

期末專案

組別：A

B10b02019 心理二 方唯蘋

B11702021 會計一 郭柏軒

B09302142 政治三 許瀚宇

B08507029 材料四 張瑋哲

B10703027 財金二 董豐廷

一、專案主題

本次專案是利用C++做一個小遊戲的集合，透過開源的套件SFML[2]，以及網路上的開源小遊戲[1]，我們對遊戲進行了遊戲性優化、統一美術風格，將各個遊戲分別處理完後，再寫一個介面進行遊戲的選取以及死亡畫面的呈現，藉此完成一個完整的小遊戲集合。

二、系統設計

2.1 主畫面

主畫面我們選擇封裝成class，並透過sfml的功能讀取圖檔，在主畫面中我們實現的功能主要有：

1. 遊戲選擇:

在主畫面中按下按鍵，會跳出主畫面的class，並進入各個小遊戲中。按下X、Y、Z鍵會分別進入Doodle jump、Minesweeper、Arkanoid。

2. 音樂:

在主畫面中我們利用 sfml 的 music 模組，只要在選單頁面便會撥放音樂。

2.2 Doodle jump

在我們參考的 Doodle jump 中，由於遊戲運行方面的 bug 較少，我們主要專注在遊戲功能的擴充。針對Doodle jump的遊戲，我們共對以下功能進行新增及優化

1. 新增死亡機制：

由於我們所參考的 Doodle Jump 並沒有死亡機制的設置，導致遊戲無法按照正常程序結束。因此，我們在遊戲系統中加入了結束遊戲一系列的條件判斷，讓玩家能夠擁有正常、完整的遊玩體驗。

2. 新增角色選擇：

一般網路上的 Doodle Jump 並沒有提供選擇角色的功能。為了加強玩家的遊戲體驗，我們利用 sfml pollevent 的功能，讓玩家能在進入遊戲之前選擇自己心怡的角色。此外，我們也加入了角色預覽的功能，讓玩家們在確定自己的角色之前，預覽所有角色在遊戲中所呈現的樣貌。我們相信藉由角色預覽的設計，能夠讓玩家們更方便地選出自己想要的角色。

3. 優化分數計算：

由於我們所參考的 Doodle Jump 並沒有將分數的計算機制設定好，導致玩家們就算沒有達成正確的分數條件也能得到分數。像是在原本的遊戲中，玩家就算只是

在原地跳躍也能夠得到分數。因此，我們在遊戲系統中加入了新的分數條件判斷，讓玩家能夠正常地計算自己的分數。

4. 新增雙人模式供玩家選擇：

為了讓玩家們能夠與好友一同享受遊戲的樂趣，我們透過視窗切割、新增第二個 sprite、增加的條件式等方法，實現遊戲的雙人模式。玩家會在選擇該遊戲之後進入到模式選擇的介面，並且透過鍵盤按鍵來選擇遊玩模式。在雙人模式中，我們對原先的角色選擇程式碼進行了改良，讓第二為玩家也能選到自己心怡的角色。此外，我們也對原先在單人模式中的角色進行顏色的區分，讓第二位玩家更好地分辨自己的角色。

5. 新增音效：

透過 sfml 套件置入音效，讓系統能夠提供更好的使用者體驗。

2.3 Minesweeper

由於我們所參考的 Minesweeper 存在許多遊戲運行方面的 bug，因此我們主要努力的部分是在將遊戲的功能完善，讓它能夠順利地進行。針對 Minesweeper 的遊戲，我們共對以下四個功能進行新增及優化遊戲機制：

1. 正常的踩地雷遊戲中，往往會透過一些功能讓遊戲能夠更完整地進行。而我們組別選擇擴充的功能是周圍的空格展開，也就是讓踩到空格的玩家能夠將該空格外格的空格翻開。這樣設計的目的除了能夠讓遊戲更快速地進行、減少玩家前期踩到地雷的機會外，也能讓玩家更好地發揮出自己的實力。

