

程式設計(111-1)期末專案報告書 第 H 組

周昀蓉 B1170523、謝友毅 B11705024、林溪稚 B11705025、李盈盈 B11705039

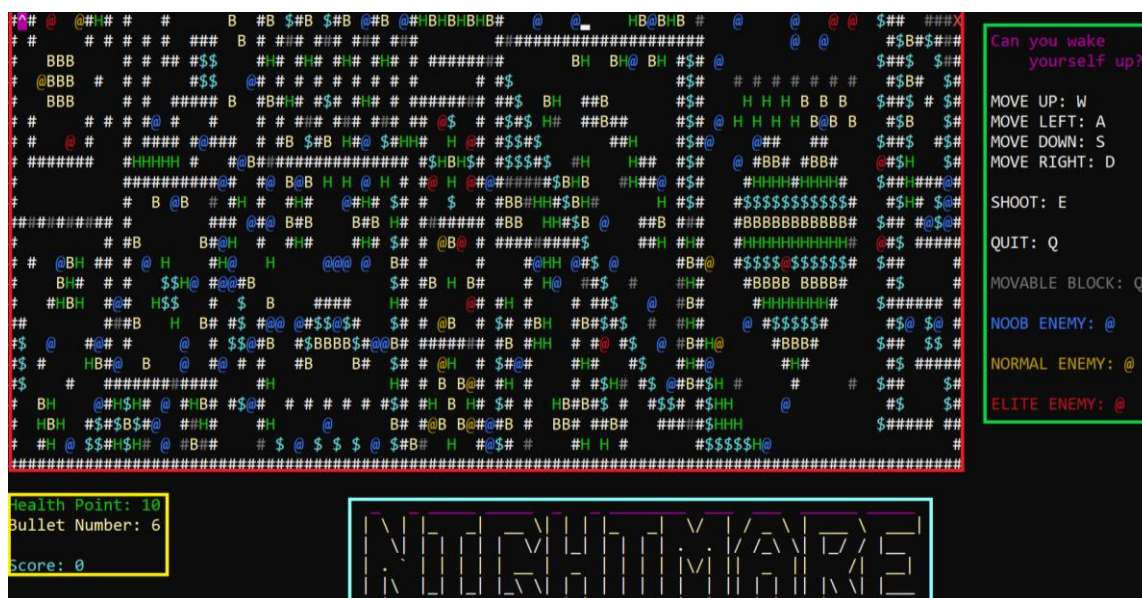
1、摘要

經過這學期九次程式作業，加上一次期中專案的震撼教育，我們寫程式的能力獲得了大幅的提升。得知期末專案可以自由選擇主題，希望藉這個機會學以致用的我們，選擇了變化性多且視覺化的遊戲製作，希望藉這個機會好好地學以致用。

2、遊戲簡介

在查找過許多資料以後，我們打算製作一款在主控台(console)上操作的復古風單機遊戲，玩家將利用鍵盤操控主角移動，拾取地圖上的各種道具，技巧性地移動物件，同時躲避、殲滅各種狡猾的敵人，目標是盡可能地維持生命值抵達終點，同時獲取更多的分數。

我們將本遊戲命名為「NIGHTMARE」，一方面純粹是因為主控台是黑色的，一方面是因為遊戲內容像夢魘一樣。



圖一、遊戲畫面

- (1) 左上紅框：地圖
- (2) 左下黃框：玩家資訊(角色生命值、子彈數量、分數)
- (3) 下方藍框：標題
- (4) 右方綠框：遊戲提示

遊戲方法

玩家可以利用 W、A、S、D 鍵(上、左、下、右)操控角色移動，朝著終點推進，同時蒐集道具、推動牆壁來擊倒敵人或躲避敵人的襲擊。另外也可以使用 E 鍵發射子彈，以消滅敵人，或是破壞特定物件；在子彈數降至 0 時，角色將無法射擊。

