

# 逢甲大學

資訊工程學系專題報告

互動式幼兒教學

軟體製作

學生：林貫中(四甲)

林孟傑(四甲)

指導教授：謝信芳教授

中華民國 93 年 5 月

# 目 錄

目錄 .....	1
第一章 導論.....	1
1.1 背景.....	1
1.2 動機.....	1
1.3 目的.....	3
第二章 系統概述.....	4
2.1 劇本.....	5
2.2 角色.....	7
2.3 環境.....	7
2.4 軟硬後端.....	7
第三章 系統分析.....	8
第四章系統製作.....	16
4.1 記憶大考驗.....	16
4.2 發射飛彈遊戲.....	26
4.3 選擇題設計.....	35
4.4 填充題設計.....	43
4.5 趣味連連看.....	49
4.6 鍵盤打字練習.....	54

第五章 系統評估.....	62
5.1 記憶大考驗.....	62
5.2 飛彈射擊遊戲.....	62
5.3 選擇題設計.....	63
5.4 填充計算設計.....	63
5.5 趣味連連看.....	64
5.6 鍵盤打字練習.....	64
第六章 心得感想與展望.....	65
6.1 心路歷程.....	65
6.2 未來展望.....	66
6.3 感想.....	67
6.4 參考文獻.....	67

## 圖表目錄

圖 3.1 系統架構圖	8
圖 3.2 記憶大考驗邏輯流程圖	9
圖 3.3 發射飛彈遊戲邏輯流圖	10
圖 3.4 選擇題設計邏輯流程圖	11
圖 3.5 填充計算題邏輯流程圖	12
圖 3.6 趣味連連看邏輯流程圖	13
圖 3.7 鍵盤打字練習邏輯流圖	14
圖 4.1 記憶大考驗之 cast 面板	16
圖 4.2 完成後之記憶大考驗 Score 面板	18
圖 4.3 記憶大考驗遊戲開始圖	24
圖 4.4 記憶大考驗遊戲中圖	24
圖 4.5 記憶大考驗得分畫面圖	25
圖 4.6 發射飛彈遊戲 Cast 面板	26
圖 4.7 完成後之 Score 面板之一	28
圖 4.8 完成後之 Score 面板之二	28
圖 4.9 飛彈射擊遊戲演員 1	29
圖 4.10 飛彈射擊遊戲初始畫面	31
圖 4.11 飛彈射擊遊戲中畫面	31

圖 4.12 飛彈射擊遊戲演員 2	32
圖 4.13 飛彈射擊遊戲初始畫面	34
圖 4.14 飛彈射擊遊戲中畫面	34
圖 4.15 選擇題設計之 Cast 板	35
圖 4.16 完成後之 Score 面板	37
圖 4.17 選擇題設計遊戲畫面	41
圖 4.18 選擇題設計答對畫面	42
圖 4.19 選擇題設計答錯畫面	42
圖 4.20 填充題設計之 Cast 面板	43
圖 4.21 填充題完成後之 Score 面板	45
圖 4.22 填充題遊戲畫面	47
圖 4.23 填充題答對畫面	48
圖 4.24 填充題答錯畫面	48
圖 4.25 趣味連連看之 Cast 面板	49
圖 4.26 趣味連連看完成後之 Score 面板	51
圖 4.27 趣味連連看遊戲初始畫面	52
圖 4.28 趣味連連看遊戲中畫面 counter=3	53
圖 4.29 趣味連連看遊戲完成畫面	53
圖 4.30 鍵盤打字練習之 Cast 面板	54

圖 4.31 鍵盤打字練習完成後之 Score 面板之一.....	56
圖 4.32 鍵盤打字練習完成後之 Score 面板之二.....	57
圖 4.33 鍵盤打字練習完成後之 Score 面板之三.....	58
圖 4.34 鍵盤打字練習完成後之 Score 面板之四.....	58
圖 4.35 鍵盤打字練習遊戲初始畫面登入姓名.....	60
圖 4.36 鍵盤打字練習遊戲開始畫面.....	61
圖 4.37 鍵盤打字練習遊戲結束畫面~排行版.....	61
表 4-1 記憶大考驗之演員參數表.....	17
表 4-2 發射飛彈遊戲之演員參數表.....	27
表 4.3 選擇題設計之演員參數表.....	36
表 4-4 填充題設計之演員參數表.....	44
表 4-5 趣味連連看之演員參數表.....	50
表 4-6 鍵盤打字練習之演員參數表.....	55

## 第一章 導論

### 1.1 背景

現在市場上的遊戲大多以營利賺錢為目的，然而舒不知這種遊戲對小朋友們一點用處都沒有，只會浪費金錢和時間，對於本身沒有幫助，很少為了小朋友們作一些對功課對智慧啟發有幫助的軟體，所以我們趁這次專題的機會，實作一些遊戲，讓小朋友們可以一邊遊戲中學習新的知識。

### 1.2 動機

隨著電腦的日新月異，人們對於精密機械的需求日益增加，為了達到節省人力、物力，空間與時間以求生存及成長，電腦科技應運而生，教育事業的現代化當然也需要電腦技術的引導與推動。就教育觀點而言，時代背景的不同，人們目前所需吸收知識的範圍和深度都不同於往昔，為適應個別化的學習，及使教學不受時空限制，電腦輔助教學(簡稱 CAI)的應用越來越廣泛，藉由電腦輔助教學軟體來幫助訓練與學習，不但可以避免時間與空間的限制，更能節制訓練的經費

與成本。

由於科技的進步和時代的變遷，我們的生活不在像以前那樣的單純，所以現在的小朋友們日常生活中接觸到的事物是越來越多，而且越來越多元化，所需吸收的知識範圍和深度都異於往昔。我們都知道，傳統定時定點的教學方式，不僅現在學生學習的空間、時間，往往讓學生興趣缺缺，學習效果不彰，為了適應個別化的學習，及使教學不受時間和空間的限制，電腦輔助教學方式更成為教育應用上的焦點。

大家都知道，國內的教學方式是以「填鴨式」的方法來強迫學生學習，非常的刻版，而一般的教科書，內容大多枯燥乏味，造成小學生缺乏學習的興趣與求知的慾望，若能將教學融入在遊戲之中，必能大大的提高學生學習的是興趣和效果，且老師的教學方式是一對多，無法掌握每一位學生的學習進度，故教育當局近幾年來，大力的推動各級中小學的教學電腦化。電腦是現今生活中不可或缺的工具，使用的年齡層也有下降的趨勢，為了因應未來資訊時代的到來，提早讓小學生接觸到電腦是必然的趨勢，若有一套軟體，能夠讓

小學生輕鬆的學習電腦課程，對於未來的社會，一定會有大大的幫助。

### 1.3 目的

1. 對於學校式的教學，小朋友們感到非常的枯燥乏味，所以我們想改變以往的教學方式，使他們能達到寓教於樂的目的。
2. 由於目前的小朋友對於數學尤感非常排斥，一聽到數學二字往往就放棄，這種排斥的心理對於在日後學習數學會有很嚴重的影響，所以我們兩個就想要藉著這次專題的實驗，實作一套有關兒童數學教學軟體，讓小朋友們藉著我們的軟體一邊遊戲一邊學習。
3. 因為電腦輔助教學可依學生程度的不同，提供不同的教學方式，互動式的教學方式，可以兼顧到每位學生的進度；生動的教學，可讓學生快樂的學習，提高學習效果；更大的優點是可以跳脫時空的限制，即使學生在家，也能輕鬆的學習，得到知識。