2. 新增剩餘地雷的數量顯示：

為了讓玩家能夠順利地進行遊玩，我們透過 sfml font & text 的功能，讓玩家能夠隨時知道自己還有多少個炸彈需要標記。

3. 優化計分功能：

在正常的踩地雷遊戲中，玩家不應該在遊戲結束前知道自己的分數。因此，我們將計分顯示的功能關閉，讓玩家在觸發遊戲終止條件之後才能看見自己的分數。

4. 新增不同難度：

踩地雷遊戲中，很常導入不同地圖大小的功能，讓玩家自行選擇遊玩的難度，因此我們也加入此功能，讓玩家可以自由選擇簡單的10*10或是較為困難的18*18地圖大小。

2.4 Arkanoid

針對 Arkanoid 的遊戲，我們共對以下個功能進行新增及優化：

1. 新增用空白鍵開始遊戲的功能:

因為最一開始的版本遊戲中，當進入遊戲時，球會直接向下跑，玩家必須很快地反應過來，並用平板去接球。但經過多次調整後，我們將發球的位置放在平板上，並且新增了用空白鍵開始讓球移動，給玩家充足的反應時間。

2. 優化球速問題:

除此之外，我們也注意到初始球速過快的現象，因此亦修改了球速，由慢到快，避免了遊戲一開始就因球速太快而接不到球。

3. 新增分數計算:

在遊戲中同樣加入分數計算功能，只要消掉一個磚塊，玩家便會得到分。

2.5 死亡畫面

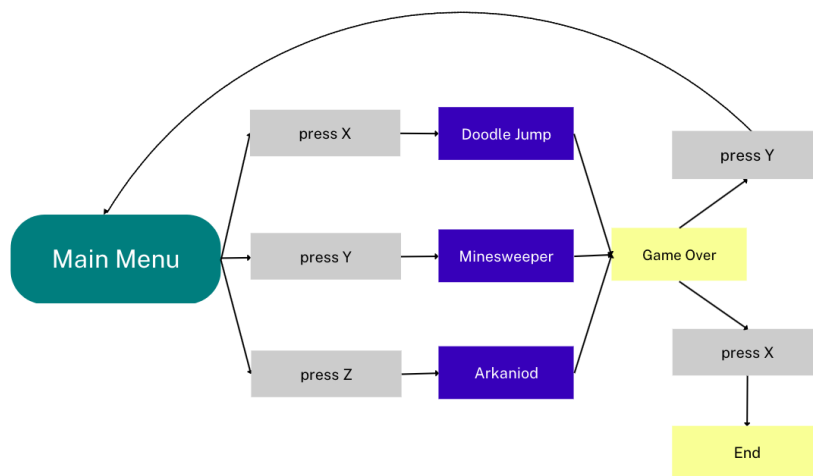
1. 遊戲結束後返回或退出

如同大多數的遊戲系統，在玩家死亡後，視窗將會切換成Game Over的畫面，我們設計利用按Y或X鍵，讓玩家能夠選擇要繼續遊玩（回到主畫面選擇遊戲），或關閉遊戲（程式關閉），兩個選擇。