獲勝條件是在角色生命值>0 的狀態下觸及終點，若達成，將進入獲勝畫面，玩家勝利；反之，角色被擊倒，玩家失敗，則進入失敗畫面。



圖二、按鍵操控示意圖

3、 介面與程式內容介紹

一、開始選單：

在遊戲一開始時，會先看到遊戲標題，接著進入主選單畫面，包含開始 (PLAY)、模式(MODE)和 設定(SETTINGS)。玩家可以利用 W、S 鍵上下選取選單的選項 (兩側有箭頭指示)，按下空白鍵以後，系統將執行選項上的內容。

按下開始將直接進入遊戲畫面，按下模式時可以選擇簡單、困難兩種遊玩模式，按下設定時可以選擇是否開啟遊戲音樂，其中模式和設定這兩個子選單內有回到首頁選項，按下可以返回到首頁。

二、地圖：

地圖包含以下幾項資訊：

- 角色位置與面對方向
- 敵人的移動位置
- 不同道具
- 遊戲邊界範圍
- 終點位置

為了方便記錄螢幕、地圖上特定位置的資訊，我們建立了一個**自訂變數型態 Position**，可以存放名為 **x、y 的兩種數值**，用以方便標示角色、道具等物體的位置(另外有對 Position 做加法的 operator overloading，在射擊的函式中有使用到)。

內容記錄

地圖的內容是由一個**靜態二維陣列 map** 紀錄(寬度為 MAP_SIZE_WIDTH、高度為 MAP_SIZE_HEIGHT)，並對遊戲場地上不同種類的物件以不同編號標示，紀錄其位置以及狀態：

物件	終點	主角	空地	牆壁	子彈補給
編號	-2	-1	0	1	2
物件	生命值補給	分數包	可移動方塊	敵人	陣亡敵人
編號	3	4	5	90 以上	-99

內容更新

在遊戲期間，角色的移動、射擊、利用道具，以及敵人的移動、陣亡都必須要及時更新地圖陣列的紀錄，以便後續的各種操作、判斷。

三、角色：

對於角色，我們將它設置成**名為 Player 的物件**，並賦予其所需的數值如位置、生命值資訊等，也加上一些 member function 增加程式可讀性。玩家累積的得分以全域變數紀錄。另外，當生命值降至 0 以下，角色將死亡，此時遊戲結束。

如同前述，玩家可以操控角色在地圖上移動、攻擊，以不同方向的圖示表示角色正在面對的方向：

^	V	<	>
朝上	朝下	朝左	朝右

角色資訊

角色有兩項重要且會顯示給玩家看的資訊：**角色生命值與子彈數**，另外還有記錄角色位置的變數和記錄角色目前移動方向的變數。為了方便在主程式中控制玩家按鍵輸入對角色移動方向的改變，我們將記錄方向的變數設為 public variable。

角色移動

由玩家操控 W、A、S、D 鍵(上、左、下、右)進行移動。我們首先判斷角色移動方向的前方是否有物體，並根據不同狀況執行對應的程式：

(一) 前方無物體

將角色原先在螢幕上顯示的位置改為空格，並在其移動方向上的下一格輸出角色(以該移動面向的圖示輸出)，同時將地圖陣列中角色所在的位置改為 0，陣列中移動方向上的下一格改為 -1。

(二) 前方為道具(分數包、子彈補給、生命值補給)

增加對應的項目的數值，並將該格如同上述方式更改數值。

(三) 前方為可移動牆壁

會先判斷可移動牆壁下一格是否有物體，若可以推動，則角色向前移動，可移動方塊則移至角色的下下格。

(四) 前方為牆壁

角色不移動。

角色射擊

玩家按下 E 鍵時，角色有一發射的函式對應，當子彈擊中敵人時，會將該敵人在地圖陣列中原本所在位置的值改為 -99，當敵人下次移動時，若發現其原本所在位置的值已被修改為 -99，則判斷其已被擊倒，消失在地圖上。

當子彈在行進過程中遇到道具時，將會從道具底下穿過；若遇到可移動牆壁將會將其消除；遇到牆壁子彈則會停止。

子彈的行進過程中只有判斷與輸出，而不會在地圖陣列中記錄其數字，故我們沒有將它設置為物件。也因為我們的設計方式，子彈會快速地在一段函式中以一迴圈執行，導致子彈射出的過程中，其他物體皆不會移動。