4. 因為以上幾種原因，我們專題決定做 6 種小遊戲：

(1) 記憶大挑戰

(2) 發射飛彈遊戲

(3) 選擇題設計

(4) 填充計算題庫設計

(5) 趣味連連看

(6) 鍵盤打字練習

這六種遊戲可以引發學生的記憶能力，還有數學能力，增加學生對於學習上的興趣。



## 第二章 系統概述

### 2.1 劇本

#### 1 記憶大挑戰

此遊戲的創作目的，用來訓練記憶力。遊戲一開始會有三秒的開圖時間讓你瀏覽並記下幾個相似的圖示。

遊戲規則：在 60 秒的時間內只要連翻兩個一樣的圖示即得分。



#### 2 發射飛彈遊戲

發射飛彈遊戲屬於比較活的動態遊戲，射擊遊戲考驗的是瞬間時間的反應力，應變力。

遊戲規則流程，主人翁可以控制一台可發射飛彈的飛機，並射擊上方四個帶有數字的汽球，遊戲融合了讓小朋友認識數學的機制，讓小朋友樂在其中。

### 3 選擇題設計

我們設計了多個數學上的問題，融合數學的基本概念來考小朋友們，問題會隨機出現，遊戲結束會統計小朋友答對的題數。

### 4 填充題設計

填充題的設計比較偏向加法的相加，十位數的相加。遊戲的進行方式和選擇題的模組相似，遊戲結束會做個加總統計分數，這兩個遊戲是屬於測驗的遊戲。

### 5 趣味連連看

趣味連連看的設計目的是讓小朋友認識數字的連續性，譬如說 1 下一個數是 2 下一個是 3，數字結著卡通圖案作數字學習的遊戲

### 6 英文打字練習

簡單的認識英文，靠手部的技巧，邊看邊學，也可以練習打字的速度。

## 2.2 角色

角色的設定在射擊遊戲作特別的說明，因為其他的三個遊戲是屬於比較靜態的。

在射擊遊戲裡，主人翁控制著一台飛機 以左右控制水平的移動方式，空白鍵發射飛彈，上方會有四個氣球自由落下。

## 2.3 環境

遊戲的環境在 OS:Windows98 以上皆可執行

遊戲的製作 MicroMedia DirectorX 8.5

## 2.4 軟硬後端

硬體基本設備： Cpu 600 Mhz

Ram 128 Mb

顯示卡 64Mb

顯示器 15 吋

硬碟 10 G

音效卡

軟體配備： 需下載 MircoMedia 系列 DirectorX

### 第三章 系統分析

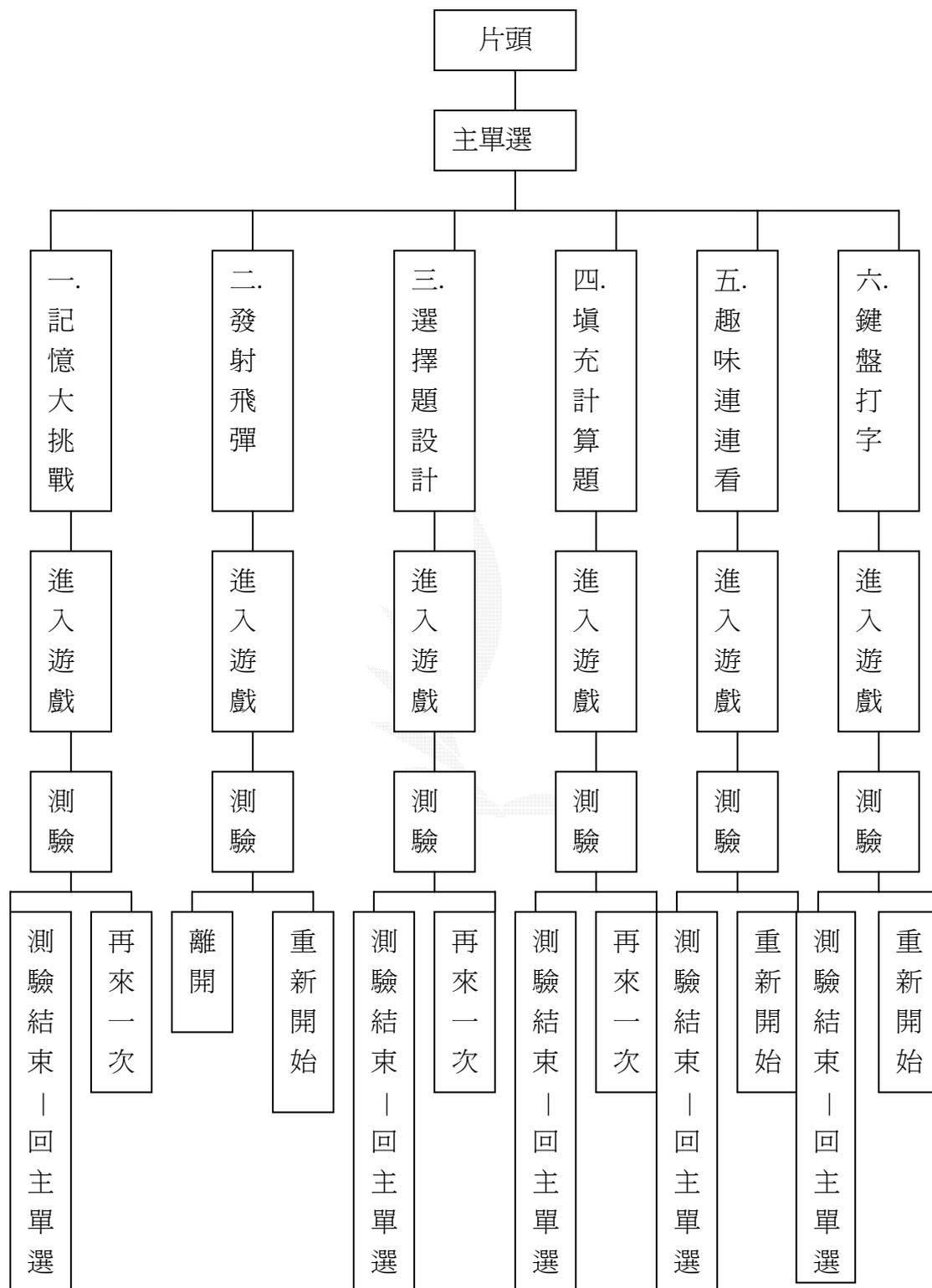


圖 3.1 是系統架構圖

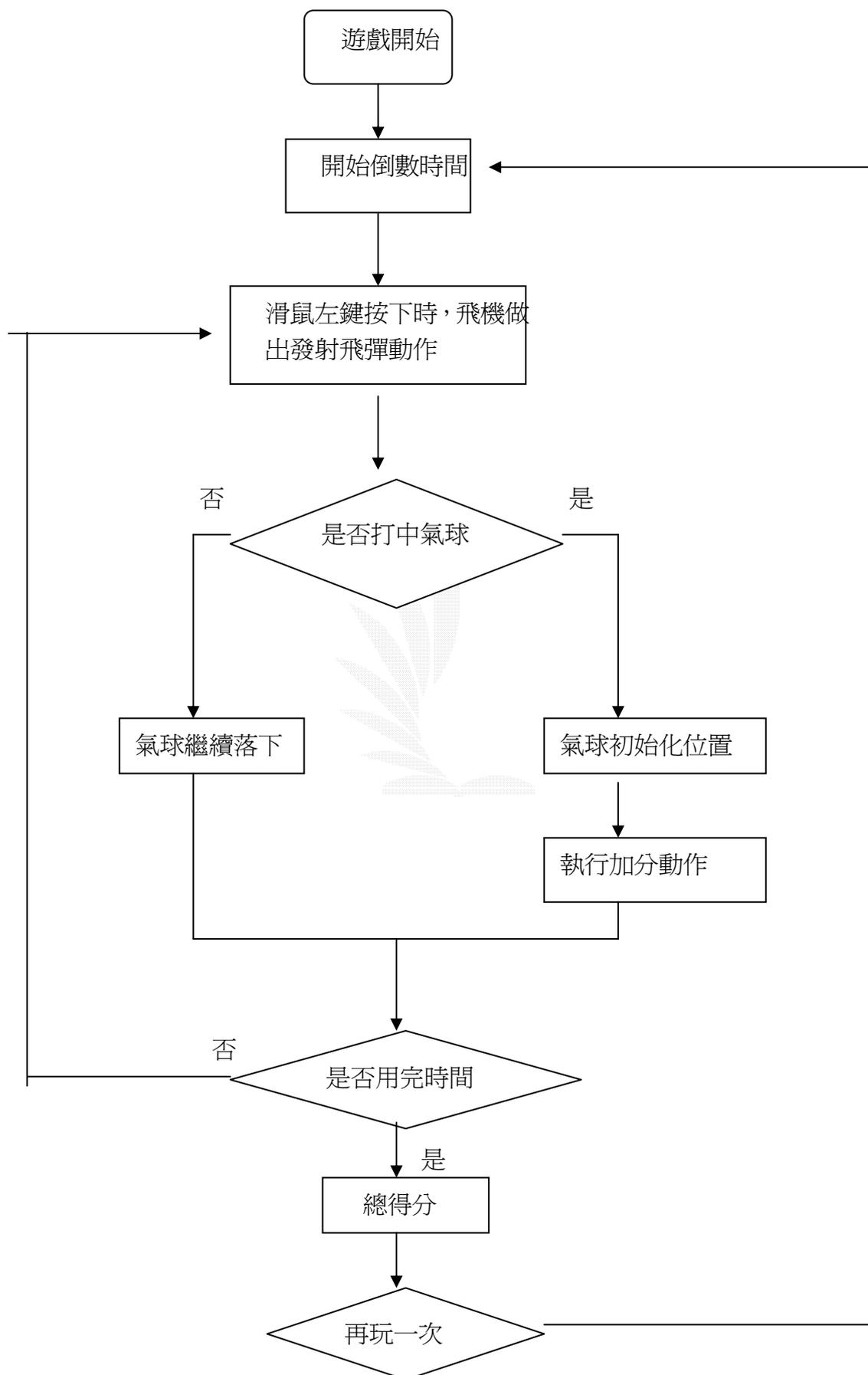


圖 3.2(發射飛彈遊戲邏輯流程圖)

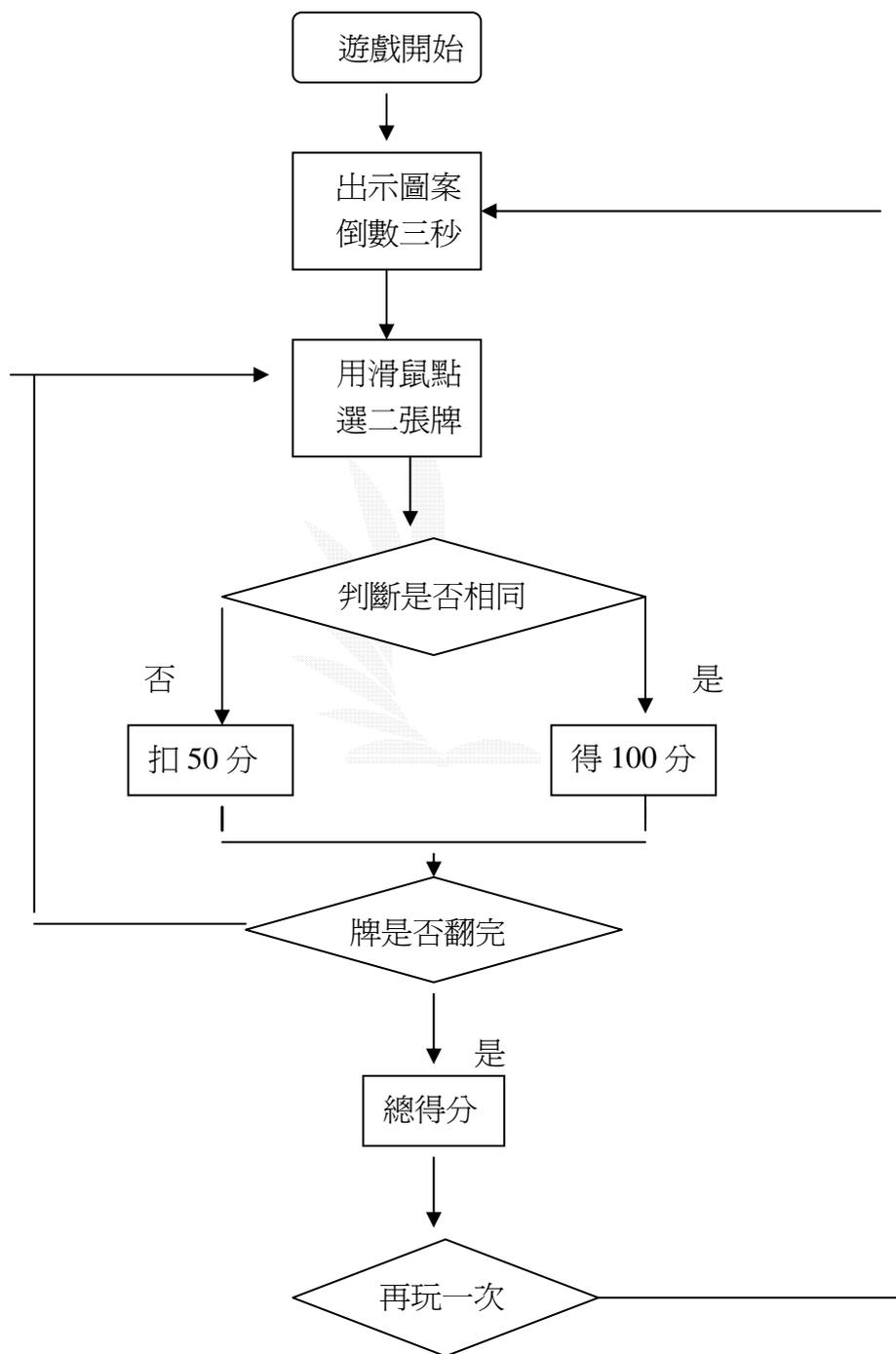


圖 3.3(記憶大考驗邏輯流程圖)

