計算機程式設計作業二

——貪吃蛇

物理一 章舒涵 B09202014

一、題目動機

在函式的講義裡教了我們做出類似動畫效果讓球做自由落體，我就想是不是可以利用類似的方法做出會動的小遊戲，又想到了小時候在手機裡玩過的貪吃蛇遊戲，主要就是按上下左右不會太複雜，因此就決定利用 C#嘗試做出貪吃蛇。

二、構想解說

貪吃蛇主要可以分為以下步驟：

(一)開啟新遊戲：重新畫地圖、隨機產生蛇以及食物的初始位置

(二)移動：重畫蛇

玩家按下鍵盤上下左右欲改變方向：控制方向參數改變，蛇轉彎

玩家未按鍵盤：控制方向參數不變，蛇轉彎，蛇直走

(三)吃到食物得分→蛇長度變長、在隨機位置再畫出新的食物

(四)頭碰到牆壁及身體→遊戲結束

玩家按下 Enter：回到(一)，開啟新遊戲

玩家按下 esc：程式關閉

三、程式測試規劃及流程圖

(一)開啟新遊戲：(合為一函式 newGame)

1.將所需的一些變數歸回初始值：

(1)蛇的長度 length(設為 4 節，也就是會長###<)

(2)蛇的行徑方向(設定為向右)

(3)儲存蛇位置座標的 500x2 的陣列 snakePosition

(清空原本的紀錄)

註：snakePosition[i,0]表示第 i 節的 y 座標

snakePosition[i,1]表示第 i 節的 x 座標

(4)分數歸零

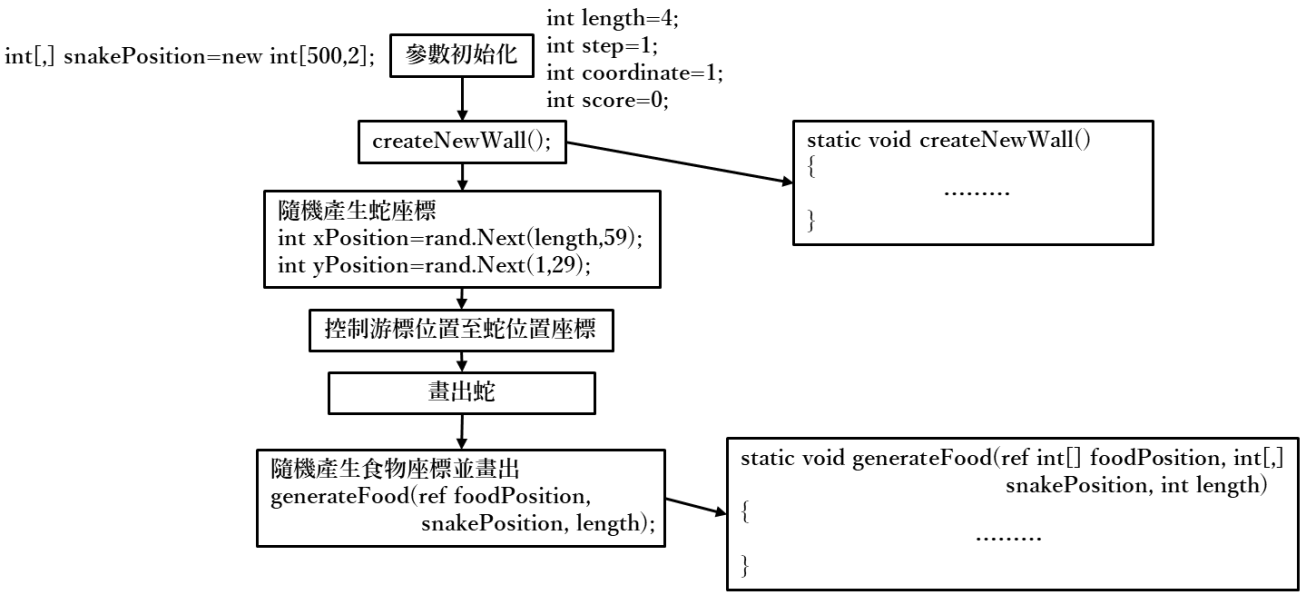
2.利用函式 createNewWall 畫出新的地圖(包括牆壁及中間的空間)

