

HELL ATTACK



B96902026 楊應甲

B96902062 鍾宇桓

目錄

製作靈感.....	3
遊戲簡介.....	4
實作過程.....	5
函式簡介.....	7
訪問工作人員.....	8
工作人員名單.....	9

製作靈感

我們製作這個遊戲的靈感來自於一個 flash game——BoxHead，這是一款射擊遊戲，玩家控制一位主角抵抗不斷來襲的殭屍，在不斷擊殺殭屍的過程中，隨著分數的提高，也會出現更多種類的槍給玩家使用，而殭屍也會越來越多提高關卡的難度。



BoxHead 關卡**永遠不會結束**，關卡會持續到主角死亡為止，玩家的目的是存活下來，擊殺最多的殭屍賺取最高的積分。

遊戲簡介

我們所製作的遊戲基本上主題與 BoxHead 類似，遊戲目的都是賺取最高的積分，不過不同的是，我們設計了兩種遊戲模式：

第一種是射擊模式，這種模式和 BoxHead 的遊戲方式相同，只不過我們設計的較為陽春，主角一開始在左下角，只有一把槍，殭屍則出現在左上跟右上角，殭屍死掉也會從這裡重生，只要主角被殭屍接觸到遊戲就會結束，玩家除了移動來躲避殭屍的攻擊外，也可以用槍擊殺殭屍來賺取積分。要注意的是主角和殭屍都不能穿過石頭，不過子彈可以從石頭上面飛過。



第二種是生存模式，我們設定這個模式是隱藏模式，要啟動的話必須在遊戲開始畫面時按 B，在這個模式下主角的彈藥已經用盡，只能不斷逃跑，生存的時間越久積分就越高。



實作過程

我們整個程式的製作大致上分成三個部分，

第一部份是要學習如何使用 HAM 來寫 GBA 遊戲，基本上是從零開始學習，搜尋網路上 HAM 的教學文件，我們找到一個有 12 天教程的英文網站及其簡體翻譯網站，將它翻譯成正體中文後開始努力吸收這些全新的知識。為了實做這個遊戲，我們學習如何使用 tileset 拼成整個背景、利用偵測按鍵切換不同的背景、顯示 sprite 並偵測按鍵使其移動、在特定條件成立時刪除該 sprite、顯示字幕……等等，接著便是將這些功能實際應用在遊戲製作。

第二部份是架構遊戲基礎，我們為開始背景和遊戲背景建立成兩個不同的調色盤，為所有 sprites 建立一個調色盤，建立一個二維陣列儲存地圖(1 表示可走/0 表示不能走)，建立各種功能函式(後面會有所有函式的介紹)，而主角和殭屍轉彎的時候會變換方向是利用同一張圖片不同開始指標所辦到的。每次偵測到按鍵被按下都會觸發一連串的函式:更改座標變數→更改代表方向的數字(上 0 右 1 下 2 左 3)→更新 sprite 圖片→實際更新 sprite 的座標內容。這些機制都在 `vbl_func()` 裡面執行，這個函式也就是整個遊戲的主幹。另外，”重新開始”是利用重新初始所有變數再呼叫 `main()` 函式達成，這樣很省 code 空間但可能造成資料 `allocate` 出問題，然而，這要在一次遊戲結束時的等待畫面按著空白鍵持續加速十秒鐘左右再按 `Enter` 重新開始才會出問題，我們認為一般玩家不會這樣做，因此就不再做修正了。

第三部分是殭屍 AI 的部分，這個地方我們嘗試了很久，可以說是整個製作過程中最困難的部分，我們的目標是讓殭屍繞過過障礙物朝主角前進，基本上前進並不難，

這和主角的移動是一樣的，困難的地方在於如何讓殭屍繞過障礙物，為此我們奮鬥了將近 12 個小時。我們先設定殭屍向主角的方位移動（如：右下），若兩者位置重疊就讓殭屍停止移動；如果在某個需要前進的方位（以下舉例為 x 向）有石頭，就先判斷主角的左或右是否有石頭擋住，有的話就未被擋住的方向走；沒有的話就判斷是否需要另一個方位的移動（y 向），若有需要就往那個方位移動；若沒有需要就判斷前方的石頭是否是兩顆排成一排，是的話就往沒有石頭的那個方向走，否的話就隨機選一邊走；另外如果在殭屍要前進的對角方向有石頭（如：右下方），而且和主角的 x 座標或 y 座標只差 1，就往 y 座標或 x 座標移動。最後是判斷殭屍和主角的 x 或 y 座標是否再同意條線上，如果是的話就直直往主角跑過去，不是的話就隨機選一個往主角的方向走。

函式簡介

void restart() — 讓遊戲重新啓動

void vbl_func() — 是整個遊戲執行的主幹

void query_buttons() — 用來判斷玩家所按的按鍵

void update_army_gfx() — 會更新主角移動的座標和圖像

void query_bullet_dir() — 會更新子彈座標並判斷是否打到殭屍

void update_bullet_pos() — 讓螢幕上的子彈確實移動到剛才更新的座標上

void query_ghost_walk(u8*) — 控制傳入的殭屍的移動並判斷是否碰到主角

int check_touch(int,int,int,int) — 會判斷傳入的座標是否相同，是則回傳 1，否則回傳 0；

int check_ghost_route(u8,u8,u8,u8) — 用來判斷殭屍該往哪邊走，也就是 A I 的部分，回傳 0、1、2、3 分別代表上、右、下、左。

訪問工作人員

問：你們製作時有碰到哪些瓶頸？

鍾：我主要就是 A I 的部分，測試的時候殭屍不是常常莫名其妙的亂跑，不然就是在石頭附近做簡諧運動，花了將近 1 2 個小時才讓它們乖乖的走。

楊：一開始一直弄不懂背景 tileset 的意義和使用方式，經過許多常識仍然會出現破裂的背景和不能顯示字幕等問題，而利用 comment line 處理背景和物件的轉檔問題也是一開始難以接受的概念，不過做到後面就越來越清楚了。

問：你們對這次的製作有什麼感覺？

鍾：因為我最近重感冒，一整天鼻子都是塞住的，身體狀況整個就很差，不過在遊戲做好的瞬間還是很高興的。

楊：第一次住系館，為了一個目標一起努力打拚的感覺很值得回味！

問：這個遊戲是否有 B U G ？

答：官方回答當然是沒有，不過事實上有一個是人和子彈會穿殭屍而過的 B U G ，至於其他的就有待玩家去發掘嚕。

問：這個遊戲有密技嗎？

答：為了遊戲的樂趣和平衡性，我們並沒有設計密技，不過有一個隱藏模式，就是在開始畫面時按 B 可以進入生存模式。

工作人員名單

企劃 楊應甲、鍾宇桓

主程式設計 楊應甲

A I 設計 鍾宇桓

美術設計 鍾宇桓

特別感謝

美術指導 小雞

熱情測試 拍拍、小雞、雞頭、阿衝