以下為射擊函式的 pseudocode：

```

set that meet item = false
for i from 1 to range
    if there is nothing at the next position
        bullet move to the position and record that meet item = false
        if didn't meet item and is not first step -> clear previous position
    else if there is enemy
        if is first step -> change position on map to -99
        else if didn't meet item
            clear previous position;
            change the place on the map into -99
    else if it is an item
        bullet move to the position;
        if didn't meet item and is not first step -> clear previous position
        record that meet item = true
    else if it is a fixed wall -> return;
    else if it is a movable wall
        if didn't meet item and is not first step -> clear previous position
        crash the movable wall;
if didn't meet item and is not the first step
    clear previous position

```

四、道具：

道具可說是這款遊戲中的通關基礎，玩家可利用子彈擊倒敵人、增加生命值，也因此玩家能否獲勝的關鍵往往取決於其獲取的道具是否足夠。道具共分為三種，分別是分數包、子彈補給與生命值補給，大致介紹如下：

道具名稱	圖示	功能
分數包		拾取後增加分數
子彈補給		拾取後增加子彈數
生命值補給		拾取後增加生命值

五、敵人：

進行遊戲時，玩家將會遇到各式各樣的敵人，每種敵人擁有不同的**生命值**、**攻擊力**、**速度**以及**行進方式**。

當敵人碰觸到玩家，或是玩家主動碰觸敵人時，角色的生命值將會根據敵人的攻擊力數值減少，同時敵人會消失。除此之外，當敵人之間互相碰觸時，敵人將各自折往其他方向移動。另外，敵人在碰觸到道具時，由於敵人不會拾取道具，故會先記錄道具的種類，隨後在敵人離開該位置時，將該種類的道具再放回。

敵人是此遊戲中的另一種物件，包含的變數有**位置**、**移動速度**、**生命值**(也是敵人會對角色造成的傷害數值，同時決定了敵人的等級)、**移動方向**以及**編號**(為了區分不同種類的敵人。設置的數字皆大於 90)，另外還有敵人**是否存活**、**移動計數器**(moveCnt)和**經過道具時用來記錄道具種類的變數**。

由於我們希望設計各種類型的敵人，故**利用了 inheritance 和 polymorphism** 來模組化程式，其中 Parent object 的名稱即為 Enemy。

敵人種類與移動

在 class Enemy 中，設有一用來控制敵人移動速度的靜態變數。敵人以移動方式區分種類，包含以下三種：

- (一) 右迴旋型：此類型敵人會在碰觸牆壁後向其右方前進，以順時針方向移動
- (二) 左迴旋型：此類型敵人會在碰觸牆壁後向其左方前進，以逆時針方向移動
- (三) 隨機移動型：此類型敵人會隨機地移動，需要格外小心，轉眼間或許他正朝著你所在的位置快速移動

各敵人在碰觸到牆壁或其他敵人時都會轉彎，而為加速轉彎的判斷，我們設置讓各敵人判斷是否需要轉彎並變換移動方向的時機和敵人可移動的時間分開，讓敵人在需要轉彎時能較快地變換至前進的方向，敵人移動時的 pseudocode 如下：

```
if enemy is not dead
    moveCnt += speed;
    if meet wall or meet another enemy or the end
        change moving direction
    else if moveCnt >= moveThershold
        enemy move
```

敵人隊伍

在我們的設計中，應有許多敵人同時出現在地圖上，故我們利用 polymorphism 的概念，建立了一名為 **EnemyTeam** 的物件，內含**生成各式敵人的函式**，以及一個**讓所有敵人移動的函式**。

六、牆壁

牆壁是地圖中很重要的架構，限制了遊戲的邊界以及角色、子彈、敵人三者的移動範圍，讓關卡呈現出須遵循的基礎規則。牆壁共分為兩種，大致介紹如下：

(一) 固定牆壁：

包含玩家、子彈、敵人在內皆無法穿越，地圖上以白色#顯示

(二) 可移動牆壁

可以被子彈擊破、被角色推動，但無法穿越，地圖上以灰色#顯示。玩家操作得當時，可以用它阻擋危險的敵人，不當則可能讓自己陷入困境。注意當可移動方塊後有另一個可移動方塊或是敵人時，角色將無法推動。

