

計算機程式期末專題書面報告

姓名：謝佑芃 學號：B06901184 系級：電機一

(一) 遊戲設計

一開始我們分別上網找了很多遊戲實況的影片，再把三個人喜歡的遊戲類型取交集，再做一些統整跟添加之後，得出三個主要大點：1.雙人對戰 2.縱向滾軸 3.特殊道具，三個主要訴求。由於其中一名組員的綽號是小浴巾，而我又喜歡寶寶，因此我們的主角便是一個白寶寶與一個黑寶寶，道具則是各色的浴巾。黑浴巾（後因為覺得戴墨鏡比較合理而改成墨鏡）能使對方減慢速度，綠浴巾能使自己加快速度，藍浴巾能使兩人位置互換，而紅浴巾則能往下掉一階。玩家一（白寶寶）用AD控制左右，而玩家二（黑寶寶）用左右鍵移動。在得到道具後，前者按W能使道具發揮作用，後者則是按上。遊戲無止盡延續下去，直到玩家碰到畫面頂端便結束。

(二) 工作分配

謝佑芃：人物動畫及控制程式、圖片美工。

郭育銘：地圖與主程式。

吳悠嘉：影片、音效、Menu設計與程式。

(三) 基本程式設計與執行流程

1. 第一個迴圈

```
673         while( !quit )
```

讓程式在 quit 之前一直執行

2. 第二個迴圈

```
676         while( SDL_PollEvent( &e ) != 0 )
```

控制各個 ScreenState 的執行

3. 用 switch 將 ScreenState 存成各個 case

```
710         switch(ScreenState)
711         {
713             case START:
716             case MENU:
733             case INSTRUCTION:
736             case COUNTDOWN:
786             case PLAYING:
```

```
888             case GAMEOVER:
```

在 GAMEOVER 中可以 RESTART 回去 PLAYING，QUIT 退出，else 回到 MENU

```
893                 if(GameOverSelect==RESTART)
899                 else if(GameOverSelect==QUIT)
905                 else
```

(四) class (物件導向程式設計) 相關部分

```
86 class item
87 {
88 private:
89     int category;
90     bool create;
91     SDL_Rect ItemRec;
92     static int ItemVy;
93 public:
94     item()
95     {
96         create=false;
97         ItemRec={SCREEN_WIDTH,SCREEN_HEIGHT,60,60};|
98     }
99     static void SetItemVy(int vy){ItemVy=vy;}
100     void SetCreate(bool y){create=y;}
101     bool GetCreate() const{return create;}
102     void CreateItem()
103     {
104         category=rand()%4;
105         ItemRec.x=rand()%540+20;
106         ItemRec.y=825;
107         if(category==BLACK)ItemRec.h=30;
108         else ItemRec.h=60;
109     }
110     void MoveItem()
111     {
112         ItemRec.y-=ItemVy;
113         if(ItemRec.y+ItemRec.h<=0)create=false;
114     }
115     void loadItem()
116     {
117         if(category==BLUE) loadpngwithrec("image/bluetowel.png",ItemRec);
118         if(category==GREEN)loadpngwithrec("image/greentowel.png",ItemRec);
119         if(category==BLACK)loadpngwithrec("image/sunglasses.png",ItemRec);
120         if(category==RED) loadpngwithrec("image/redtowel.png",ItemRec);
121     }
122     int BabyGetItem(const SDL_Rect &player)
123     {
124         //from left top || from right top
125         if((player.x+player.w>=ItemRec.x && player.x+player.w<=ItemRec.x+ItemRec.w && ItemRec.y<=player.y+player.h && ItemRec.y>=player.y)
126             ||(player.x<=ItemRec.x+ItemRec.w && player.x>=ItemRec.x && ItemRec.y<=player.y+player.h && ItemRec.y>=player.y))
127         {
128             create=false;
129             get=true;
130             return category;
131         }
132         else return -1;
133     }
134 };
```

class item

1) 物件化優點：

為了做出擁有多樣道具的遊戲，即便道具功能不同，程式碼中需要重複多次相同功能，像是 item 的高度寬度、隨機出現、隨畫面移動、圖片讀取、

人物得到道具等功能。如果新增一個道具都要重新寫一次所有基本設定，會讓程式碼很大浪費記憶體也浪費時間，因此用 class 處理相同類型的物件。

2) Constructor:

item() 將高寬設為 60x60，出現範圍設為與 screen 同大小

3) Static 的使用：

用 static void SetItemVy()讓物件上升速度一直等於畫面上升速度

in class floor:

```
147         static int GetFloorVy(){return FloorVy;}
```

in main:

```
669         item::SetItemVy(floor::GetFloorVy());
```

(五) 個人程式撰寫與設計

1. 寶寶移動動畫

把 sprite[12]寫在 class baby 外，原圖為 7 張圖，但是為了讓動作更精緻細膩，讓動作保持連續性，所以讓他是跑到最後再往前跑，以下程式碼來看，跑到 480pixel 的時候會往回到 400，連續下去。

```
564     const SDL_Rect baby::sprite[]={ { 0,0,80,80},{ 80,0,80,80},{160,0,80,80},{240,0,80,80},{320,0,80,80},{400,0,80,80},
565                                     {480,0,80,80},{400,0,80,80},{320,0,80,80},{240,0,80,80},{160,0,80,80},{ 80,0,80,80}};
```

class baby 中

private:

```
259         const static SDL_Rect sprite[12];
```

public:

MoveBaby

```
350         if(vx!=0)
351         {
352             frame++;
353             frame%=12;
354         }
355         if(vx<0)
356         {
357             left=true;
358             vx=-Speedx;
359         }
360         if(vx>0)
361         {
362             left=false;
363             vx=Speedx;
364         }
365         babyrec.x+=vx;
366         if(babyrec.x<=0)babyrec.x=0;
367         if(babyrec.x>=SCREEN_WIDTH-BabyWidth)babyrec.x=SCREEN_WIDTH-BabyWidth;
```

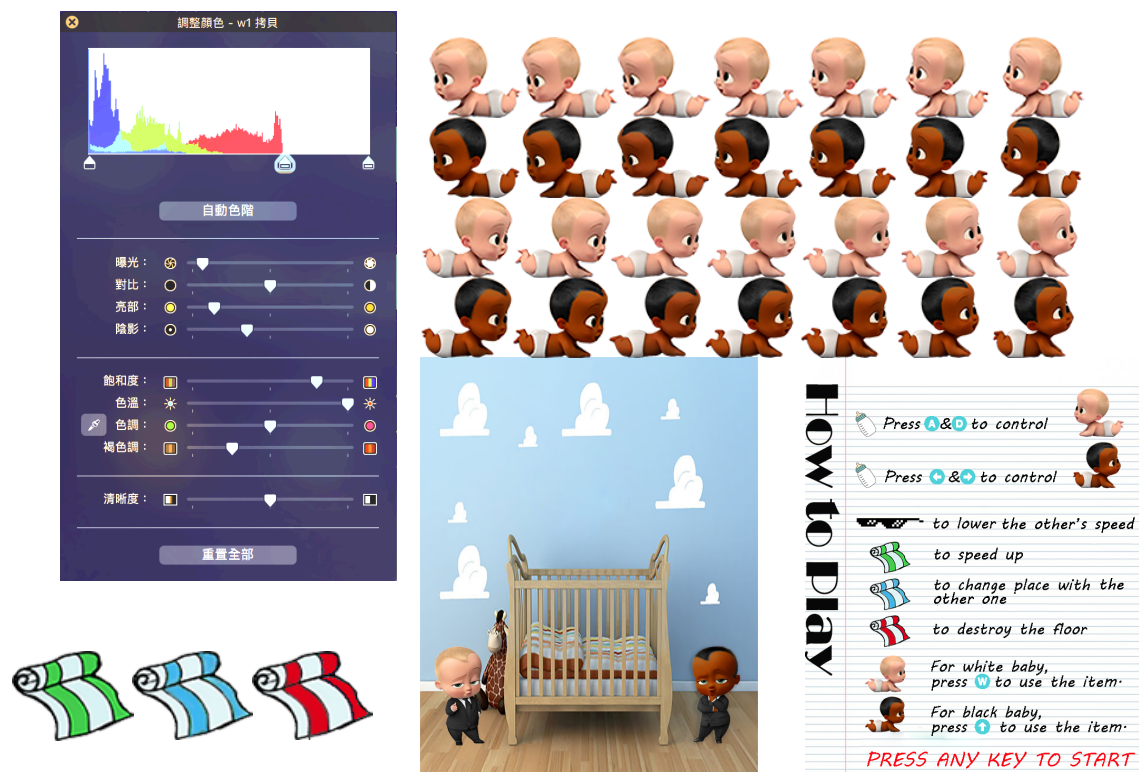
handleEvent:

```
491     void handleEvent(SDL_Event& e,int playerNum)
492     {
493         switch(playerNum)
494         {
495             case 1:
496                 //If a key was pressed
497                 if( e.type == SDL_KEYDOWN && e.key.repeat == 0 )
498                 {
499                     //Adjust the velocity
500                     switch( e.key.keysym.sym )
501                     {
502                         case SDLK_a:
503                             vx -= Speedx;
504                             break;
505                         case SDLK_d:
506                             vx += Speedx;
507                             break;
508                         case SDLK_w:
509                             use=true;
510                             break;
511                     }
512                 }
513                 //If a key was released
514                 else if( e.type == SDL_KEYUP && e.key.repeat == 0 )
515                 {
516                     //Adjust the velocity
517                     switch( e.key.keysym.sym )
518                     {
519                         case SDLK_a:
520                             vx += Speedx;
521                             break;
522                         case SDLK_d:
523                             vx -= Speedx;
524                             break;
525                     }
526                 }
527                 break;
```

下面的 case 2 為第二個玩家，adw 控制鍵改為左右上，其餘以此類推。

2.美工部分

這次專題我貢獻了很多修圖，為了呈現出動態的寶寶，我把影片連續截圖，再一共修了 $7 \times 4 = 28$ 個寶寶圖片，下面第一張圖為白寶寶轉換成黑寶寶的調色，調色後將每個寶寶的尿布與眼白部分修剪下，貼上黑寶寶，接著把白寶寶的頭髮也一一剪裁下來，調整黑白，再貼上黑寶寶，以達到更生動的動畫效果。以下為修圖成果們～



(六) 心得

這次專題中，我覺得在各種方面的能力上都進步與收穫良多。期中前對 **C++** 的扎根，到期末能認識 **SDL** 並寫出一個自己的遊戲，把原本只在腦中想像的畫面呈現到眼前真的十分感動。一開始讀 **lazyfoo** 網站的時候常常讀不下去，但到最後我能感受到自己閱讀速度與程式真的大幅的進步許多。美工的部分，我熟悉了從來都沒用過的 **Photoshop**，做出自己十分滿意的圖片。我認為我沒有做好的地方是，在這學期期末時，考試與報告的壓力讓我到最後繼承的部分沒有學得夠扎實，或許是因為自己能力不足，或許是因為自己不夠努力，程式裡更簡潔的方法沒有用到，這是我認為最可惜的地方。但也因此，我會在寒假的時候對這個程式做修改，也會再試著做一個完完整整屬於自己的遊戲。