3.隨機產生蛇的頭一開始的位置座標，頭後面每節的位置便朝左遞減

4.先控制游標位置後，畫出蛇

5.利用函式 generateFood 隨機產生食物位置座標並畫出

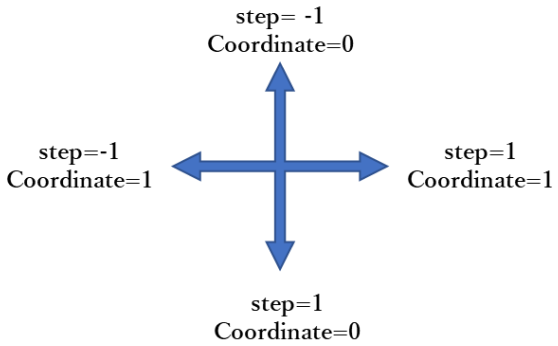
以下為流程圖：



(二)移動：

1. 玩家改變方向(按上下左右)

利用函式 **snakeDirection** 以及兩變數 step、coordinate 決定蛇的行進方向，以利後面更改下一個時間單位蛇的位置座標。Coordinate 為 0 或 1 代表待會蛇是往 x 方向或 y 方向走(coordinate=0→y 方向;);step 為-1 或 1，由 coordinate 決定是走 x(或 y)方向後，再由 step 決定是朝 x(或 y)的哪邊走，因此玩家按下上下左右後，可以分別對到以下四種情形：

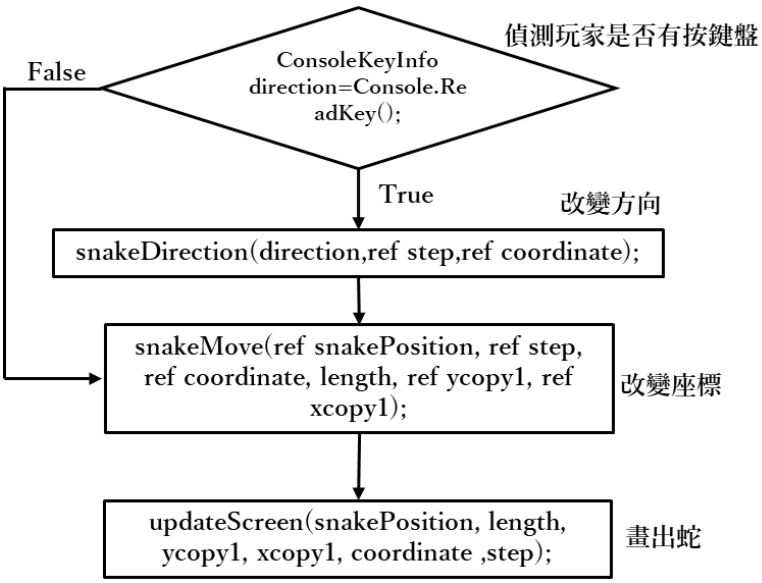


2. 更新蛇的位置座標

有了 step 和 coordinate 後，利用函式 **snakeMove** 更新蛇頭的新位置，也就是向 coordinate 的方向移動 step 步(這就是為甚麼 step 是 1 和-1)。接著再用 for 迴圈將頭以後的第 i 節更新為原本第 i-1 節的位置)，便能完成座標陣列 snakePosition 的更新。

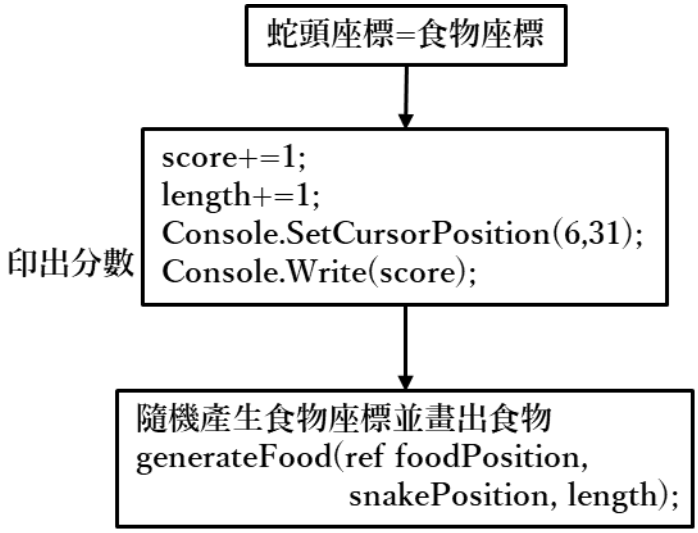
3. 畫蛇

利用函式 **updateScreen** 畫出蛇，將游標設定至位置(snakePosition[0,0], snakePosition[0,1])，再 write 出(<、^、>、v)，代表頭，試方向決定是哪種。身體的部分則是用#。



