

# 程式設計 期末專案書面報告

第八組 蔡承佑(B13705037) 陳允直(B13705054)

黃妍僖(B13705057) 林宇婕(B13705001)

## 壹、 遊戲簡介

### 一、 遊戲名稱 YOUR LIFE IN NTU IM

### 二、 簡介

玩家扮演一位剛入學的台大資管系新生，為了順利完成學業並從台大畢業，必須通過五個階段的挑戰。每一關代表大學生活中的一個年級，迷宮中充滿了各種考驗與道具，玩家需要努力搜集足夠的學分才能順利畢業。若學分不足，則會面臨延畢的命運！

## 貳、 實際流程

### 一、 遊玩方式說明

透過 W, A, S, D 鍵控制角色向上、左、下、右移動一格，而玩家初始 GPA（血量）為 3.0，歸零時就會被退學（遊戲結束）。另外，遊戲結束時會統計玩家獲得的學分數，超過 134 才算畢業，反之遊戲結局會被判定成延畢。

另外在第三關開始會新增黑暗模式，玩家視野大幅縮小，只能看到自身周圍的地圖樣貌。

而地圖上會遇到許多符號：

#：牆壁，玩家無法穿過

=：特殊牆壁，玩家身上需要有腳踏車才能通過

E：出口，身上需要有學生證

.：學分，搜集足夠的學分才能被判定成畢業，反之則為延畢

@：小傑（敵人），玩家每移動一步時，敵人會往隨機方向移動一格，若玩家碰到一次會被扣 GPA 1.0

特殊道具：

%：學生證，每關必備通關道具，撿到後走到出口才能進入下一關

!：情人，成就道具，撿到與否會得到不同結局

～：腳踏車，輔助道具，撗到後可穿過特殊牆壁

\*：食物，輔助道具，撗到後可增加 GPA 1.0

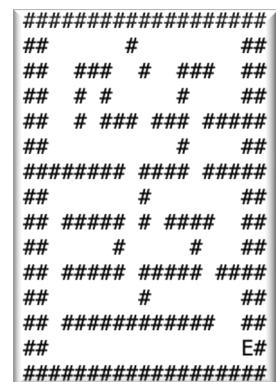
## 二、 關卡詳細流程

輸入 Q 以外的英文字母來開始遊戲（輸入 Q 為結束遊戲），並為自己的角色創建一個名字，會根據取的名字有不同的小彩蛋。

### 第一關：椰林大道 Royal Palm Blvd. (入學)

說明：讓玩家熟悉遊戲操作

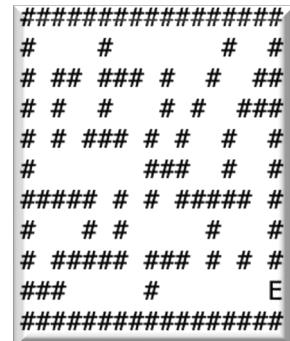
道具：學生證



### 第二關：管院 College of Management (大一)

說明：加入敵人，玩家需試著躲避

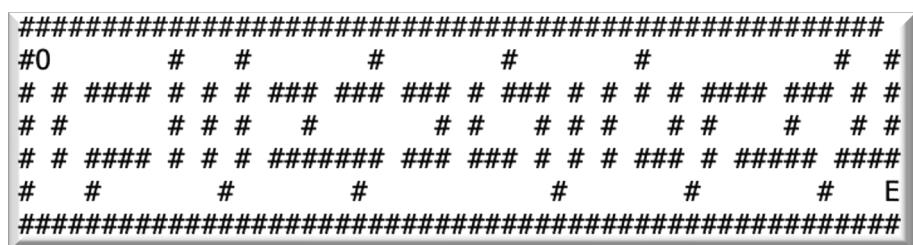
道具：學生證、小傑



### 第三關：公館商圈 Gongguan Night Market (大二)

說明：具有黑暗模式。讓玩家有機會增加 GPA (血量)