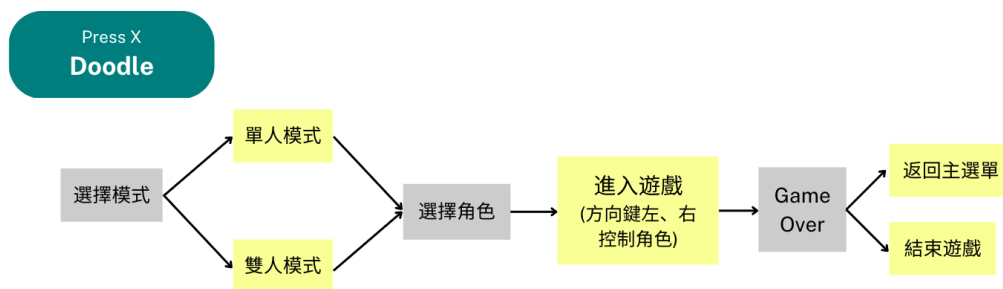
三、演算法

系統流程圖

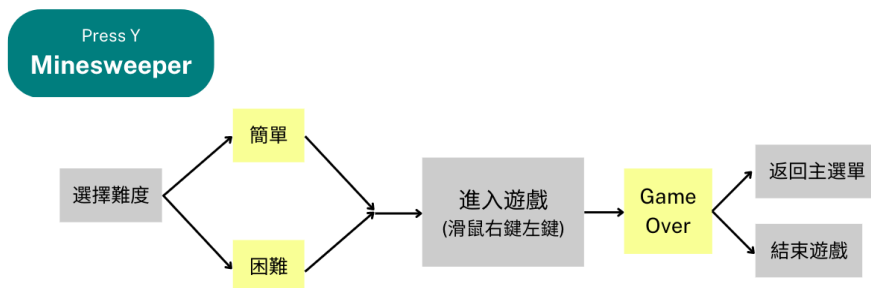
(1) 主選單



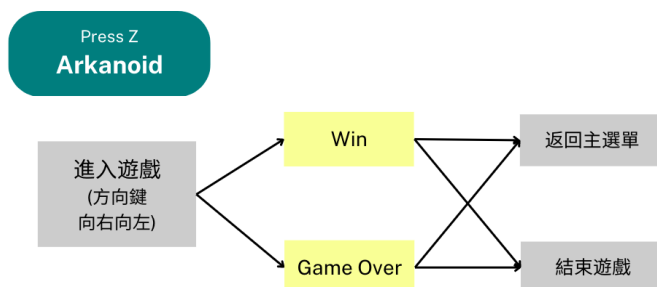
(2) Doodlejump



(3) Minesweeper



(4) Arkaniod



主要函數

3.1 使用到的函式庫與命名空間

```
#include <SFML/Graphics.hpp>
#include <SFML/Audio.hpp>
#include <time.h>
#include <iostream>
#include <string>
using namespace sf;
using namespace std;
```

3.2 將三個遊戲以 class 的方式進行封裝

```
int score = 0; // additional function: global variable to record score
Font font;
sf::Text score_text; // score showing
void game(); // function that runs the whole program
void dead(); // additional function: death function and interface
```

```
class Arkanoid
{
public:
    // some functions

private:
    // some textures and functions
};
```

```
class Minesweeper
{
public:
    // some functions and variables
    void check_blank(int, int, int); // additional function: check
                                     // surrounding blanks

private:
    // some functions and variables
    Font font;
    sf::Text score_text; // optimization: record score and hide
    Font font2;
    sf::Text remain_text; // additional function: show the
                           // number of remain bombs
};
```

```
class Doodle
{
public:
    class DoodleForOne // class for single-player mode
    {
    public:
        void processInputOne(); // process for single-player mode

    private:
        // some textures and variables
        string character;
        void selectForOne(); // additional function: select character
        sf::SoundBuffer buffer;
        sf::Sound sound; // additional function: music & sound effect
    };

    class DoodleForTwo // class for two-player mode
    {
    public:
        void processInputTwo(); // process for two-player mode

    private:
        // some textures, variables, music & sound effect
        string character1, character2;
        void selectForTwo1();
        void selectForTwo2(); // character select for two players
    };

private:
    void selectMode(); // additional function: mode select
};
```

3.3 game() 函數：開啟視窗，並依照玩家指示開啟對應的遊戲

```
void game()
{
    open a window called "Final project"
    load background picture and turn it into a sprite
    load and play bgm

    while (window is open)
    {
        declare an event
        while (event is working)
        {
            if (event.type == close event)
                close the window
            else if (event.type == pressing keyboard)
                if (pressed X, Y, Z)
                {
                    stop bgm and close the window
                    X: declare an Doodle type object and run it
                    Y: declare an Minesweeper type object and run it
                    Z: declare an Arkanoid type object and run it
                }
        }
    }
    display the window and draw the sprite (background picture)
}
```

3.3 dead() 函數

```
void dead()
{
    open a window called "Dead!"
    load background picture and turn it into a sprite
    load score and draw it
    while (window is open)
    {
        draw background
        declare an event
        while (event is working)
        {
            if (event.type == close event)
                close the window
            else if (event.type == pressing keyboard)
                if (pressed X)
                    close the window
                else if (pressed Y)
                {
                    close the window
                    rerun the game() function
                }
        }
    }
    display the window and draw the sprite (background picture)
}
```

