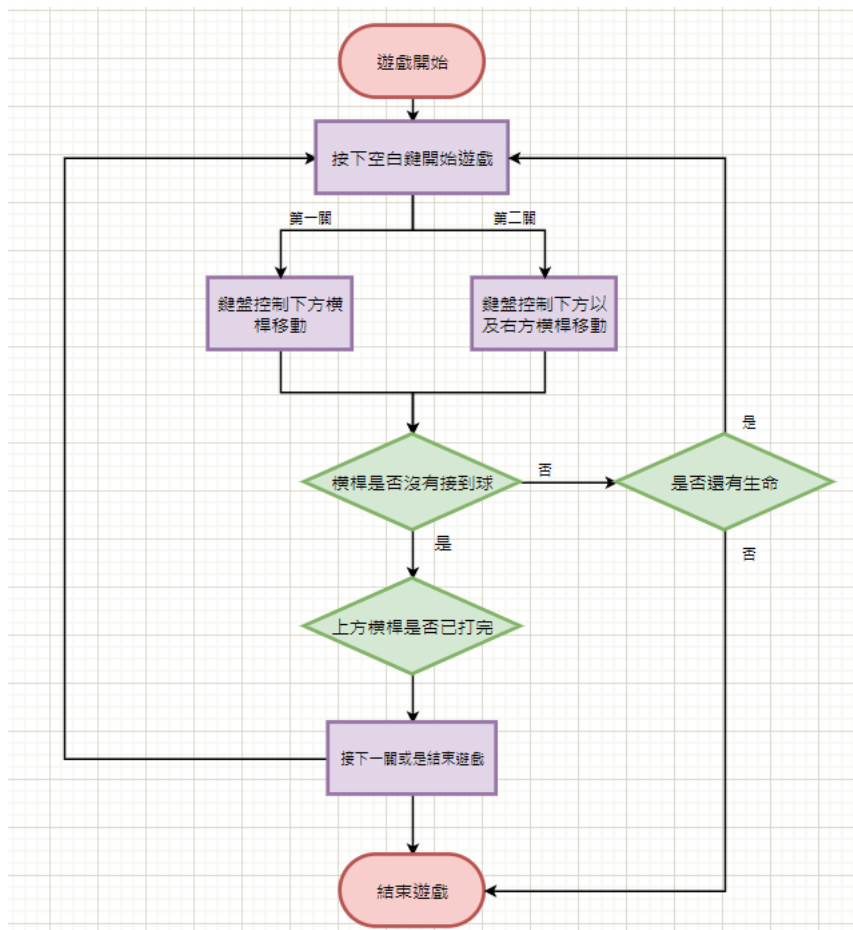


# 打磚塊遊戲

班級：資工四乙 學號：406262333 姓名：吳佩臻

## 一、 程式架構



## 二、 討論

除了最基本的下方有移動板以及球彈跳之外，為了提升遊戲困難度，於第二關時增加了右方的移動板。與同學討論後決定增加右下方的生命提示，方便玩家了解目前的生命狀況。

### 三、 執行畫面

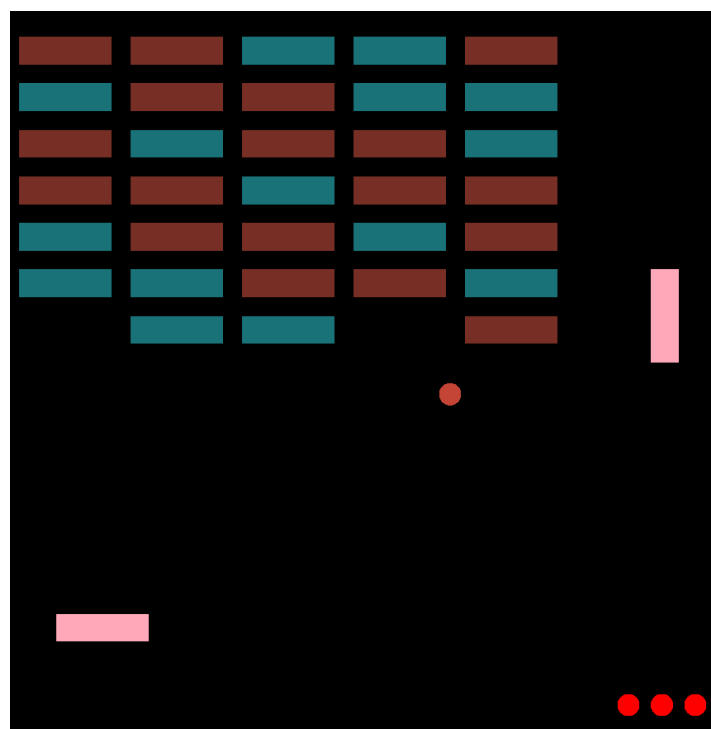
第一關：

第一關只有下方的移動橫桿。右下方有生命提示。



第二關：

第一關下方以及右方都有可移動的橫桿。右下方有生命提示。



## 四、 程式碼

### 1. 產生長方形的副程式

```
81 void rectangle(double x1, double y1, double x_len, double y_len ,int r , int g , int b)
82 {
83     glColor3ub(r, g, b);
84     glVertex2f(x1,y1) ;
85     glVertex2f(x1,y1+y_len) ;
86     glVertex2f(x1+x_len,y1+y_len) ;
87     glVertex2f(x1+x_len,y1) ;
88
89 }
```

### 2. 用 FOR 迴圈一次產生所有上方的橫條的位置

```
97 void CreateRectangle()
98 {
99     ....// 方塊X座標
100     ....double TempPositionX = -0.9f;
101     ....double TempPositionY = 0.8f;
102     ....for(int i=0 ; i<col*row ;i++)
103     ....{
104         RectX[i] = TempPositionX;
105         TempPositionX += 0.3f;
106         if(i % col == col-1 )
107             TempPositionX = -0.9f;
108
109         RectY[i] = TempPositionY;
110         TempPositionY -= 0.125f;
111         if(i % row == row-1 )
112             TempPositionY = 0.8f;
113
114         RectShow[i] = 1;
115     }
116 }
```

### 3. 鍵盤輸入事件

```
122 double speed = 0.025;
123 int KeyBoardInput(GLFWwindow *window)
124 {
125     if (glfwGetKey(window, GLFW_KEY_A) == GLFW_PRESS)
126     {
127         if(DX >= -1)
128             DX -= speed;