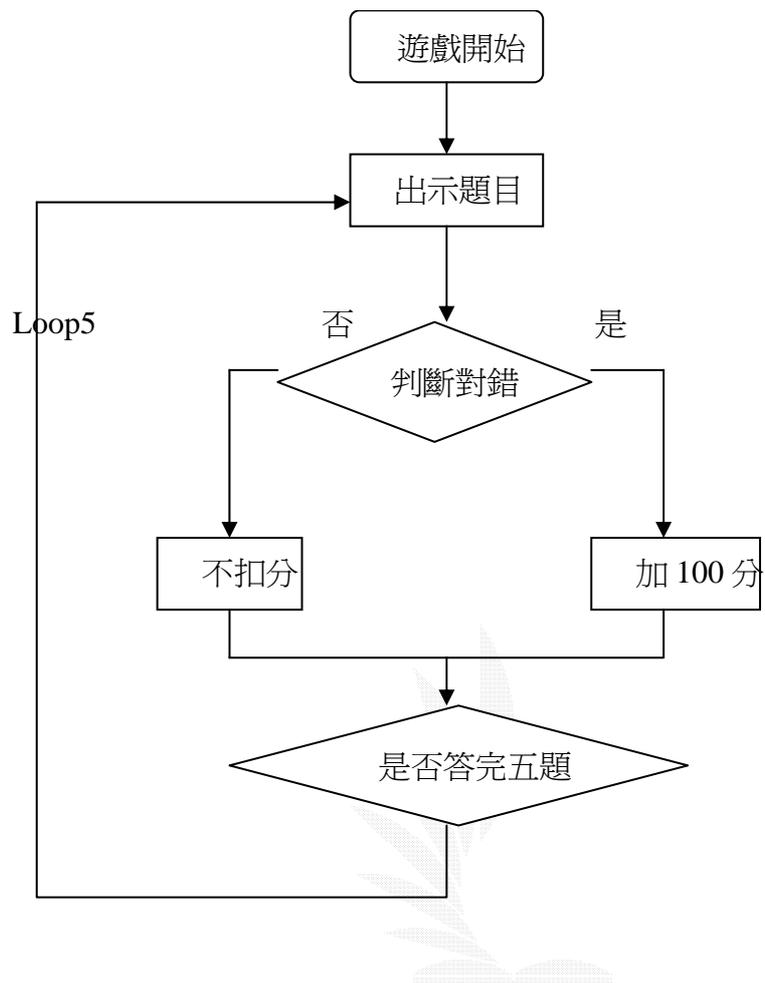


圖 3.4(選擇題設計邏輯流程圖)

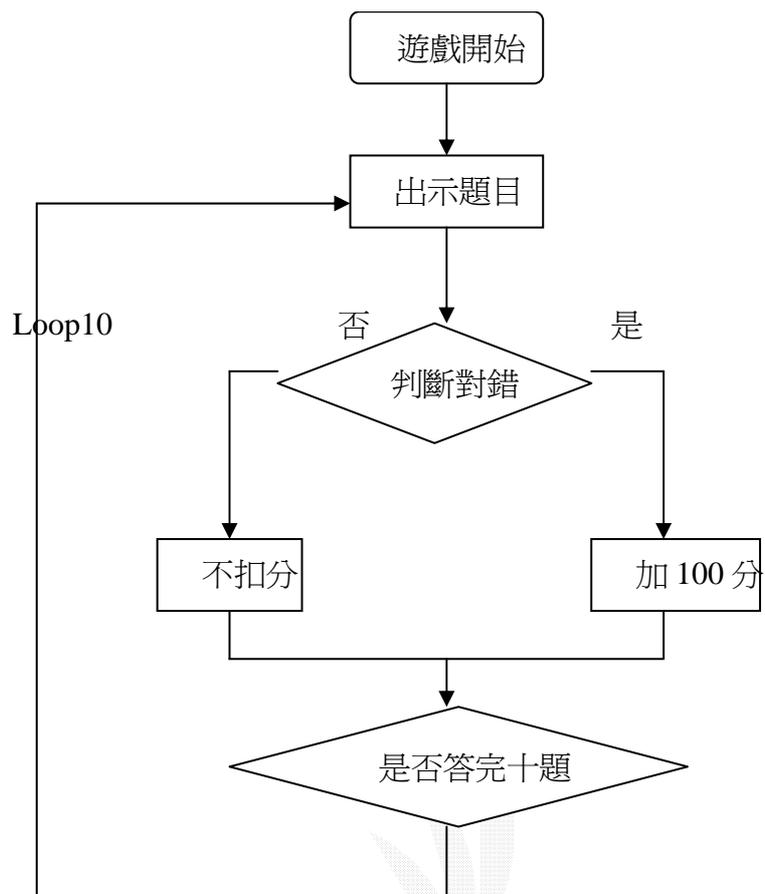


圖 3.5(填充計算題邏輯流程圖)

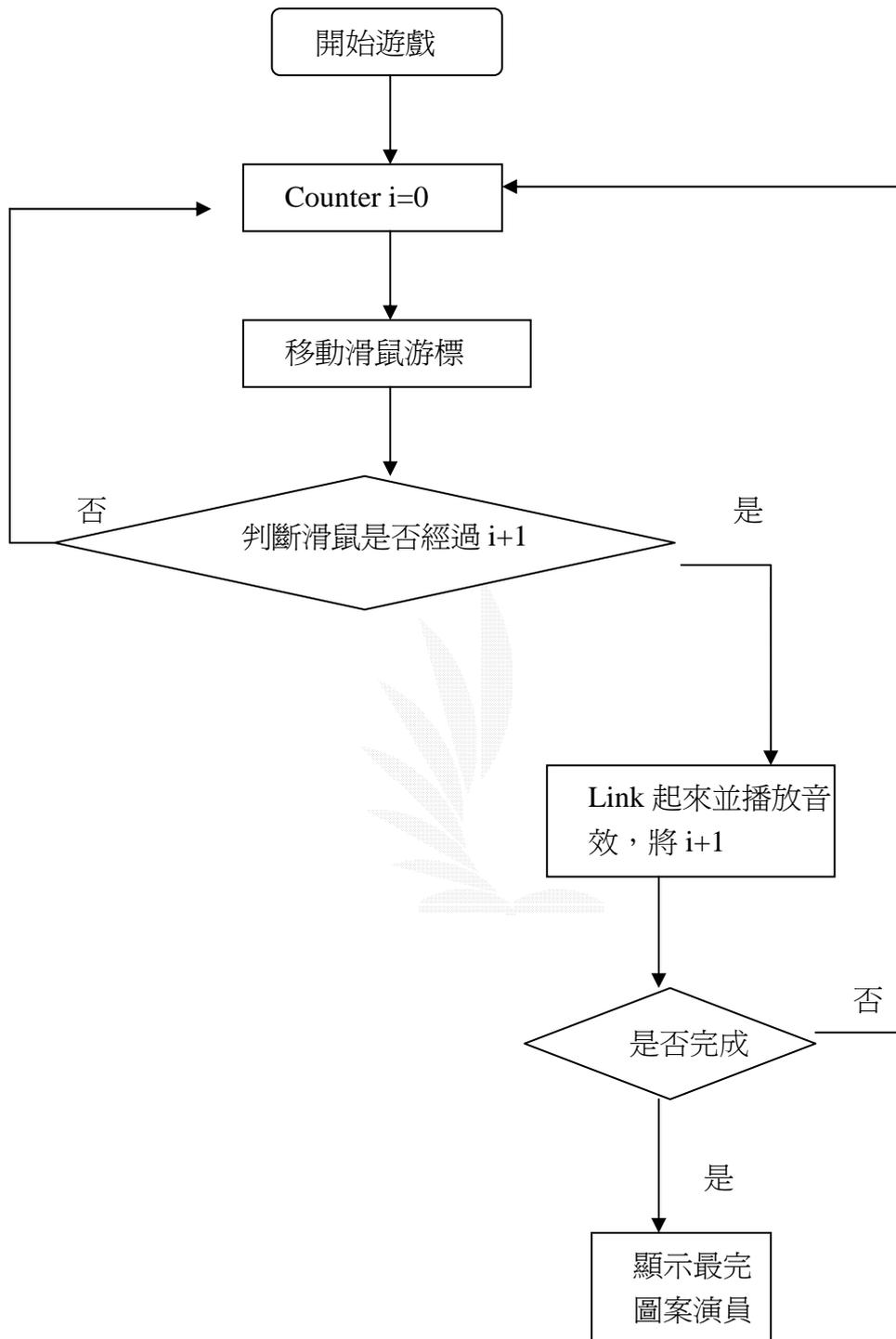
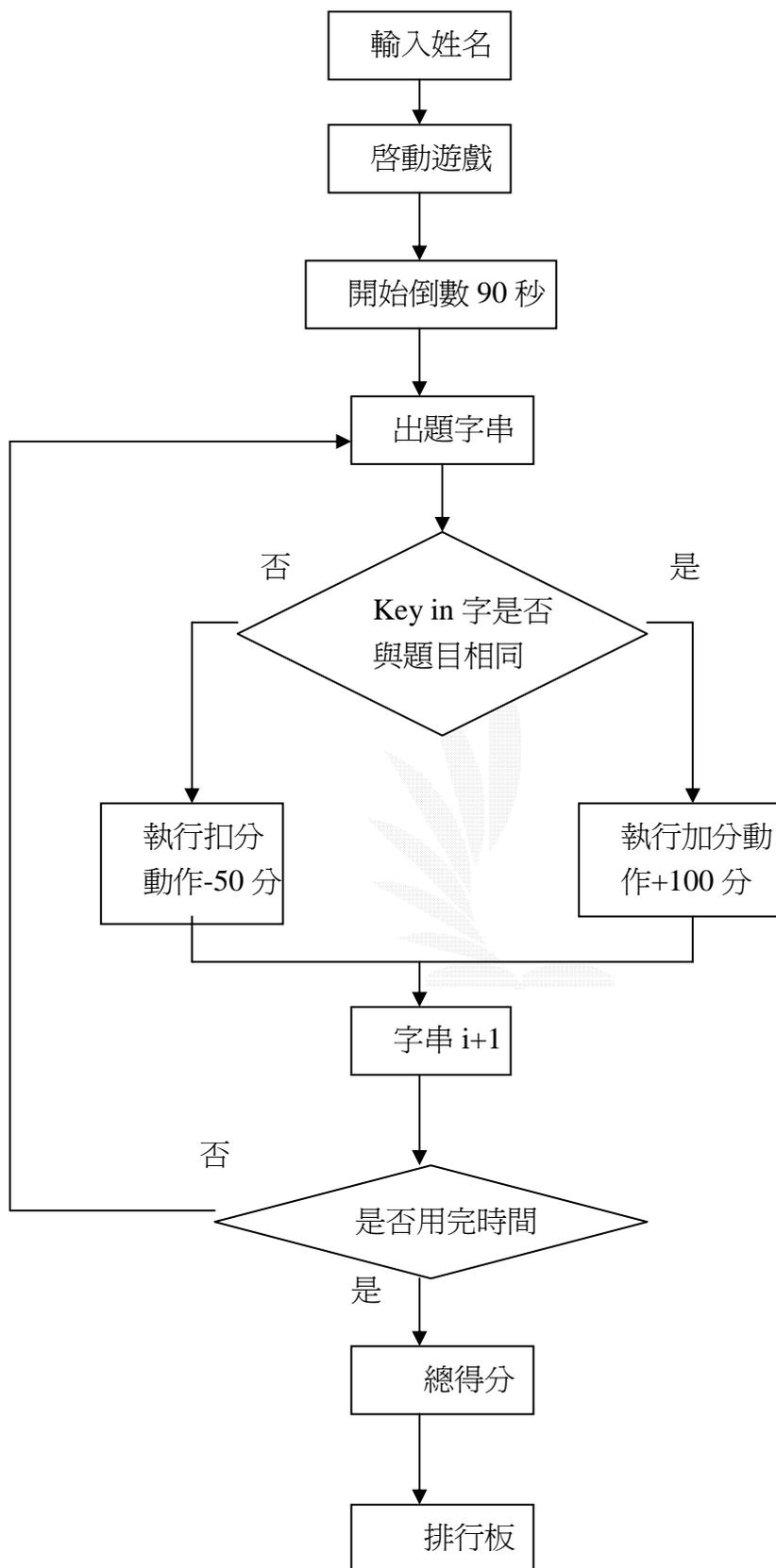


