

# 期末專案書面報告

組別：8 經經蘆筍資



B09705023 魏芃文    B09705038 劉芷瑜  
T09705309 楊政璇    B07303120 蔡妤柔  
B07801008 吳謹好

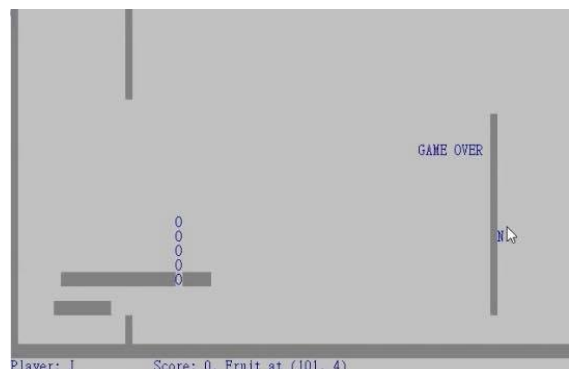
## □ 主題構思及遊戲玩法

我們的遊戲名稱叫做“Greedy Serpent”，是一款輕鬆又簡單的無限循環小遊戲。遊戲的控制非常的簡單，使用鍵盤上的按鍵↑，↓，←，→來避免蛇蛇撞到障礙物，並且在過程當中吃到正確的蘋果來獲得分數。

在遊戲一開始的時候，使用者可以選擇蛇的速度（從 1 最慢～9 最快）。當玩家累積到了一定的分數時，速度將會變快，並且會調整蛇蛇移動的延遲時間。隨著遊戲的進行，每次玩家吃到一個蘋果時，蛇的身體就會增加一個字母，但是，每次玩家吃到一個毒蘋果時，玩家的分數就會減少。當遊戲重新啟動時，屏幕上的障礙物都會重新的分佈在新的位置，並且每次屏幕刷新時，顏色模式會改變。這個遊戲一共有 7 種不同的顏色模式。一旦玩家不小心碰到障礙物或是自己的身體時，遊戲就會結束。玩家的最終目的是盡可能的長時間存活，得到最高的分數。



▲ 選擇蛇蛇的外表



▲ 蛇蛇撞到牆壁後遊戲就結束了



▲ 遊戲結束後選擇是否再玩一次



▲ 選擇蛇蛇的速度

## □ 安裝環境

首先，在執行遊戲之前，如果你是 PC 端電腦的使用者，只要在電腦的編輯器（例如：Dev C++）上運作我們的程式碼，就可以開啟遊戲。如果是 macOS 系統的使用者，就需要在電腦中下載 ncurses 和 homebrew 的 library。

- **Ncurses library**：這個 library 提供了 API（應用程式介面），可以允許程式設計師編寫獨立於終端的基於文字的使用者介面。
- **Homebrew library**：是一款自由及開放原始碼的軟體套件管理系統，用來簡化 macOS 系統上的軟體安裝過程。

#### 安裝方式：

- **Ncurses**：在終端 terminal 輸入：`apt-get install libncurses5-dev`
- **Homebrew**：在終端 terminal 輸入：`/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"`

## □ 系統設計

為了方便閱讀與 debug 的效率，我們將程式分成三個檔案，分別為：

1. **main**：包含 main function 裡面會呼叫到的函數以及呼叫的動作。
2. **header file**：class fSnakeFuntion（NewSnake）的宣告。
3. **cpp**：程式內部函數。