四、分工表

方唯蘋：上台簡報製作、上台報告、系統UI設計、踩地雷難度、期末專案報告製作

郭柏軒：上台簡報製作、上台報告、踩地雷優化（空白處展開）、

許瀚宇：上台簡報製作、上台報告、雙親展示影片、

張瑋哲：上台簡報製作、上台報告、系統架構（遊戲起始、結束畫面）、主要程式撰寫、期末專案報告製作

董豐廷：上台簡報製作、上台報告、Doodle Jump雙人模式、計算剩餘地雷數、系統UI設計、期末專案報告製作

五、成員心得

方唯蘋:

這次專案的過程可以說是一波三折，因為Mac的系統很難安裝Sfml，所以有好一段時間都在嘗試各式各樣的方法，也到課堂上和教授、助教討論了好一陣子，後來還是有些困難，所以在專案中我負責的工作比較不是寫程式的部分，後來成功運行之後也勉強能寫一點。

我覺得這次專案因為時間急迫，加上開放式的題目，給大家一個需要自學的機會。看著我們組做的小遊戲系統，從最一開始討論題材，架設系統，把一個一個小遊戲變得更加完整，隨著時間我們的專案也持續的在進步，這種感覺非常的微妙，雖然系統還有很多增進的地方，但看到大家一起做出來的成果就覺得很開心。

最後我想感謝組員們都非常的認真負責，還有教授、助教們的協助，讓我在這門課學到了非常紮實又有趣的技能。

郭柏軒:

時間緊湊的期末專案是一個非常有難度的作業，需要統整程式設計課程教過的觀念，還需自主學習從未接觸過的新程式庫與語法，光是理解就需花上不少時間。

我們選擇參考網路上開源的陽春小遊戲之程式碼，並使用SFML將其包裝成遊戲集合的程式，包含主選單、單雙人模式、計分、及程式、介面優化等等。我非常感謝組員們提供的各式想法，並且花了不少時間在專案的製作，也謝謝教授給我們這個摸索程式寶庫的機會，琢磨自主學習的技能。

許瀚宇:

因為時間過於緊湊的緣故，我認為非常地愧疚，因為並沒有在專案貢獻上很多，也很佩服我的組員能夠有效率地完成大部分，我只能選擇很零碎的部分努力；編碼的過程中需要匯集過去所學的語法以及程式架構，最大的感受莫過於佩服網路上的資源，因為很多的東西都需要在網路上看著前人的成就，再將其內化或改編為自己的程式。只能說程式是一門很花時間但是成就感很顯著的專業。

張瑋哲:

這次期末專案從一開始就遇到蠻多困難，尤其是sfml環境的安裝，就花上我們好一段時間。安裝完後寫程式的過程反而較為順利，說明了經過一學期的學習大家都有所成長。如同期中專案，在專案開發過程中，最難的還是如何分工起來才會最有效率，我們選擇的方式是將遊戲分別封裝成class，如此一來不但方便分工，也較容易debug，看著原本很簡陋的遊戲逐漸完整，內心充滿了成就感。

經過一學期的教學，再加上期中、期末兩次專案實際完成任務的經驗，讓我對自己的能力更加有自信，也更加堅定自己在程式語言這條路上學習的決心，相信這學期打下的深厚基礎，是未來修更進階課程時的重要養分。

董豐廷:

由於自己是用mac來寫程式，所以幾乎花了一半時間在處理sfml的安裝問題，也跑了兩次的助教課，最後還是因為讀不進圖檔沒有成功，只能透過隊友的電腦來跑程式。但由於sfml的函式庫與event架構還算直觀，所以還是可以寫一些能幫助到團隊的函數。也由於花了許多時間在網路上尋找安裝sfml的方法，我也了解到了網路上各種資源的方便性。從原來xcode上有十幾個bug，透過網路上的各種方法，到最後也只剩一個。

這次的期末專案讓我學到了許多編寫遊戲的技巧，在擴充與優化程式的過程也十分有趣，也感謝隊友在我能幫上的忙有限的情況下完成了大部分的工作。

六、參考資料

[1] <https://github.com/suraj95/16-Games-in-SFML>

[2] <https://www.sfml-dev.org/index.php>

註: 功能展示影片連結如下: https://drive.google.com/file/d/1H3brJ4TA6uYQJcU7wEFnRjLMAG_Dxdw/view?usp=sharing

另外影片也有附在資料夾中