圖 3.6(趣味連連看邏輯流程圖)



### 圖 3.7(鍵盤打字練習邏輯流程圖)



## 第四章 系統製作

### 4.1 記憶大考驗

下圖為”記憶大考驗”所需的演員

記憶大考驗之 Cast 面板

Cast 面板說明

編號 1~15 的演員為題目的圖示演員，編號 21~22 為主要背景演員。

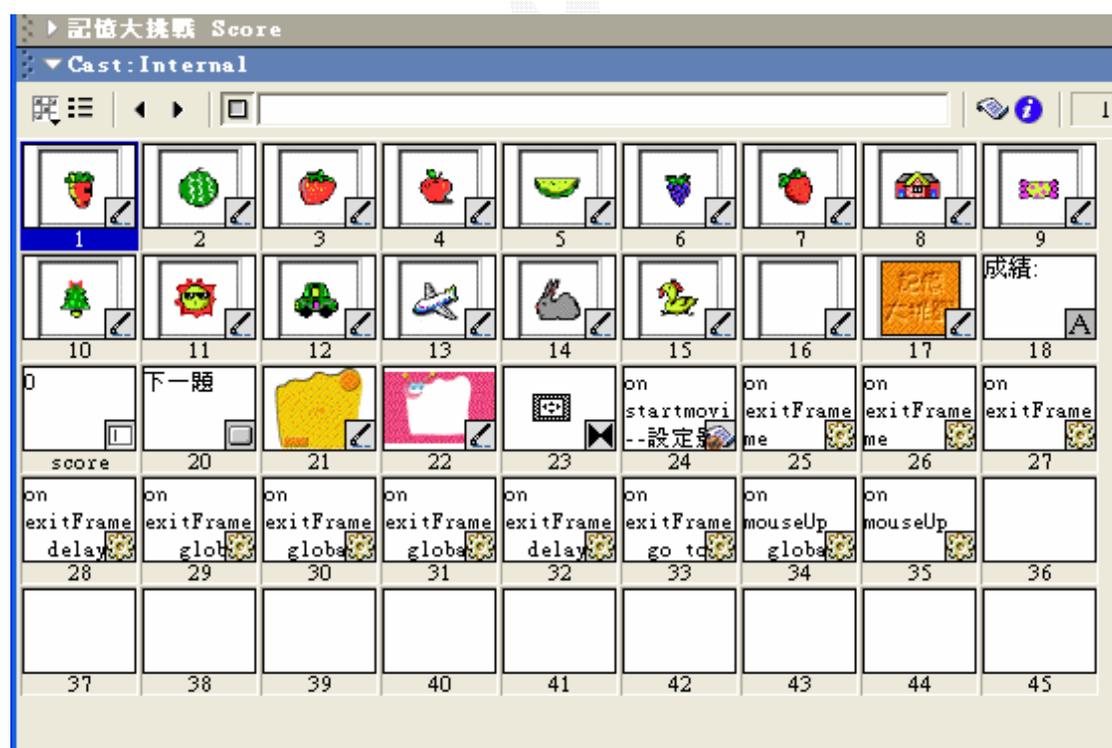


圖 4.1 (記憶大考驗之 cast 面板)

## 編排的 Score 面板

說明:將 Cast 面板的演員拖移到 Score 面板中，做時間點上的安排 frame 為 Score 上的一個時間單位，X, Y 值為演員在舞台上的位置參數。

Cast	Sprite	放置範圍	X 值	Y 值	Ink 值
Cast21	Sprite1	Frame1~12	293	186	Copy
Cast12	Sprite2	Frame1~12	300	200	Background Transport
Cast17	Sprite3	Frame4~7	234	150	Copy
Cast17	Sprite4	Frame4~7	295	150	Copy
Cast17	Sprite5	Frame4~7	356	150	Copy
Cast17	Sprite6	Frame4~7	417	150	Copy
Cast17	Sprite7	Frame4~7	234	207	Copy
Cast17	Sprite8	Frame4~7	295	207	Copy
Cast17	Sprite9	Frame4~7	356	207	Copy
Cast17	Sprite10	Frame4~7	417	207	Copy
Cast17	Sprite11	Frame4~7	234	264	Copy
Cast17	Sprite12	Frame4~7	295	264	Copy
Cast17	Sprite13	Frame4~7	356	264	Copy
Cast17	Sprite14	Frame4~7	417	264	Copy
Cast18	Sprite15	Frame4~11	394	43	Background Transport
Cast19	Sprite16	Frame4~11	392	68	同 Cast 18
Cast20	Sprite17	Frame4~11		306	同 Cast 18
Cast23	Channel	Frame8~8			

表 4-1(記憶大考驗之演員參數表)

完成後，Score 面板顯示如下：

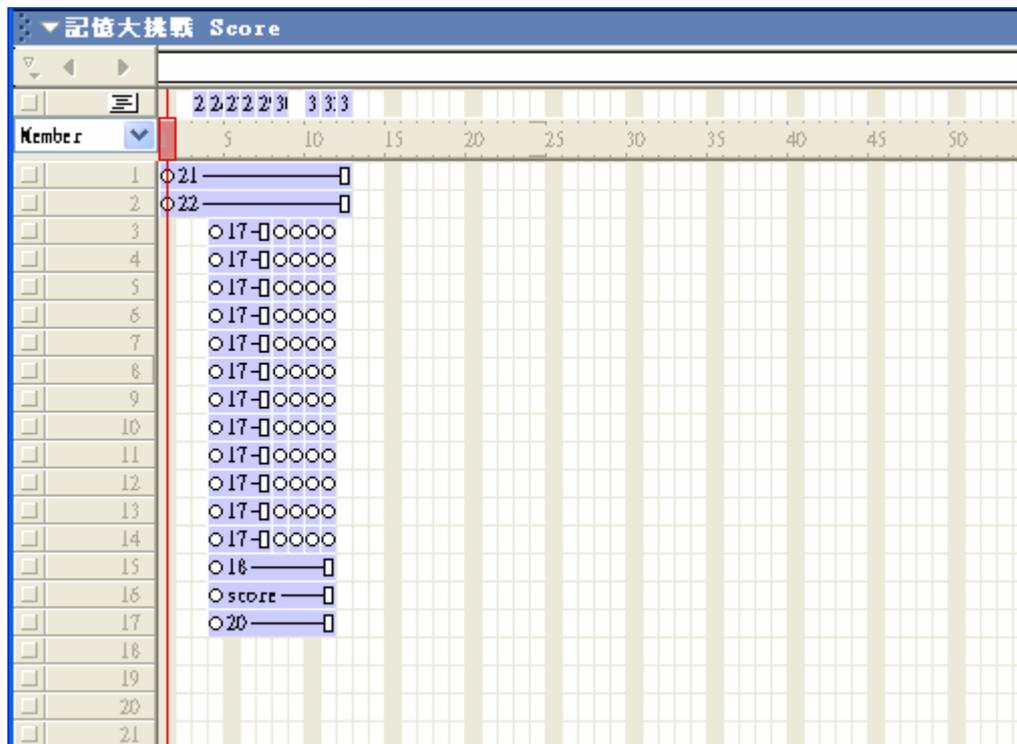


圖 4.2(完成後之記憶大考驗 Score 面板)

## Procedure

```

on startmovie //設定影片啓動播放時先執行的程式碼
  global startchannel,closepic,gamescore //宣告共用變數
  startchannel=3 //設定startchannel變數等於3
  closepic=17 //設定closepic變數等於17
  randompiclist //執行randompiclist程式
  gamescore=0 //設定gamescore變數等於0
  put gamescore into field "score"//設定演員 "score"的內容等於gamescore
end

on randompiclist //定義下面的程式碼的名稱randompiclist
  global piclist //宣告piclist為共用變數
  piclist=[1,2,3,4,5,6,7,8,9,10,11,12,13,14]//設定piclist變數的內容
  total=14 //設定total變數的內容等於14
  m=0
  repeat while m<total //這段程式會將piclist變數內容隨機排列
    m=m+1
    n=random(total)
    x=getat(piclist,n)
    y=getat(piclist,m)
    setat piclist,m,x
    setat piclist,n,y
  end repeat
end

on getgamelist //定義下面程式碼的名稱getgamelist
  global gamelist,piclist //宣告gamelist,piclist為共用變數
  gamelist=[1,1,2,2,3,3,4,4,5,5,6,6] //設定gamelist變數的內容
  n=0
  repeat while n<6 //從piclist變數中取得6個數值
    n=n+1 //放入gamelist中，且每個數字放入兩個位置
    x=getat(piclist,n)
    setat gamelist,n*2-1,x
    setat gamelist,n*2,x
  end repeat
end

```

```
on randomgamelist //定義下面程式碼的名稱randomgamelist
  global gamelist //宣告gamelist為共用變數
  m=0
  repeat while m<12
    m=m+1
    n=random(12)
    x=getat(gamelist,n) //將gamelist變數中的數值隨機排列
    y=getat(gamelist,m)
    setat gamelist,m,x
    setat gamelist,n,y
  end repeat
end

on showgamepic //定義下面程式碼的名稱為showgamepic
  global gamelist,startchannel //宣告gamelist、startchannel為共用變數
  n=0
  repeat while n<12
    // 以gamelist變數中的數值做為sprite3~14的演員編號，並顯示在面板上
    c=startchannel+n
    n=n+1
    puppetsprite c,TRUE
    no=getat(gamelist,n)
    the castnum of sprite c to no
  end repeat
  updatestage
end

on resetgamepic //定義下面程式碼的名稱為resetgamepic
  global gamelist,startchannel //宣告gamelist,startchannel為共用變數
  n=0
  repeat while n<12
    c=startchannel+n
    n=n+1
    puppetsprite c,FALSE
  end repeat
```

```
    updatestage
end

on closegamepic //定義下面程式碼的名稱為closegamepic
  global startchannel,closepic //宣告startchannel,closepic為共用變數
  //宣告startchannel,closepic為共用變數
  n=0
  repeat while n<12
    c=startchannel+n
    //將sprite 3~14的演員編號換成17，使呈現圖形隱藏的的畫面
    n=n+1
    the castnum of sprite c to closepic
  end repeat
  updatestage
end

on checkfinish //定義下面程式碼的名稱為checkfinish
  global finishflag,gamepicflag //宣告finishflag,gamepicflag為共用變數
  finishflag=0 //設定finishflag變數等於0
  n=0
  repeat while n<12
    n=n+1 //這判斷所有的圖形是否已經翻開，若已翻開，則結束程式
    x=getat(gamepicflag,n)
    if x=0 or x=1 then exit
  end repeat
  finishflag =1
end

on open_game_pic //定義下面程式碼的名稱為open_game_pic
  global startchannel,gamelist //宣告startchannel,gamelist為共用變數
  c=the clickon //變數c等於滑鼠所按演員的Channel編號

  x=c-startchannel+1 //設定變數x等於c-startchannel+1，即圖形
  //在串列中的位置
  n=getat(gamelist,x) //變數n等於變數gamelist中的第幾個值
  the castnum of sprite c to n //設定sprite c的演員編號為
  //這樣畫面會將圖形翻開

end
```

```

on check_pic_status          //定義下面程式碼的名稱為check_pic_status
  global gamepicflag,startchannel,picstatus//宣告共用變數
  c=the clickon             //變數c等於滑鼠所按演員的Channel編號
  x=c-startchannel+1        //設定變數x等於c-startchannel+1,即
                             //圖形在串列中的位置
  picstatus=getat(gamepicflag,x)      //設定變數picstatus等於
                                     //gamepicflag變數中第x位置的值
end

on set_pic_status           //定義下面程式碼的名稱為set_pic_status
  global
  gamepicflag,startchannel,picstatus,click_status,firstpic,secpic
                                     //宣告gamepicflag等為共用變數
  c=the clickon             //變數c等於滑鼠所按演員的Channel編號
  x=c-startchannel+1
                             //設定變數x等於c-startchannel+1,即圖形在串列中的位置
  setat gamepicflag,x,1      //設定gamepicflag中第x位置為1
  if click_status=1 then    //檢查翻開圖形的狀態,是第1張或第2張
    firstpic=x              //若第1張,則設定變數firstpic等於x
  else                       //若不是
    secpic=x                //則設定secpic變數等於x
  end if
end

on reset_pic_status        //定義下面程式碼的名稱為reset_pic_status
  global gamepicflag,startchannel,picstatus//宣告為共用變數
  c=the clickon            //變數c等於滑鼠所按演員的Channel編號
  x=c-startchannel+1
                             //設定變數x等於c-startchannel+1,即圖形在串列中的位置
  setat gamepicflag,x,0
                                     //設定變數gamepicflag的第x位置的值為0
end
global firstpic,secpic,gamelist,gamepicflag//宣告為共用變數
global gamescore,startchannel,closepic    //宣告為共用變數
on scoring
  x=getat(gamelist,firstpic)
  y=getat(gamelist,secpic)