七、結束畫面

結束畫面分為中途退出、失敗與勝利畫面三種：

中途退出

按下 q 鍵後遊戲停止，畫面只顯示「Game Over!」，且程式結束執行。

失敗畫面

若玩家在遊玩過程中生命值歸零，角色陣亡後將進入失敗畫面，播放失敗音樂的同時，會先出現一段文字告知角色失敗，按下空白鍵可繼續看下行文字，在該行文字的輸出過程中若按下空白鍵則可使文字一次顯示，最後程式結束。

通關畫面

玩家若在生命值歸零前成功觸及終點，則可進入勝利畫面，首先出現一行文字並播放音樂，文字顯示結束後，玩家可看見記分板，在其顯示一段時間後，程式便會結束執行。過程中按下空白鍵皆可加速介面執行。

八、音樂

遊戲音樂皆使用音樂軟體 LMMS 製作。

程式的部分使用 thread，以平行音樂和遊戲程式的運行。遊戲過程中總共有四首不同的音樂，藉由 PlaySound() 函式，我們成功地播放了音樂，但過程中因為對此函示以及 thread 的功能不熟悉，我們花了一段時間查詢資料與摸索，多次測試後才能順利地在畫面切換時切換音樂。

4、 可改進的地方&未來展望

(一) 未能修正的遊戲錯誤

在各選單畫面中按下其下段程式中可讀取的按鍵時，會使下段程式開始執行時接收到該按鍵，並立即執行對應的程式(未確定是在起初的程式中已無需接收按鍵時才會發生，或可以跳過目前所在程式去對下段程式造成影響)

(二) 遊戲中可精進的部分

1. 敵人應該要有自己的血量、顯示其攻擊力(如以顏色標示)
2. 發射子彈時角色和敵人應該也要能夠同時移動
3. 各項物體大小與螢幕顯示長寬比的不對稱
4. 音樂無法切換

(三) 更多未來展望

1. 玩家可以選擇擁有數值不同的角色，例如：力量型、防禦型、敏捷型
2. 新增成就系統，例如：三分鐘內通關、消滅所有敵人等
3. 敵人、道具可以更加豐富，例如：射擊型、爆炸型敵人，護盾（阻擋攻擊）

5、 工作分配

B11705023 周昀蓉：書面報告、影片製作

B11705024 謝友毅：程式 60%、書面報告

B11705025 林灃稚：程式 40%、書面報告

B11705039 李盈盈：影片製作、書面報告

6、 組員心得

周昀蓉

雖然以前就有撰寫遊戲的經驗，不過只使用 C++ 是第一次，在還不知道想做什麼風格類型的遊戲的時候查了很多資料，像是在 youtube 上觀摩其他人寫的小遊戲以及程式碼範例，接觸到 SFML 和 SDL2 這兩個 library，雖然在架設環境的時候遇到了一點小問題，不過實際上手測試後，能夠跳脫 cmd 的黑框框，實現多元的視覺化，比一開始想像能夠做到的事情還豐富許多。

經過討論後，我們決定遵循傳統的 console application ASCII 來製作遊戲。有了其中專案的經驗，這次費了比較大工夫在把程式碼好好模組化，讓組員間能夠龐大的程式結構裡快速了解夠部分的執行功能。另外這也是第一次使用 GitHub，不過感覺沒有最大化利用其版本控制的優點，如果能夠更熟練使用的話，對於以後團隊的程式專案開發應該會更有效率。

雖然這次的遊戲專案沒有用上 SFML，不過在觀摩其他組的遊戲後讓自己也很想著手實作一番，希望在快兩個月的寒假裡能夠成功撰寫一個屬於自己的小遊戲。也感謝組員們的互助合作，以及老師給我們一個傳統考試以外的期末作業，我覺得比起期末考，這樣的遊戲專案反而更能促使大家自學，了解 C++ 更豐富的應用面。