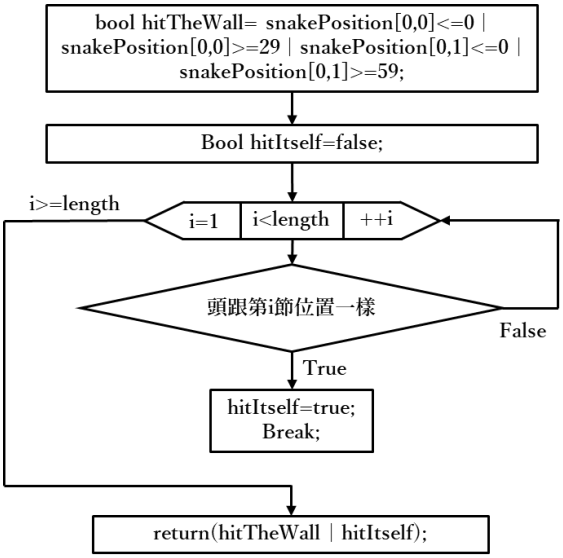
(三)吃到食物得分

1. 判斷蛇是否吃到食物(蛇頭座標=食物座標)，若有，分數及蛇長度接加一，並印出分數。
2. 食物被吃掉後，再一次利用函式 generateFood 新增新的食物。

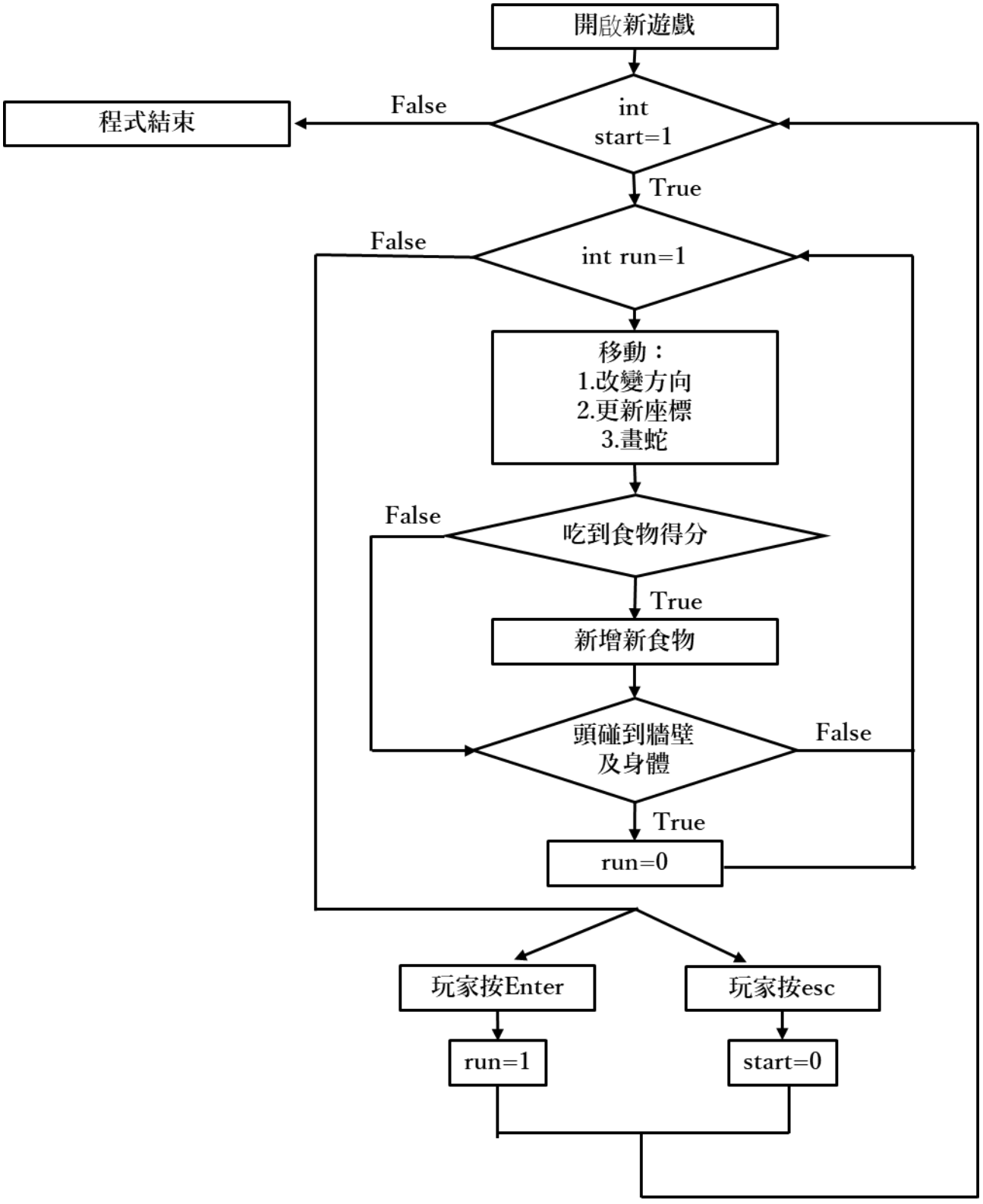


(四)頭碰到牆壁及身體→遊戲結束

利用函式 die，判別頭是否有碰到牆壁或是身體，如果有碰到牆壁，變數 hitTheWall=True，如果有碰到身體，變數 hitItself=True，最後回傳(hitTheWall | hitItself)，也就是只要有一個是 True 就會回傳 True。