接著介紹程式是如何進行以及重要函數的介紹：

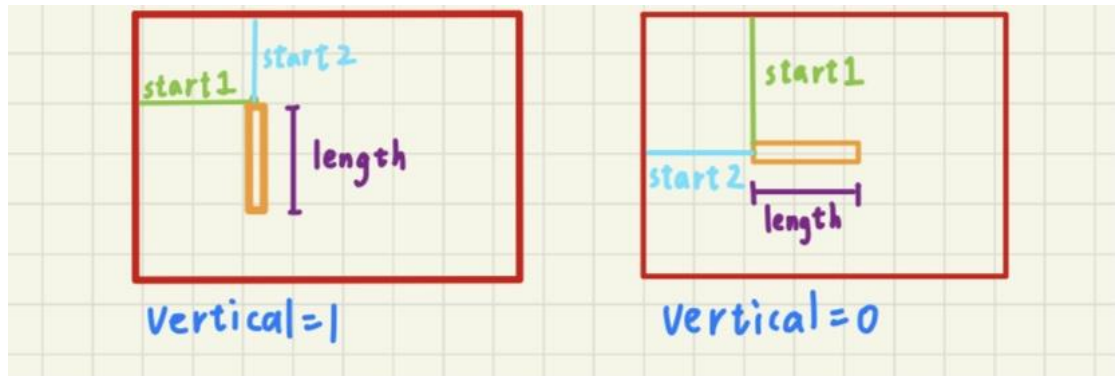
當接收到 isUserReady 等於 'y' 時，main 裡面會呼叫 chooseSpeed 這個函數，讓使用者輸入想要的速度（使用 ncurses 裡的 usleep，設定 Snake 移動的延遲時間）。接著呼叫 choosebody，讓使用者輸入一個字元成為 Snake 身體的樣式。然後呼叫 askUserName 函數請使用者輸入自己的姓名，最後呼叫 NewSnake 裡的 playGame 進入最終遊戲畫面。

playGame 函數裡做的事如下：

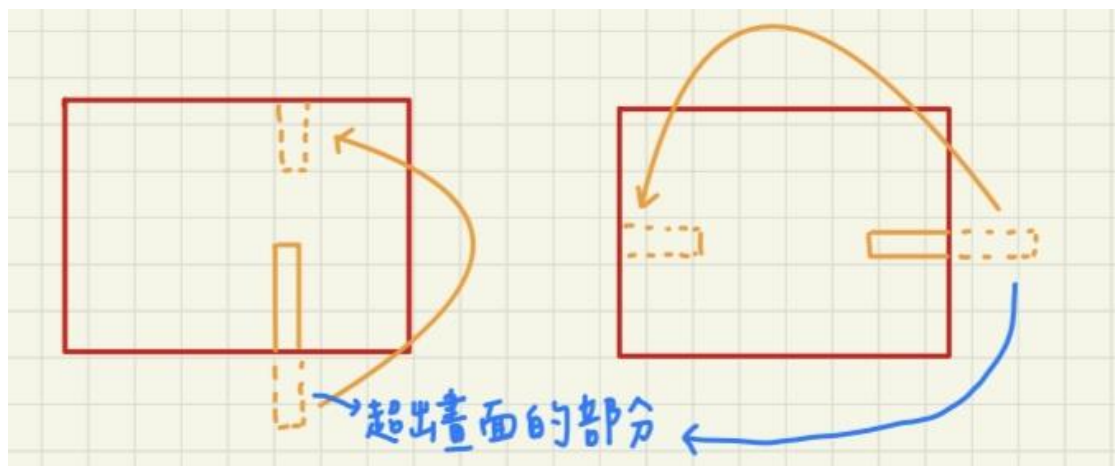
1. 呼叫 FatalCollision 以及 WallCollision 判斷是否撞到四周牆壁或是畫面中的牆壁，若撞到，則在畫面中顯示 GameOver，此次遊戲終止。
2. 呼叫 GetsFruit 函數，若 Snake 吃到 fruit（代表 Snake 的 x 坐標和 y 坐標與 fruit 一致），則分數加 10。
3. 呼叫 GetsBadFruit 函數，若 Snake 吃到 badFruit（代表 Snake 的 x 坐標和 y 坐標與 badFruit 一致），則分數減 10。

接著介紹 fruit 如何產生。在 GetsFruit 裡面會呼叫 PositionFruit，是用來隨機生成蘋果的位置。但是生成的位置是有限制的，所以要做判斷。限制為：1.不能在 Snake 的身上 2.不能超出 gameWindow 的範圍 3.不能生成在畫面中（垂直或水平）的牆壁上。若函數中隨機生成的 tmpx 與 tmpy 符合上述限制，則要重新生成，直到不滿足限制為止。badFruit 的生成與判斷則同上。

再來介紹隨機生成的 wall 是如何產生的。我們設了幾個變數，分別是 `vertical` (此為 `bool`，1 for vertical, 0 for horizontal)，`length` (牆壁的長度)，`start1` 和 `start2` (牆壁的起始位置)。這些都是使用 `rand` 隨機產生的。接著跑 `for` 迴圈畫出牆面，`window` 的畫法也是用 `for` 迴圈畫出來的。



