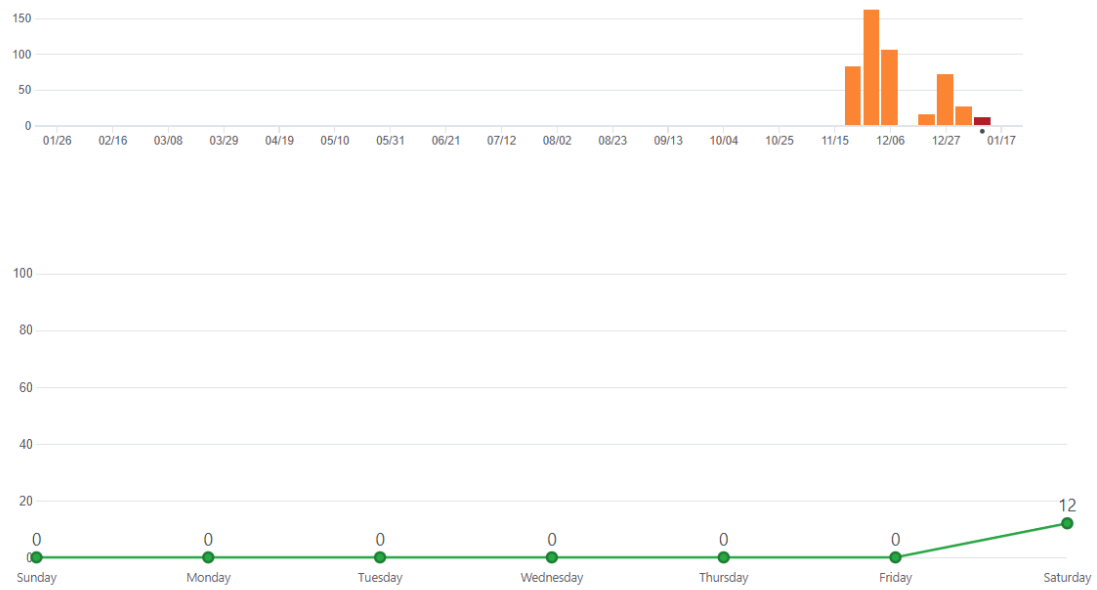
December 17, 2020 – January 17, 2021

Period: 1 month ▾



Excluding merges, **4 authors** have pushed **128 commits** to main and **157 commits** to all branches. On main, **0 files** have changed and there have been **0 additions** and **0 deletions**.





Network graph 部分截圖