```

```
if x=y then //判斷變數x是否等於y
  setat gamepicflag,firstpic,2
  setat gamepicflag,secpic,2
  gamescore=gamescore+100
else
  setat gamepicflag,firstpic,0
  setat gamepicflag,secpic,0
  gamescore=gamescore-50
  close_2_pic
end if
put gamescore into field "score"
end
on close_2_pic
  delay 90 //延遲1.5秒
  a=firstpic+startchannel-1
  sprite(a).castnum=closepic
  b=secpic+startchannel-1
  sprite(b).castnum=closepic
  updatestage //更新舞台畫面
end
```



Output: 題目顯示遊戲啟動畫面，開牌三秒後將會蓋牌。

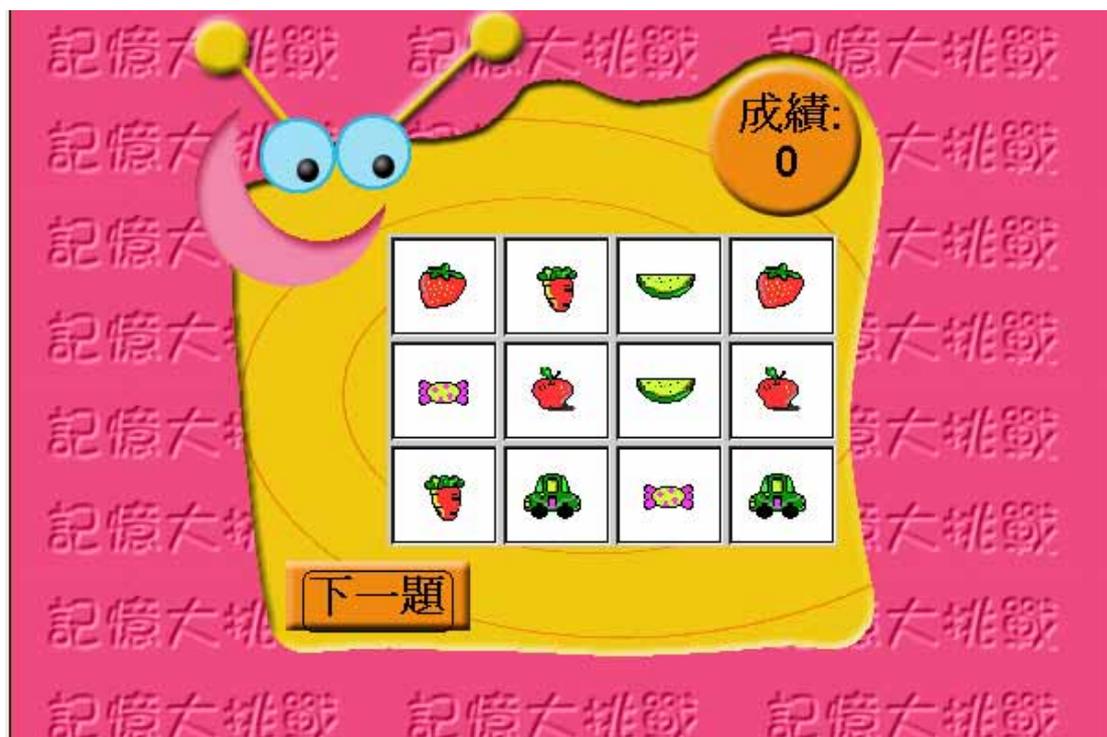
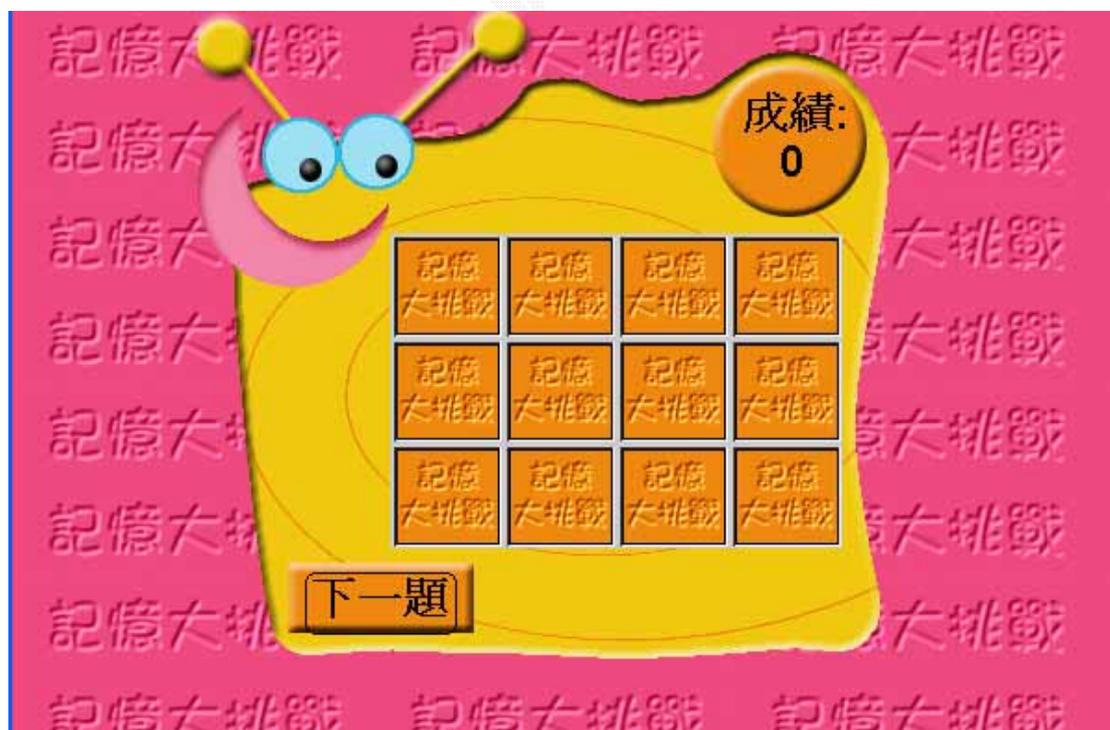


圖 4.3(遊戲開始圖)



此為三秒後的圖

圖 4.4(遊戲中)

答題後分數畫面

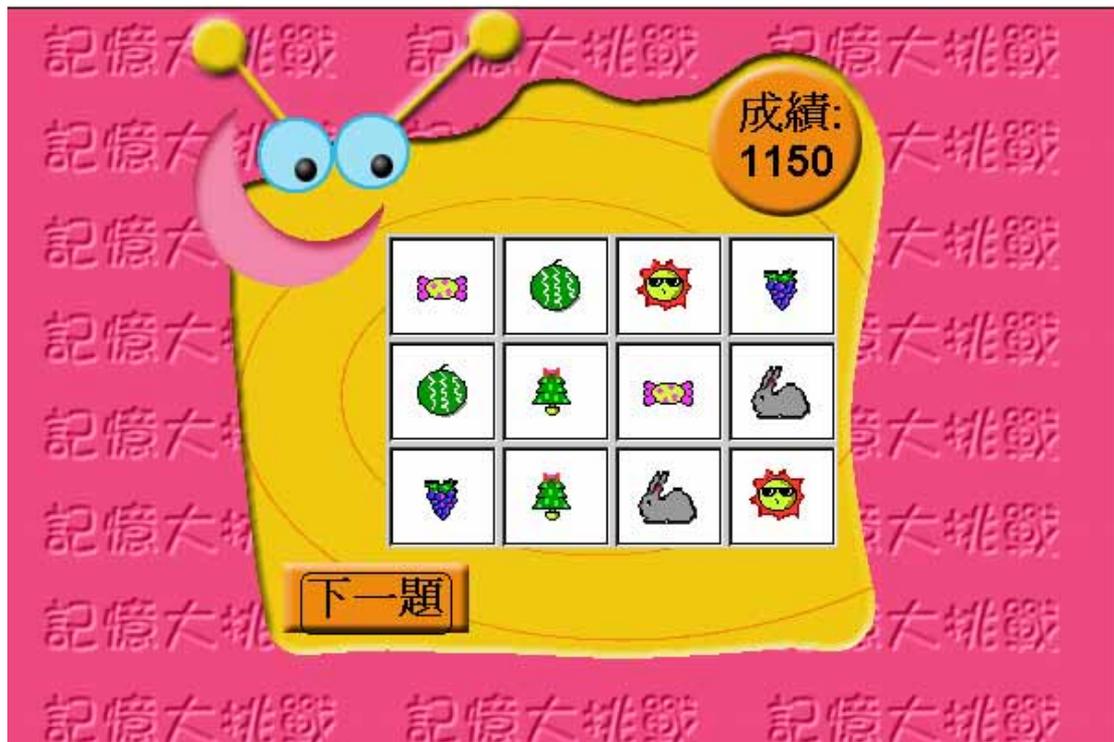


圖 4.5(得分畫面)

## 4.2 發射飛彈遊戲

下圖為”飛彈射擊遊戲”所需演員，演員編號依序如下

Cast 面板說明：

演員編號 1~2 為主要演員，遊戲開始後，Cast1 會有四個 sprite 做自由落下的動作，Cast2 可以由主人翁做左右水平的移動，並以滑鼠右鍵做發射飛彈的動作