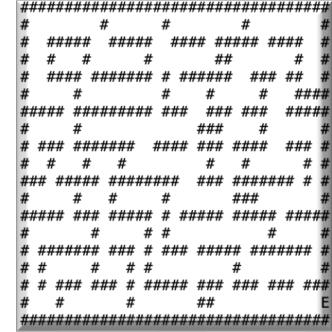
道具：學生證、小傑、食物



## 第四關：管圖 IM's place (大三)

說明：具有黑暗模式。獲得腳踏車的唯一機會，在第四、五關都會有特殊區域要有腳踏車才能解鎖

道具：學生證、小傑、腳踏車、食物



## 第五關：總圖 Labrary (大四)

說明：具有黑暗模式。可以使用腳踏車解鎖的特殊區域，同時新增情人，遇到情人後可以獲得不同的結局。

道具：學生證、小傑、情人



恭喜通關！在最後，系統會統計遊戲過程中玩家所獲得的 GPA、學分數（判斷是否畢業）、是否獲得情人（特殊小成就）

## 參、 程式說明

### 一、 特殊演算法

#### 1. 敵人隨機移動方式

玩家每移動一次，敵人從上、下、左、右四個方向中隨機選一個方向移動一格，如果會撞到牆壁則停留在原地。

每當玩家按每一次 w, a, s, d，*game class* 會呼叫 *main()* 生成一個範圍由 1 至 4 的隨機整數並傳入 *update()*，取得的隨機數傳入每一個地圖上的 *enemy class* 中，執行 *moveEnemy()*。

在 *enemy class* 中，根據取得的數字生成移動方向（1 為上、2 為下、3 為左、4 為右），並判斷新位置是否是合法的，如果為牆壁或其他敵人，則這次的移動不有效；如果合法，則更新座標。

## 2. 地圖輸出

逐一檢查各個座標位置，若是處在夜間黑暗模式，會檢查此座標是否在玩家可視範圍內，不在範圍內會輸出黑色（終端背景為黑色）並跳到下一個坐標繼續。若是有不同角色或是道具重疊，會依照優先顯示順序輸出。

## 二、特殊檔案說明

### 1. Global.h 與 Global.cpp

用來集中宣告與定義一些全域函數的檔案。提供讀取玩家名稱、敵人座標、道具資料和迷宮地圖的函數，並集中處理物件清除、關卡切換等作業。將初始化、顯示與資料載入等功能集中，提升程式架構的清晰度，也易於維護。

***displayMaze()***：負責將遊戲地圖、玩家、敵人、道具在螢幕上以適當顏色與格式顯示。也有「夜間模式」(*isNight*)與「可視半徑」(*radius*)的機制，透過距離判斷使玩家在夜晚只能看到一定距離內的地圖內容。

***displayCell()***：負責顯示單一格子中的內容，包含牆壁、道具、玩家等，以及通過顏色區分特殊元素。

### 2. Game.h 與 Game.cpp

為整個遊戲流程的核心管理中心，包含玩家移動、地圖與敵人生成、輸入等。透過 *currentLevel*、*isRunning*、*isNight* 等變數，掌握整場遊戲的流程。