129     }
130     else if (glfwGetKey(window, GLFW_KEY_D) == GLFW_PRESS)
131     {
132         if(DX <= 0.75)
133             DX += speed;
134     }
135     else if (glfwGetKey(window, GLFW_KEY_SPACE) == GLFW_PRESS) // start game
136     {
137         if(StartGame == false && Level == 2)
138         {
139             DistoryRectNumber = 0;
140             StartGame = true;
141         }
142         else if(StartGame == false && Level == 1)
143         {
144             StartGame = true;
145         }
146     }
147     else if (glfwGetKey(window, GLFW_KEY_S) == GLFW_PRESS)
148     {
149         if(D2Y >= -1)
150             D2Y -= speed;
151     }
152     else if (glfwGetKey(window, GLFW_KEY_W) == GLFW_PRESS)
153     {
154         if(D2Y <= 0.75)
155             D2Y += speed;
156     }
157 }
```

### 4. 繪出上方的橫條

```
// Start Drawing here
glBegin(GL_QUADS);
{
    // 障礙物
    bool TempColor = 0;
    for(int i=0 ; i<col*row ; i++)
    {
        if(i % col == 0)
            TempColor = !TempColor;
        if(TempColor && RectShow[i])
            rectangle(RectX[i],RectY[i],0.25,0.075,119, 46, 37);
        else if(!TempColor && RectShow[i])
            rectangle(RectX[i],RectY[i],0.25,0.075,25, 114, 120);
    }
    // 下方移動桿
    rectangle(DX,DY,0.25,0.075,255, 424, 186);

    if(Level == 2)
        rectangle(D2X,D2Y,0.075,0.25,255, 424, 186);
}
glEnd();
// 王老五
```