▲ 水平及垂直的 wall 分別的 `start1` 和 `start2`。



▲ 當隨機生成的牆壁超出介面，則向上或向左遞補多餘的牆壁。

而 wall 碰撞的偵測，是呼叫 `WallCollision` 函數，跑 `for` 迴圈判斷 Snake 的 `x` 和 `y` 座標是否和牆壁重疊，若重疊則 `return true`，宣告遊戲失敗。

在遊戲介面中，我們還設計了 7 種顏色主題增加使用者的樂趣。這是寫在 `clearCentre` 這個函數裡面，並且使用 `ncurses` 裡面的 `start_color` 和 `init_pair` 功能。這 7 種主題是我們先選好搭配 (兩個顏色為一組) 再使用 `rand` 隨機產生其中一組。因為在 `chooseSpeed`、`choosebody`、`askUserName` 裡都會呼叫到 `clearCentre`，所以每呼叫一次都會改變一次顏色，帶給使用者繽紛的遊戲介面。

```

45     start_color();
46     init_pair(1, COLOR_RED, COLOR_GREEN);
47     init_pair(2, COLOR_BLACK, COLOR_WHITE);
48     init_pair(3, COLOR_BLUE, COLOR_YELLOW);
49     init_pair(4, COLOR_GREEN, COLOR_YELLOW);
50     init_pair(5, COLOR_YELLOW, COLOR_RED);
51     init_pair(6, COLOR_MAGENTA, COLOR_CYAN);
52     init_pair(7, COLOR_CYAN, COLOR_MAGENTA);
53     //attron(COLOR_PAIR(1);
54     int color = rand() % 7 + 1;
55     wbkgd(stdscr, COLOR_PAIR(color));

```

我們的程式有用到 **Ncurses library** 裡面的一些函數，這邊是要簡短的說明一下其中一些函數的用途。

- **refresh()**: 當你呼叫螢幕輸出函式，要對螢幕上的東西做改動時，**ncurses** 並不會直接改變，而是要呼叫 **refresh()**後，才會一次將所有改動全部改到螢幕上。
- **move()**: 控制游標的函式。
- **echo()/noecho()**: 控制從鍵盤輸入字元時是否要將字元顯示在終端機上，系統預設是開啟的。
- **getstr(level)**: 從鍵盤上讀取字元。
- **endwin()**: 呼叫此函數來關閉 **curses** 模式。
- **clear()**: 清除整個螢幕。

## □ 未來改善

我們發現這個程式是有 **bug** 的，雖然發生的機率不高，但是若可以改善以下這些 **bug** 會讓程式更好：

1. 當牆壁的生成圍成一個矩形時，**fruit** 和 **badFruit** 不可以產生在矩形裡面，不然 **Snake** 永遠都吃不到。
2. 若一開始 **Snake** 的位置就在 **wall** 上，則遊戲一開始就會 **game over**，所以應該做判斷，牆壁不可以生成在 **Snake** 的起始位置。

希望新增的功能：

1. 介面：玩家積分表
2. 使用者選擇背景音樂
3. 蘋果的位置並非 **random**，可以改變成 **distribution**，例如難度越高的話，蘋果會越靠近牆角

另外，我們大家一致的認為說這一次我們最大的缺陷是分工合作，我們認為在開始所有事情之前，我們應該要先計畫好我們在遊戲裡面要增加一些什麼樣的

功能，再去實際的寫出程式。以這一次的例子來講，我們是邊寫邊想，然後有一些組員根本不知道程式寫到了什麼地步。所以我們認為，如果在一開始我們就坐下來一起討論說我們在這個遊戲要增加什麼樣的功能，才能更好的去分配說誰寫障礙物、蘋果、速度等功能。

## □ 分工方式

組員	工作內容
魏芃文	系統設計、書面報告、拍攝影片
劉芷瑜	書面報告、拍攝影片
楊政璇	書面報告、拍攝影片
蔡妤柔	系統設計、上台報告
吳謹妤	系統設計、上台報告

## □ 心得感想

楊政璇：

For this final project, we have welcomed a new member to participate with us. In the beginning we decided that we wanted to recreate the dino game from google chrome. However, as we encountered more and more problems such as using mac and PC, we decided that we were going to change our game to the snake game. Throughout the process of dividing work among us all, we all had different ideas. In the beginning we decided that 2 people would work on coding and the rest of us would each work on one item for 書面報告，上台演講，and 拍攝影片. But things really did not go the way we thought it would go. We did start discussing on our project as soon as the professor announced, however, because of the lack of communication between members, there were some arguments that has arose during this project. In the course of this project, I kind of took on the role of forcing people to produce work, forcing people to hand in work on a specific day, communicating in between, and basically just being the mean one. I also participated in the description of introducing the game plus presenting to parents. I am very thankful that in the end, everything was settled and things did go the way we wanted it to go. But if we were to work on a similar project for next time, I believe that we could have distributed the work a different way.