（五）主程式流程圖

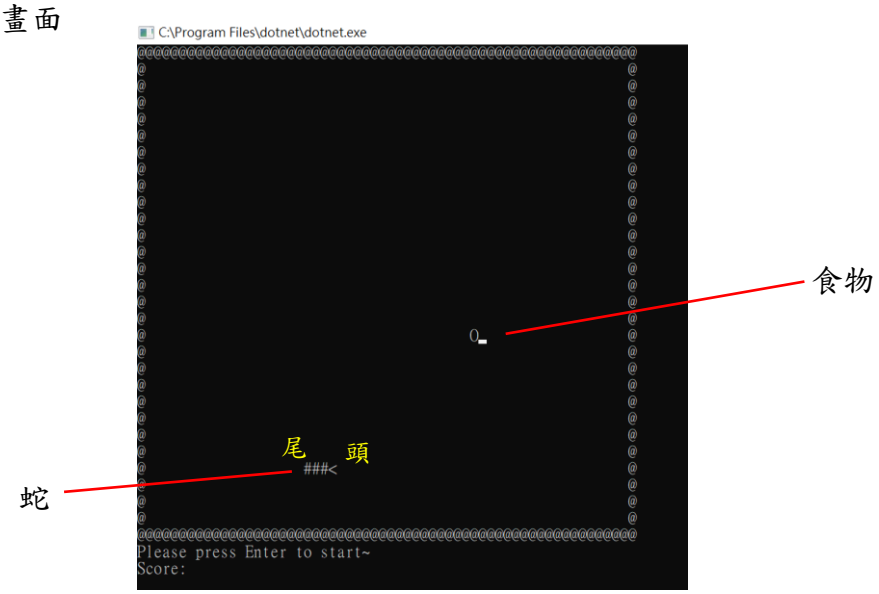

開啟新遊戲
int
start=1
False
程式結束
True
int run=1
False
移動：
1.改變方向
2.更新座標
3.畫蛇
吃到食物得分
False
True
新增新食物
頭碰到牆壁
及身體
False
True
run=0
玩家按Enter
玩家按esc
run=1
start=0

四、程式列表

1. snakePosition=new int[500,2];
2. Random rand = new Random();
3. Console.SetCursorPosition
4. Switch
5. bool
6. static void createNewWall()
7. static void newGame(ref int[,] snakePosition,ref int length, ref int score, ref int coordinate, ref int step,int[] foodPosition)
8. static void updateScreen(int[,] snakePosition,int length, int ycopy1, int xcopy1, int coordinate, int step)
9. static void snakeDirection(ConsoleKeyInfo direction, ref int step, ref int coordinate)
10. static void snakeMove(ref int[,] snakePosition, ref int step, ref int coordinate, int length, ref int ycopy1, ref int xcopy1)
11. static bool die(int[,] snakePosition,int length)
12. static void generateFood(ref int[] foodPosition, int[,] snakePosition, int length)

五、程式測試執行結果

(一)初始畫面



(二)玩到一半

過程都非常的順，也沒有閃頻的問題，分數也有正確計算

(三)遊戲結束後

遊戲結束後會出現

Press Enter to Restart 以及

Press esc to escape

按 Enter 後畫面便會重新整理變成新的一局

(五)總結

利用 C#即使不用視窗程式，也能做出簡單的貪吃蛇遊戲，遊戲過程中也都很順，雖然不如一班貪吃蛇依樣美觀，不過就功能來講都體驗的到。

六、參考文獻

1. [C#] C# 中 Sleep() 的作用及用法
   https://awei791129.pixnet.net/blog/post/24432898
2. counterpart of kbhit in C#
   https://bobobobo.wordpress.com/2008/11/03/counterpart-of-kbhit-in-c/

3.Microsoft (C# 程式設計手冊)
   https://docs.microsoft.com/zh-tw/dotnet/csharp/programming-guide/arrays/jagged-arrays
3. Console.ReadKey 方法
   https://docs.microsoft.com/zh-tw/dotnet/api/system.console.readkey?view=net-5.0
4. ConsoleKey 列舉
   https://docs.microsoft.com/zh-tw/dotnet/api/system.consolekey?view=net-5.0