## 5. 繪出 彈跳球

```
188 // 球球
189 int n = 100;
190 GLfloat R = 0.03f;
191 GLfloat Pi = 3.1415926536f;
192 glBegin(GL_POLYGON);
193 {
194     glColor3ub(196, 69, 54);
195
196     // 還沒開始遊戲
197     if(StartGame == false)
198     {
199         BX = DX + 0.12f;
200         BY = DY + 0.11f;
201         for(int i=0; i<n; ++i)
202             glVertex2f(BX+ R*cos(2*Pi/n*i) ,BY+ R*sin(2*Pi/n*i));
203     }
204
205     // 遊戲開始
206     if(StartGame)
207     {
208         if(BX > 1.0f)
209             xPath = -1;
210         if(BX < -1.0f)
211             xPath = 1;
212         BX += (0.01f * xPath);
213
214         if(BY > 1.0f)
215             yPath = -1;
216         if(BY < -1.0f)
217             yPath = 1;
218         BY += (0.01f * yPath);
219
220         for(int i=0; i<n; ++i)
221             glVertex2f(BX+ R*cos(2*Pi/n*i),BY +R*sin(2*Pi/n*i));
222     }
223 }
224 glEnd();
```

## 6. 繪出生命提示球

```
227 // 生命提示
228 if(heart >= 1)
229 {
230     glBegin(GL_POLYGON);
231     {
232         glColor3ub(255, 0, 0);
233         for(int i=0; i<n; ++i)
234             glVertex2f(0.92f + R*cos(2*Pi/n*i), -0.92f+R*sin(2*Pi/n*i));
235     }
236     glEnd();
237 }
238 if(heart >= 2)
239 {
240     glBegin(GL_POLYGON);
241     {
242         glColor3ub(255, 0, 0);
243         for(int i=0; i<n; ++i)
244             glVertex2f(0.92f - 0.09f + R*cos(2*Pi/n*i), -0.92f+R*sin(2*Pi/n*i));
245     }
246     glEnd();
247 }
248 if(heart >= 3)
249 {
250     glBegin(GL_POLYGON);
251     {
252         glColor3ub(255, 0, 0);
253         for(int i=0; i<n; ++i)
254             glVertex2f(0.92f - 0.18f + R*cos(2*Pi/n*i), -0.92f+R*sin(2*Pi/n*i));
255     }
256     glEnd();
257 }
258 }
```

## 7. 判斷球是否撞擊橫桿

```
261 void BoolHit()
262 {
263     // 撞擊下方
264     for(int i=0; i<col*row; i++)
265     {
266         if(BX >= RectX[i] && BX <= RectX[i]+0.25f && BY >= RectY[i] && BY <= RectY[i]+0.075f && RectShow[i])
267         {
268             // 撞擊移動的桿子 easy
269             if(BX >= DX && BX <= DX+0.25f && BY <= DY+0.075f)
270             {
271                 BY = DY+0.075f;
272                 yPath = -yPath;
273             }
274             if(BY <= DY)
275             {
276                 heart -= 1;
277                 StartGame = false;
278             }
279             // hard
280             if(Level == 2)
281             {
282                 if(BY >= D2Y && BY <= D2Y+0.25f && BX >= D2X)
283                 {
284                     BX = D2X;
285                     xPath = -xPath;
286                 }
287                 if(BX >= D2X+0.075f)
288                 {
289                     heart -= 1;
290                     StartGame = false;
291                 }
292             }
293         }
294     }
295 }
```

## 8. 主程式

```
388 int main(void)
389 {
390     onInit();
391     CreateRectangle();
392     while (!glfwWindowShouldClose(window))
393     {
394         KeyBoardInput(window);
395         onResize();
396         onUpdate();
397         onRender();
398         BoolHit();
399         if(DistoryRectNumber == num )
400         {
401             if(Level == 3)
402             {
403                 for(int i=0 ; i<num ; i++)
404                     RectShow[i] = 0;
405             }
406             else if(Level == 2)
407             {
408                 StartGame = false;
409                 Level = 3;
410             }
411             else if(Level == 1)
412             {
413                 StartGame = false;
414                 Level = 2;
415                 DistoryRectNumber = 0;
416                 for(int i=0 ; i<42 ; i++)
417                 {
418                     RectShow[i] = 1;
419                 }
420                 heart = 3;
421             }
422         }
423         if(heart == 0)
424         {
425             StartGame = false;
426             printf("end Game\n");
427         }
428         glfwSwapBuffers(window);
429         glfwPollEvents();
430     }
431     glfwDestroyWindow(window);
432     glfwTerminate();
433     exit(EXIT_SUCCESS);
434 }
435
```