***start()***：根據 *currentLevel* 載入對應的地圖、敵人、道具資料。控制玩家的輸入與移動，並檢查 GPA（血量）、收集的物品以及是否達成關卡勝利條件。

***getch()***：透過修改終端機設定，使玩家不需按 ENTER 電腦就能即時取得使用者鍵盤按下的字元，藉此控制角色連續移動。

### 三、各項 class 說明

#### 1. Item

為每個道具的抽象基底類別，記錄座標及道具是否已收集。  
同時包含虛擬函式 *getSymbol()*。

*checkAndCollect()*：判定若玩家座標與道具相同，則記錄下  
物品已被收集，並不繼續再地圖中顯示。

#### 2. IdCard, Lover, Bicycle, Food （繼承 Item）

繼承 *Item* 中的座標及判定撿起道具的屬性。實作各自的  
*getSymbol()*，因為每個物件都有獨自的表示符號

#### 3. Character

為每個角色的抽象基底類別，記錄座標及其表示符號。

#### 4. Player (繼承 Character)

繼承 *Character* 中的座標及表示符號的屬性，同時有 GPA、  
學分、判定是否獲取道具的布林值等基本性質。

*resetPos()*：重置玩家的位置，用於新關卡開始或特定事件後  
的初始化。

*move()*：根據玩家輸入的鍵移動玩家的位置，並同時檢查是  
否可通過該位置(牆壁或可行走區域)。

#### 5. Enemy (繼承 Character)

繼承 *Character* 中的座標及表示符號的屬性。

*moveEnemy()*：以隨機方式在迷宮中移動敵人位置。

### 四、指定 C++ 技巧

#### 1. Inheritance

將道具 (*Bicycle, lover* 等) 都繼承 *Item*，因為他們都需要有  
自己的座標，並且有函數判斷物品與玩家位置是否相同、物品是  
否已被撿起、輸出道具符號等。

將 *Player* 與 *Enemy* 繼承 *Character*，因為這兩個 class 都一樣會有座標及顯示符號。

## 2. File I/O

使用 *ifstream* 開啟地圖、敵人、道具資料的檔案，並使用 *getline* 一次讀取一行文字檔案中的資訊。

## 3. Polymorphism

在 *Item* 中的 *getSymbol()* 為一種 virtual function，同時每個物品（如：*Bicycle*, *lover* 等）都有覆寫自己的 *getSymbol()*。在 Global.cpp 中也有使用 *Item\** 來輪流指向不同類型的物件。

# 肆、 困境與突破

## 一、 無法使用 windows.h

因為組內有三個人都是 Mac OS 系統，並不支援 windows.h 的功能，因此我們最後選擇捨棄 windows.h，單純使用終端機輸出，製作最原始的 terminal game。

## 二、 Mac OS 系統與 windows 系統某些情況下不相容

我們一開始預想玩家直接透過 W, A, S, D 控制角色（向大部分的電腦遊戲一樣），但玩家輸入完必須按下 ENTER 程式才會執行。因此我們上網學習了不要 ENTER 就可以終止輸入，卻發現這個解決方法只適用於 windows 系統。我們只好再另外開發一個 for Mac OS 的版本，讓遊戲在 Mac 上也可以正常執行。

## 三、 整合程式碼的瓶頸

因為沒有清晰定義好資料結構、命名規範等，一開始有遇到函數的使用時機及傳入參數無法匹配等情況，在初期整合程式碼時就需要再修改部分自己寫好的 class 來使成員們之間的程式碼可以相互串聯。之後再做類似專案時，應先確立、討論對外的公共函式、類別介面及格式，在整合階段就不會產生期望與實作不匹配的問題。

# 伍、 心得與反思

## 一、 蔡承佑(B13705037)

這次專題讓我第一次練習如何與他人一起寫程式，其中也面臨到了很多困難。首先，在寫 class 的時候不確定隊友會怎麼讀取或使

用自己的 class 所以不確定到底該以什麼資料型態來定義函數的參數，之後再整合時也發現很多部分需要重新寫起，彼此的程式才能串接。另外，也有自學到如何讓終端機直接讀取使用者輸入的字，讓玩家可以連續移動。以及藉由字串連接的方式變更欲讀取的檔案的名稱，讓系統可以順利的換到下一關的檔案。這個專題讓我更熟悉繼承及多型的使用方法，也讓我對團體寫程式有了基本的了解，之後在做類似的專題時，要先討論好變數命名的方式，並確立好函式的使用時機，以確保彼此的程式能快速整合。