### 劉芷瑜：

經過了這次的期末專案，我發現要從無到有製作出一個簡單的小遊戲並沒有想像中那麼容易，不只需要有足夠的程式基礎，組員之間的默契也是十分的重要，即使我們組在剛開始的討論階段有一些意見不合的地方，有造成一些不愉快，但是幸好最後大家也是十分的配合，有共識要一起做好這次的專案，結果也是出乎意料的好。我們組別這次寫的小遊戲是貪食蛇，有參考一些網路上的資源，並且做了一些改動和改良，讓這個遊戲不要那麼單調乏味，雖然我不是負責設計演算法的人，但是對於組員做出的成果還是非常有成就感的，現在只要我看到遊戲或是應用程式等等的相關資訊軟體，我都會思考這些程式設計者要怎麼去寫演算法，也是加深了我對程式設計這塊領域的興趣和投入動機。

### 魏芃文：

這是我第一次接觸寫遊戲，我覺得對於大家來說都是很大的挑戰。要去學習以前沒有學過的 `library`、學習如何使用 `github` 等等，更重要的是一學習如何跟組員溝通。我覺得我們這組剛開始因為溝通上出了點問題，所以彼此都有些誤會，也讓組內的氣氛不是很好，還好後來我們還是成功完成了這份專案，讓我們都鬆了一口氣。我覺得對於寫遊戲來說，`rand` 是一個很好的功能，因為很多東西都是要隨機產生，例如蘋果的位置、牆的生成等等，這次讓我更熟悉了「隨機」的運用。最後，我覺得這個專案是能逼自己在短時間內進步的一份作業，很高興我們大家都有所成長。

### 蔡妤柔：

這次的期末專案不只增強在程式方面的能力，也體會到許多事，從不太會看 `library` 的程度，到可以擴充遊戲、`ncurses` 跟 `function` 的功能，在開發的過程中也體會到遊戲設計師的樂趣，雖然之前也做過 `Python` 的期末專案，但這次用 `C++` 寫貪吃蛇，難度增加不少，除了要避免 `inconsistency`、還要讓組員知道妳在寫什麼，甚至改了一晚上 `bug` 還沒改完，但能把課程所學應用在遊戲上，也發現了不少興趣。這次的分組謝謝謹妤跟芃文的幫忙，我發現開發程式在討論過程中就會有更有趣且有效的想法，也謝謝大家在短時間內的高效產出，除了終於了解 `github` 的 `branch` 該怎麼用之外，也 `get` 快速製作簡報、上台發表的經驗，在大家期末最忙碌的時候順利完成專案，大家辛苦了！

### 吳謹妤：

這次的產出期末 `project` 的過程沒有想像中的順利，其中最印象深刻的是發現原來 `C++` 許多 `library` 有很多與 `mac os` 系統不相容，導致我們得在中途改變我們的專案方向，換成做改良版的貪食蛇小遊戲。也幸好後來在妤柔和芃文的幫助之下，順利把陽春版的貪食蛇加入了許多我們自己的元素，能有這樣的成品真的十分開心，也很慶幸我們沒有被那些困難打倒。上學期修商管程的時候，

我們這一組也是做小遊戲，但那次在其他超罩組員的 **cover** 之下，身為初學者的我，程式方面僅被分配到負責一些簡單的部分而已，因此沒能完全搞懂我們到底是怎麼生出整個遊戲的。這學期在程式設計這門課中，我的角色則變成有點像是主導專案的進行（程式的部分），因為這個原因，我得逼自己去理解一個小遊戲背後程式語言的邏輯（例如 **file** 的架構、誰呼叫誰...等等）、**ncurse** 的使用以及學會用 **Github**，這幾件事是在這次期末專案裡，我學習到最多的地方。最後，謝謝組員們的包容與幫助，才能做出這樣子的成品，大家都辛苦了：）