謝友毅

從以前開始，我就幻想著將來的某一天，我能夠製作一款屬於自己的電子遊戲，但那時的我毫無頭緒，不知何時能夠完成這個願望……沒想到的是，這個心願在上大學，初學 C++ 程式三個月後竟然就達成了呢！

當初得知期末專案可以自由選擇主題時，我非常地興奮，同時也很焦慮，因為擁有的選擇越多，越會讓人猶豫不決。最終，我們這組選擇了製作遊戲，此時我正朝

著一個未知的領域前進，而且擁有的是僅僅三個月的程式基礎，當時的我甚至不知道會不會成功。

專案製作開始時，面臨到的又是一大堆的選擇，遊戲種類百百種，光是要選擇一個我們能夠勝任，又不失挑戰性的遊戲來製作就花費了不少時間，再三思量後，我們選擇利用控制台（console）製作一個平面的動作遊戲。

製作時，因為沒有像 SFML 的圖形輔助，我們僅利用 cout 來呈現畫面上的各種複雜動畫，然後大部分機制都是應用老師這學期教到的語法，還有運用到老師覺得大部分同學到期末都不會熟悉的 polymorphism 呢，可說是學以致用呢！

然後我們成功地製作出預想中的遊戲，發現整個遊戲真的是挺不錯的！玩法可以很多樣，要過關也有一定的挑戰性，整體視覺效果很壯觀，音樂還是組員原創的。

成果發表時，自己和大家的努力、巧思能夠被老師和同學們看見，是一件讓人非常高興、感動的事～，所以我覺得老師之後開課繼續保留期末專案的傳統其實是蠻棒的。

最後，感謝 H 組組員們的互相幫忙、合作，陪我度過了一個充實的期末。

林漢稚

這次的期末專案，我負責的主要是物件化、模組化以及敵人和音樂的製作。

在起初的版本中，各項資訊尚未被適當地包裝，使程式很冗長，編譯時也不易在加入新的內容時維持一致性，因此，我將原先角色控制的各項整理成「角色」物件中的變數和 member function 使主程式中的畫面更簡潔；在製作敵人時，我也運用了上課學到的 inheritance 和 polymorphism 讓敵人種類的新增變得更方便；關於音樂的製作則是我自己的額外嘗試，總共原創了四首音樂，雖然音樂製作和程式內容無關，但運用到 thread 以成功讓音樂循環撥放、換場景時同時換下一首，讓我很有成就感。

我認為過程中最大的挑戰是在物件、模組化時，我花了很多時間在觀察原先的架構；另外，我們也對遊戲中各個邏輯判斷花了很多心思在調整或修正。這些整理雖然很耗費精神，卻也是這份作業讓我感到有趣與更加投入的原因。

至於對我們的作品用 console 來呈現，我其實感到可惜了些，因為報告時看到幾乎所有組別都運用了 SFML 來優化視覺效果以及有許多方便的功能，總覺得如果我們有運用到那些工具，成果還能變得更好。因此作為對自己的期待，我想認識更多輔助程式或遊戲開發的工具，試著更優化遊戲開發環境以及遊戲呈現，另外是期待下次的成果呈現不會像這次的遊戲畫面一樣混亂啊...。

最後，很感謝孔令傑老師給我們做這份期末專案報告的機會！這份作業真的很令我投入且有趣，我也私心希望它能延續下去！

李盈盈

期末專案讓學習到了很多課外知識,例如:一些適用於 macOS 的開發環境、什麼是 <windows.h>、SFML？在製造遊戲上有什麼妙處？和呈現我們作品的 console 應用程式等。此次在 youtube 上學習新的語法時並沒有像開學時那樣什麼都聽不懂，這讓我有了學習下去的動力。聆聽其他組員們口頭報告讓我更直觀的感受到了程式可以運用在很多方面，所以非常希望期末專案可以延續到下一年！

很開心能和 3 位組員一起合作，非常喜歡組員們彼此間相互加油打氣的氛圍。期待自己能在程式方面越來越熟練，寫的更好！