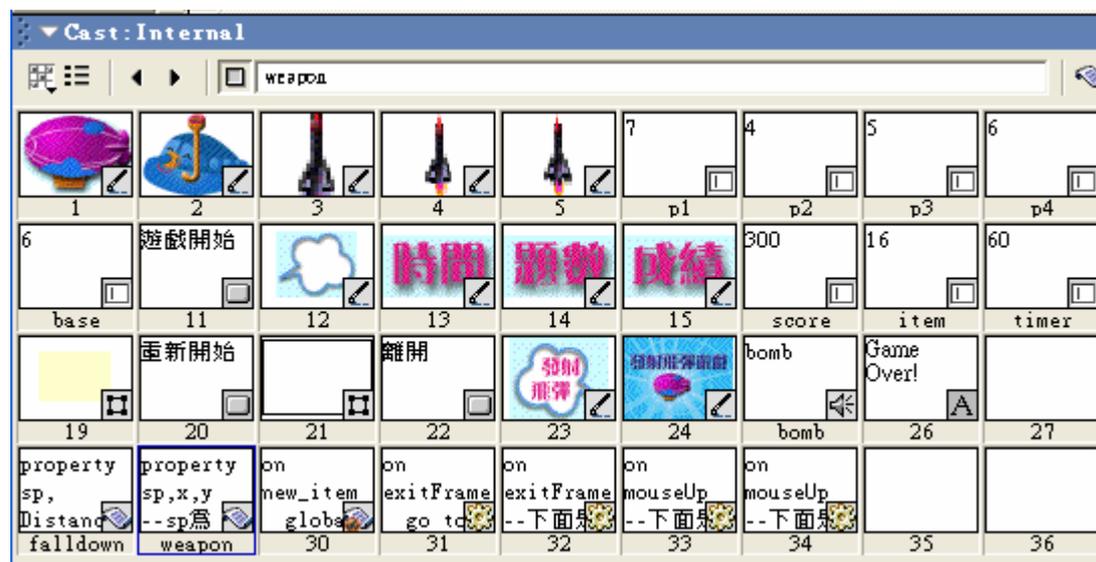


圖 4.6(發射飛彈遊戲 Cast 面板)

### 編排 score 面板

Cast	Sprite	放置範圍	X 值	Y 值	Ink 值
Cast24	Sprite2	Frame1~1	320	240	Copy
Cast11	Sprite3	Frame1~1	196	256	Background Transport
Cast1	Sprite5	Frame1~6	1039	58	同 Cast 11
Cast1	Sprite7	Frame1~6	1179	58	同 Cast 11
Cast1	Sprite9	Frame1~6	1331	58	同 Cast 11
Cast1	Sprite11	Frame1~6	1470	58	同 Cast 11
Cast6	Sprite6	Frame1~6	996	40	同 Cast 11
Cast7	Sprite8	Frame1~6	996	40	同 Cast 11
Cast8	Sprite10	Frame1~6	1050	41	同 Cast 11
Cast9	Sprite12	Frame1~6	1111	36	同 Cast 11
Cast2	Sprite13	Frame1~6	1043	454	同 Cast 11
Cast10	Sprite14	Frame1~6	1330	437	同 Cast 11
Cast3	Sprite15	Frame1~6	1375	413	同 Cast 11
Cast4	Sprite17~22	Frame1~6	1004	240	同 Cast 11
Cast15	Sprite28	Frame5~6	591	20	同 Cast 11
Cast14	Sprite30	Frame5~6	591	180	同 Cast 11
Cast13	Sprite32	Frame5~6	592	328	同 Cast 11
Cast12	Sprite29	Frame5~6	587	244	Copy
Cast12	Sprite31	Frame5~6	587	87	Copy
Cast12	Sprite33	Frame5~6	587	392	Copy
Cast16	Sprite34	Frame5~6	561	75	同 Cast 11
Cast17	Sprite35	Frame5~6	550	231	同 Cast 11
Cast18	Sprite36	Frame5~6	561	381	同 Cast 11
Cast19	Sprite38	Frame6~6	176	106	Copy
Cast21	Sprite39	Frame6~6	174	106	Copy
Cast20	Sprite40	Frame6~6	190	207	Copy
Cast22	Sprite41	Frame6~6	289	207	Copy
Cast26	Sprite42	Frame6~6	183	141	Copy
Cast23	Sprite2	Frame5~6	63	428	Copy
Cast27	Sprite1	Frame1~6	420	452	同 Cast 11

表 4-2(發射飛彈遊戲之演員參數表)

完成後，Score 面板顯示如下：

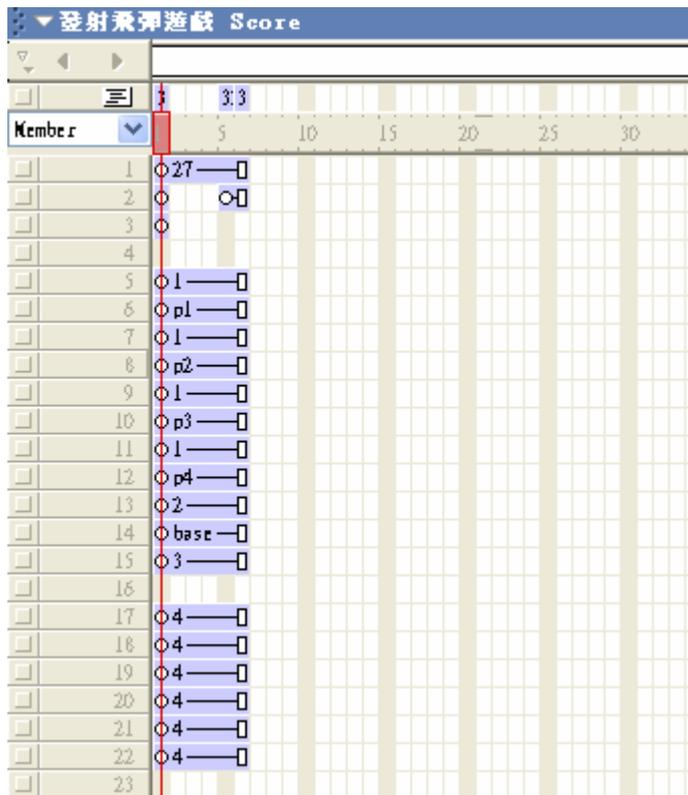


圖 4.7(完成後之 Score 面板之一)

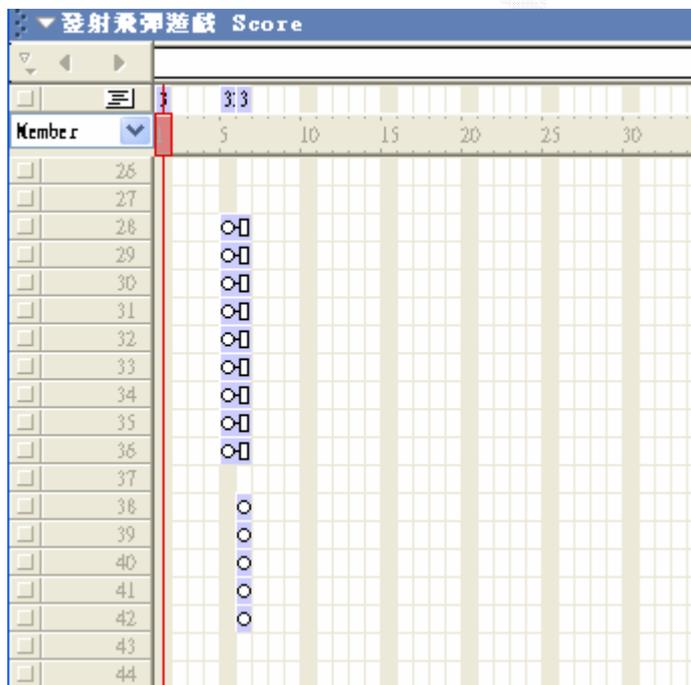


圖 4.8(完成後之 Score 面板之二)

## Input: cast1

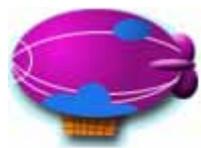


圖4.9(演員1)

### Procedure

```
property sp, Distance, Position, Shift
//設定屬性參數
//sp為sprite的編號、Distance為下降的距離、Position為水平座標位置、Shift
為左右位移的量
on new me,p1,p2,p3,p4
    sp=p1
    Distance=p2
    Position=p3
    Shift=p4
    return me
end

on falling me
    sprite(sp).puppet=TRUE
    sprite(sp+1).puppet= TRUE
    if sprite(sp).locV > 360 then
        sprite(sp).locV=1000
        sprite(sp+1).locV=1000
        new_item
    else
        sprite(sp).locV=sprite(sp).locV + Distance
        sprite(sp+1).locV=sprite(sp+1).locV + Distance
    end if
    if sprite(sp).locH<Position then
```

```
sprite(sp).locH=Position + Shift
//設定sprite sp的水平座標增加Shift
sprite(sp+1).locH=Position + Shift-5
//設定sprite sp+1的水平座標增加Shift-5
else
//若不小於Position則
sprite(sp).locH=Position - Shift
//sprite sp水平座標等於Position-Shift
sprite(sp+1).locH=Position - Shift-5
//sprite sp+1水平座標等於Position-Shift-5
end if
updatestage //更新舞台內容
end

on enlist me //下面是增加物件串列的程式
  append (the actorlist), me //將指定的串列加入actorlist中
end
```



Output: 建立cast1在舞台中的位置初始化，並使其自由落下及判斷是否超出舞台有無被擊中。

動作:

移動前:



圖 4.10(遊戲初始畫面)

移動後

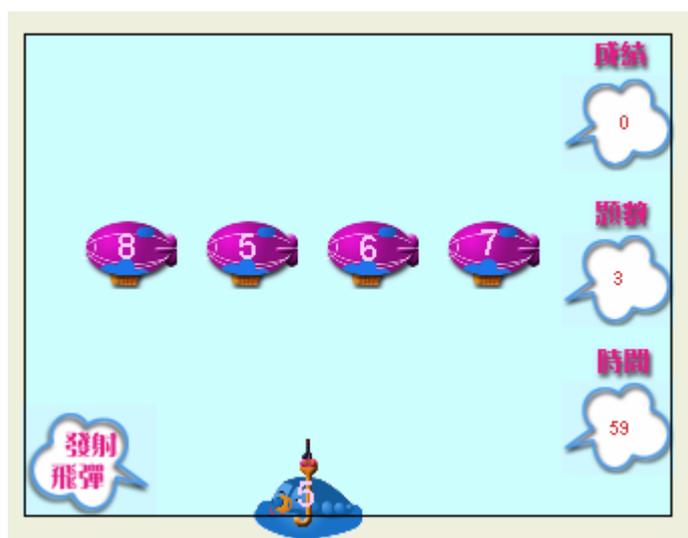


圖 4.11(遊戲中畫面)

## Input: cast2



圖 4.12(演員 2)

### Procedure:

```
property sp,x,y
on new me,r_sp
  global base_sp
  sp = r_sp + 16
  puppetsprite sp,true
  disappear = 0 //disappear等於0 (記錄物體是否被射中)
  x =sprite(base_sp).loch
  y =sprite(base_sp).locv - 20 //設定y等於sprite base_sp的垂直座標減20

  sprite(sp).loch = x //設定sprite sp的水平座標等於x
  sprite(sp).locv = y //設定sprite sp的垂直座標等於y
  return me
end

on shooting me //下面是發射砲彈的程式碼
  global weaponlist,flyspeed,total_score,disappear,answer_no
  y = y - flyspeed //設定y等於y-flyspeed
  sprite(sp).locv = y //設定sprite sp的垂直座標等於y
  disappear=0
  repeat with i = 1 to 4
  if answer_no=i and sprite sp intersects (3+i*2) then
    puppetsound "bomb" //發出射中目標的音效
    total_score = total_score + 100
    //設定得分變數total_score等於total_score+100
  put total_score into field "score"
  //設定演員score內容等於total_score
  disappear=1 //設定disappear等於1 (表示射中)
```

```
        new_item                                //取得新的題目
    end if
end repeat
n =( the castnum of sprite sp)+1
                                        //設定n等於sprite sp的演員編號加1
if n > 5 then n = 4
sprite(sp).castnum= n
if y < 0 or disappear then
    sprite(sp).loch = 1000
puppetsprite sp,false
n = sp-16
deleat(weaponlist,n)
addat(weaponlist,n,0)
end if
end
```