## 二、陳允直(B13705054)

在得知專案的資訊後，我很快就對製作什麼樣的遊戲有了一些想法，並迅速與組員達成共識。因此，原本認為這次專案不會太困難，但後續實作過程才讓我發現，這個專案並非我想像中那麼簡單。最主要的問題是我們正式開始的時間比較晚，過程中總是覺得來不及，壓力也很大。合作過程中，我們對於分工的方式感到不知所措。由於之前從未有過程式合作的經驗，加上一些原因導致我們無法使用 GitHub，使得每個人完成自己負責的部分後，還需要再進行整合。這是我覺得最困難的部分，因為需要非常細心地閱讀大家的程式碼，再逐步將其加入主程式並進行修改。但也正因如此，我在這個過程中學到了許多，並且逐漸找回寫程式的樂趣。當看到遊戲成功執行，各項功能與機制都正常運作時，我的眼淚差點掉下來。另一個挑戰是整理和簡化程式碼。在遊戲最初完成時，程式碼非常雜亂冗長。整理和簡化的過程很痛苦也很繁瑣，甚至還可能出現以前沒有的 bug。但這些讓我更加了解如何寫出好的程式碼，也提升解決問題的能力。這次專案是一個有難度但又有趣的經驗，也實現了我曾經的目標——寫一款遊戲。感謝組員們的努力，他們都非常認真，還在我花大量時間整合時，協助處理其他工作，讓專案得以順利完成。

## 三、黃妍僖(B13705057)

這次的專案讓我認真地意識到我會的東西真的只是冰山一角，也感受到 C++ 的強大。以前我一直認為 C++ 能做的只是單純的終端機輸出及後端的處理，但在準備這次專案的過程中才發現，其實有很多辦法能做出一些前端的互動介面，不論是 windows.h 還是 SFML。雖然我們最後都沒有使用到，但這是我會想要好好學習的東西，因為我認為可以讓腦中的想法透過程式碼實現是件很開心的事，希望未來能夠學習更多處理前端的程式技巧，讓對於程式的構想能更具體的實現。此外，在做專案的過程中真的感受到上課小傑說的 OOP 的重要性。以前在寫作業時很常懷疑 class 之類的使用的必要性，但

在經歷過這種規模比較大的合作專案後，才發現 OOP 真的很重要，不論是整合大家的程式碼還是 debug 都簡潔很多，真的應該想辦法更熟悉該如何應用。最後，經過這次的專案後，身為資深果粉的我又開始猶豫是否該改用 windows 系統，兩系統間的整合實在是太常出問題了，而且好多功能都不支援 Mac OS ，真令人難過…

#### 四、林宇婕(B13705001)

這次期末專案從構思到實作，給我一個從零開始創造的感覺，深刻體會到程式的「設計」以及它的強大之處。一開始我們想使用 windows.h 來建立一個視覺化的視窗，後來因為系統相容性的問題而放棄改用 terminal game ，原本以為會無趣許多，沒想到在終端機也可以製作完整度很高的小遊戲，驚喜之餘，成就感也很足！再後來是多人合作的部分，除了程式碼的整合之外，也涉及到跨作業系統的問題，還因此開發了兩種版本讓我們在 mac OS 或 windows 都可以正常執行。在版本控制方面也遇到障礙，我們原本想利用 Github 來進行版控，但因我的帳號被標記的問題所以暫時無法使用，增加了一些整合上的難度，不過最後還是順利完成。這次的小專案雖然規模不大，但卻讓我學到了許多寶貴的經驗。不僅在技術層面上對程式設計有了更深的理解，也在團隊合作中體會到合作的重要性與挑戰性，是一個收穫良多的專案。

#### 陸、 分工表

蔡承佑	處理道具相關程式、書面報告
陳允直	整合程式、關卡切換、修改細節、擴充功能
黃妍僖	敵人隨機移動、簡報製作、影片剪輯
林宇婕	地圖輸出、設計地圖、進關畫面、簡報製作