**Output：**使飛機本體發射飛彈，並判斷有無擊中目標若是有則會有音效即得分，並取得新題目。

### 發射飛彈前



圖 4.13(遊戲初始畫面)

### 發射飛彈中

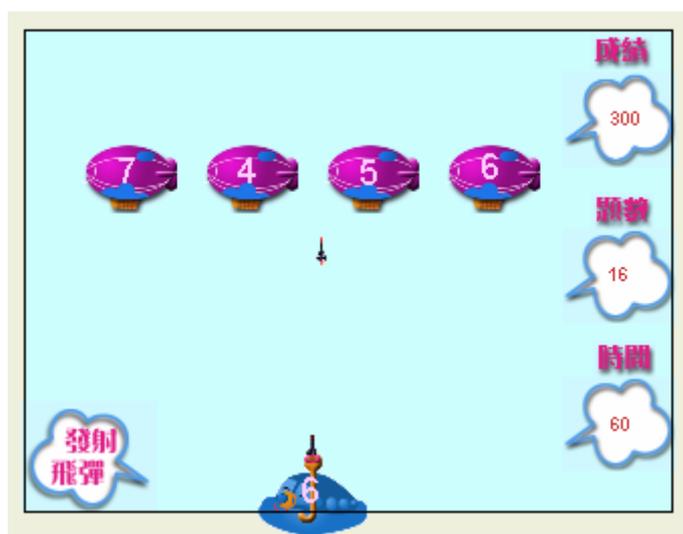


圖 4.14(遊戲中畫面)

### 4.3 選擇題設計

Input：演員

下圖為”選擇題”影片所需的演員，演員編號順序依序如下

**Cast 面板說明：**

演員編號 3~7 是為答對題目時出現的鼓勵演員，8~12 為答錯時的加油演員，13~26 為題目演員，27 為主背景演員。

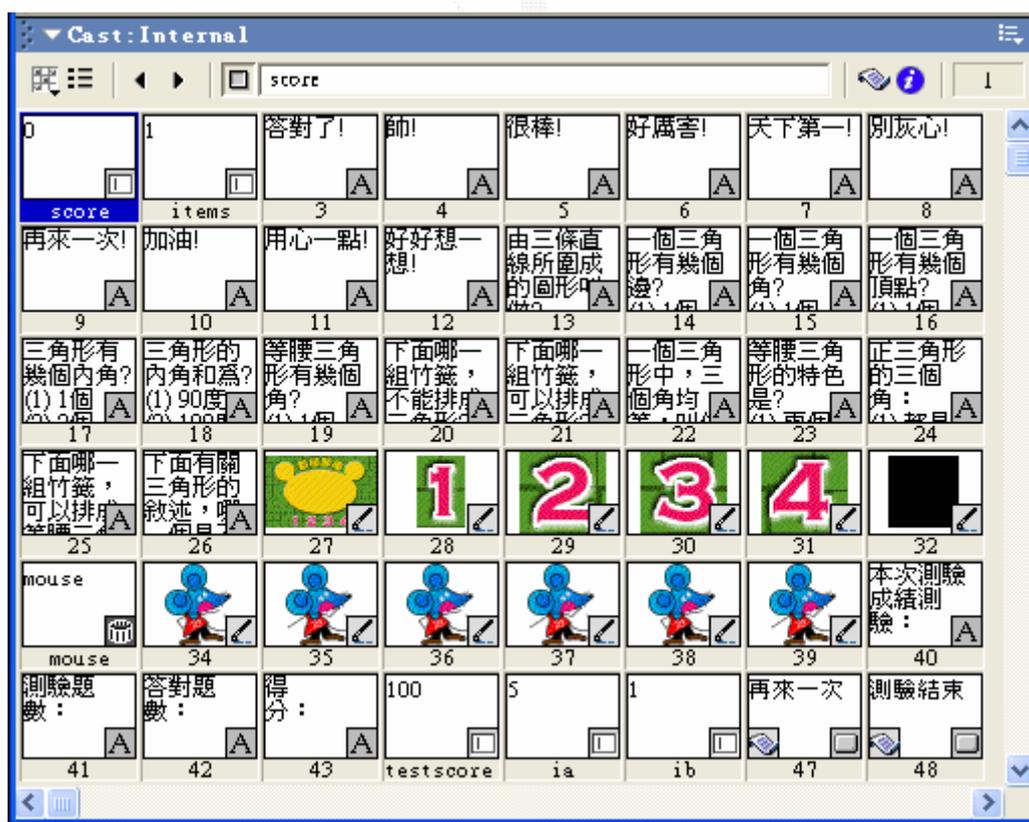


圖 4.15(選擇題設計之 Cast 面板)

## 編排的 Score 面板

將 Cast 面板的演員拖移到 Score 面板中，並依下列

表格作相關設定：

Cast	Sprite	放置範圍	X 值	Y 值	Ink 值
Cast27	Sprite1	Frame1~79	300	200	Copy
Cast33	Sprite2	Frame12~61	198	192	Copy
Cast3	Sprite3	Frame12~16	270	179	Background
Cast4	Sprite3	Frame17~21	270	179	同 Cast 3
Cast5	Sprite3	Frame22~26	270	179	同 Cast 3
Cast6	Sprite3	Frame27~31	270	179	同 Cast 3
Cast7	Sprite3	Frame32~36	270	179	同 Cast 3
Cast8	Sprite3	Frame37~41	270	179	同 Cast 3
Cast9	Sprite3	Frame42~46	270	179	同 Cast 3
Cast10	Sprite3	Frame47~51	270	179	同 Cast 3
Cast11	Sprite3	Frame52~56	270	179	同 Cast 3
Cast12	Sprite3	Frame57~61	270	179	同 Cast 3
Cast2	Sprite4	Frame12~78	71	70	同 Cast 3
Cast1	Sprite5	Frame12~78	435	70	同 Cast 3
Cast13~26	Sprite2	Frame65~78	164	116	同 Cast 3
Cast40	Sprite6	Frame62~62	160	117	同 Cast 3
Cast41	Sprite7	Frame62~62	182	168	同 Cast 3
Cast42	Sprite8	Frame62~62	182	203	同 Cast 3
Cast43	Sprite9	Frame62~62	182	238	同 Cast 3
Cast44	Sprite10	Frame62~62	320	238	同 Cast 3
Cast45	Sprite11	Frame62~62	320	168	同 Cast 3
Cast46	Sprite12	Frame62~62	320	203	同 Cast 3
Cast47	Sprite13	Frame62~62	193	269	同 Cast 3
Cast48	Sprite14	Frame62~62	300	269	同 Cast 3
Cast28,32	Sprite6,10	Frame63~78	160	365	Copy,Not Ghost
Cast29,32	Sprite7,11	Frame63~78	249	365	Copy,Not Ghost
Cast30,32	Sprite8,12	Frame63~78	337	365	Copy,Not Ghost
Cast31,32	Sprite9,13	Frame63~78	432	365	Copy,Not Ghost

表 4. 3(選擇題設計之演員參數表)

完成後，Score 面板顯示如下：

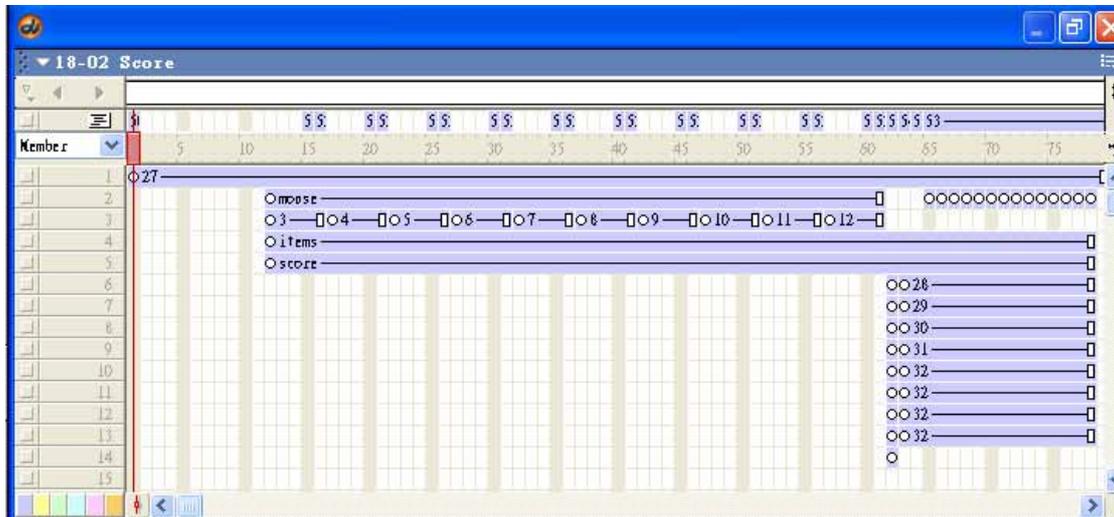


圖 4.16(完成後之 Score 面板)

## Procedure

```

on startmovie                                     //影片開始播放的事件程式
  resetitems                                     //執行resetitems程式
end

on resetitems                                     //這個程式將測驗資料歸零
  global itemno,totalitem,totalscore,correctitems //宣告公用變數
  global itemlist,answerlist,frameno
  itemlist=[65,66,67,68,69,70,71,72,73,74,75,76,77,78]
                                                    //設定串列的內容，題目放置的frame
  answerlist=[1,3,3,3,3,2,3,2,4,3,1,4,4,4]      //設定每一題的答案
  itemlib=14                                     //設定題庫的題目數
  totalitem=5                                    //設定測驗的題目數
  itemno=0                                       //設定題號歸零
  correctitems=0                                 //設定正確題數歸零
  totalscore=0                                  //設定總成績歸零
  put totalscore into field "score"             //設定score演員的內容為總成績
  randomitem                                     //執行randomitem程式
end

on randomitem                                     //這個程式將題目隨機排列
  global nolist                                  //宣告公用變數
  nolist=[1,2,3,4,5,6,7,8,9,10,11,12,13,14]    //設定串列內容
  i=0                                            //設定i等於0
  repeat while i<14                             //設定重複執行的迴圈
    i=i+1                                        //設定i等於i+1
    ran=random(14)                              //設定ran等於1至14間的一個隨機數
    x=getat(nolist,i)                           //設定x等於nolist串列的第i個值
    y=getat(nolist,ran)                         //設定y等於nolist串列的第ran個值
    setat nolist,i,y                            //設定nolist串列的第i個值為y
    setat nolist,ran,x                          //設定nolist串列的第ran個值為x
  end repeat
end

on getitem                                       //randomitem程式，這個程式將題目隨機排列

```

```

global itemno,c_answer,nolist,answerlist,itemlist,frameno
                                                    //宣告公用變數
itemno=itemno+1
                                                    //設定itemno的值等於itemno+1
x=getat(nolist,itemno)
                                                    //設定x等於nolist串列的第itemno個值
frameno=getat(itemlist,x)
                                                    //設定frameno等於itemlist串列的第x個值
c_answer=getat(answerlist,x)
                                                    //設定c_answer等於answerlist串列的第x個值
put itemno into field "items"
                                                    //設定items演員的內容等於itemno
end

on checkanswer
                                                    //randomitem程式,這個程式將題目隨機排列
global totalscore,correctitems,answer,c_answer
                                                    //宣告公用變數
fdc=[12,17,22,27,32]
                                                    //宣告串列fdc的內容為正回饋的frame位置
fdw=[37,42,47,52,57]
                                                    //宣告串列fdc的內容為負回饋的frame位置

if answer = c_answer then
                                                    //判斷答案是否正確,若正確則執行下面的程式
totalscore=totalscore+100
                                                    //總分累加100
correctitems=correctitems+1
                                                    //正確題數加一
put totalscore into field "score"
                                                    //更新score演員的內容
put totalscore into field "testscore"
                                                    //更新testscore演員的內容

x=random(5)
                                                    //設定x的值為1至5間的隨機數
y=getat(fdc,x)
                                                    //設定y等於fdc串列的第x個值
go to frame y
                                                    //影片跳到frame y
else
x=random(5)
                                                    //設定x的值為1至5間的隨機數
y=getat(fdw,x)
                                                    //設定y等於fdw串列的第x個值
go to frame y
                                                    //影片跳到frame y
updatestage
end if
end

on select no
sprite(6).visible=false
                                                    //將sprite 6,7,8,9設定為隱藏
sprite(7).visible=false
sprite(8).visible=false

```

```
sprite(9).visible=false
case no of
  1: sprite(6).visible=true
  2: sprite(7).visible=true
  3: sprite(8).visible=true
  4: sprite(9).visible=true
end case
end
```

//判斷選項  
//將sprite 6為顯示狀態  
//將sprite 7為顯示狀態  
//將sprite 8為顯示狀態  
//將sprite 9為顯示狀態



Output :測驗畫面和回饋畫面顯示如下:

測驗畫面



圖 4.17(遊戲畫面)

回饋畫面



圖 4.18(答對畫面)



圖 4.19(答錯畫面)

## 4.4 填充題設計

下圖為”填充題”所需要的演員

Cast 面板說明：

演員編號 8~12 為答對時的出現的鼓勵演員，14~17 為答錯時出現的加油演員，18 為主要背景演員。

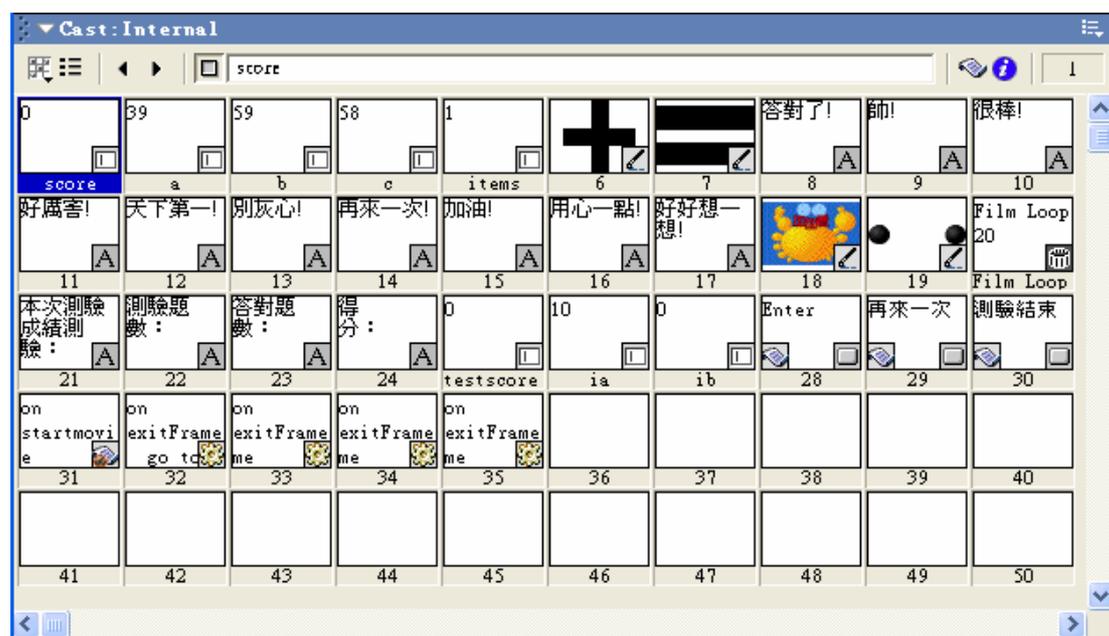


圖 4.20(填充題設計之 Cast 面板)

### 編排的 Score 面板

Cast	Sprite	放置範圍	X 值	Y 值	Ink 值
Cast18	Sprite1	Frame1~62	300	200	Copy
Cast19	Sprite2	Frame1~11	288	76	Background Transport
Cast6	Sprite3	Frame11~11	272	261	同 Cast 19
Cast3	Sprite4	Frame11~11	152	233	同 Cast 19
Cast2	Sprite5	Frame11~11	249	233	同 Cast 19
Cast7	Sprite6	Frame11~11	321	256	同 Cast 19
Cast4	Sprite7	Frame11~11	344	233	Copy
Cast28	Sprite8	Frame11~11	260	327	Copy
Cast8	Sprite3	Frame12~16	225	233	同 Cast 19
Cast9	Sprite3	Frame17~21	263	233	同 Cast 19
Cast10	Sprite3	Frame22~26	244	233	同 Cast 19
Cast11	Sprite3	Frame27~31	224	233	同 Cast 19
Cast12	Sprite3	Frame32~36	207	233	同 Cast 19
Cast13	Sprite3	Frame37~41	224	233	同 Cast 19
Cast14	Sprite3	Frame42~46	206	233	同 Cast 19
Cast15	Sprite3	Frame47~51	243	233	同 Cast 19
Cast16	Sprite3	Frame52~56	206	233	同 Cast 19
Cast17	Sprite3	Frame57~61	188	233	同 Cast 19
Cast22	Sprite3	Frame62~62	165	235	同 Cast 19
Cast23	Sprite4	Frame62~62	165	267	同 Cast 19
Cast24	Sprite5	Frame62~62	165	300	同 Cast 19
Cast21	Sprite6	Frame62~62	167	184	同 Cast 19
Cast26	Sprite7	Frame62~62	320	231	Copy
Cast27	Sprite8	Frame62~62	320	265	Copy
Cast25	Sprite9	Frame62~62	320	300	Copy
Cast5	Sprite10	Frame11~62	97	120	同 Cast 19
Cast20	Sprite11	Frame12~61	301	52	Copy
Cast19	Sprite11	Frame62~62	288	76	同 Cast 19
Cast1	Sprite12	Frame11~62	466	101	同 Cast 19
Cast29	Sprite13	Frame62~62	207	341	Copy
Cast30	Sprite14	Frame62~62	290	341	Copy

表 4-4(填充題設計之演員參數表)

完成後的 Score 面板顯示如下：

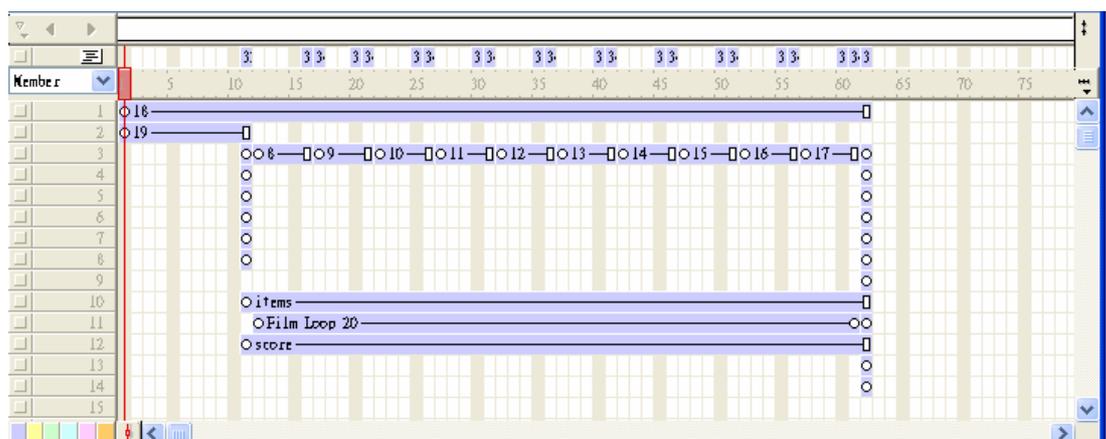


圖 4.21(完成後之 Score 面板)

## Procedure

```
on startmovie
  resetitems
  go to frame 11
end
```

```
on resetitems // resetitems程式用來將測驗資料歸零
  global items,totalitem,totalscore,correctitems // 宣告公用變數
  totalitem=10 // 設定測驗的題目數
  items=0 // 設定題號歸零
  correctitems=0 // 設定正確題數歸零
  totalscore=0 // 設定總成績歸零
  put totalscore into field "score" // 設定score演員的內容為總成績
  getitem // 執行getitem程式
end
```

```
on getitem // getitem程式可取得新的題目
```

```

global va,vb,totalitem,items
items=items+1 // 設定題號加一
put items into field "items" // 設定演員items的內容等於items的值
va=random(90)+9 // 設定va的值等於random(90)+9
put va into field "a" // 設定演員a的內容等於va的值
vb=random(90)+9 // 設定vb的值等於random(90)+9
put vb into field "b" // 設定演員b的內容等於vb的值
put "" into field "c" // 設定演員c的內容為空白
member("c").textsize=36 // 設定題號加一
end

on checkanswer // checkanswer程式用來檢查答案是否正確
global totalscore,correctitems // 宣告公用變數
fdc=[12,17,22,27,32] // 宣告串列fdc的內容為正回饋的frame位置
fdw=[37,42,47,52,57] // 宣告串列fdw的內容為負回饋的frame位置
global va, vb, vc // 宣告公用變數
vc=va+vb // 設定vc等於va+vb
if field "c" = vc then // 判斷答案是否正確，若是則執行下面的程式
    totalscore=totalscore+100 // 總分累加100
    correctitems=correctitems+1 // 正確題數加一
    put totalscore into field "score" // 更新score演員的內容
    x=random(5) // 設定x的值為1至5間的隨機數
    y=getat(fdc,x) // 設定y等於fdc串列的第x個值
    go to frame y // 影片跳到frame y
else // 答案不正確，則執行下面的程式
    x=random(5) // 設定x的值為1至5間的隨機數
    y=getat(fdw,x) // 設定y等於fdw串列的第x個值
    go to frame y // 影片跳到frame y
    updatestage // 更新舞台內容
end if
end

```

## 測驗畫面和回饋畫面

### 測驗畫面



圖 4.22(遊戲畫面)

回饋畫面



圖 4. 23(答對畫面)



圖 4. 24(答錯畫面)

## 4.5 趣味連連看

下圖為”趣味連連看”所需演員

趣味連連看之 Cast 面板

### Cast 面板說明

演員編號 1~16 為主要演員,17 為完成圖演員。

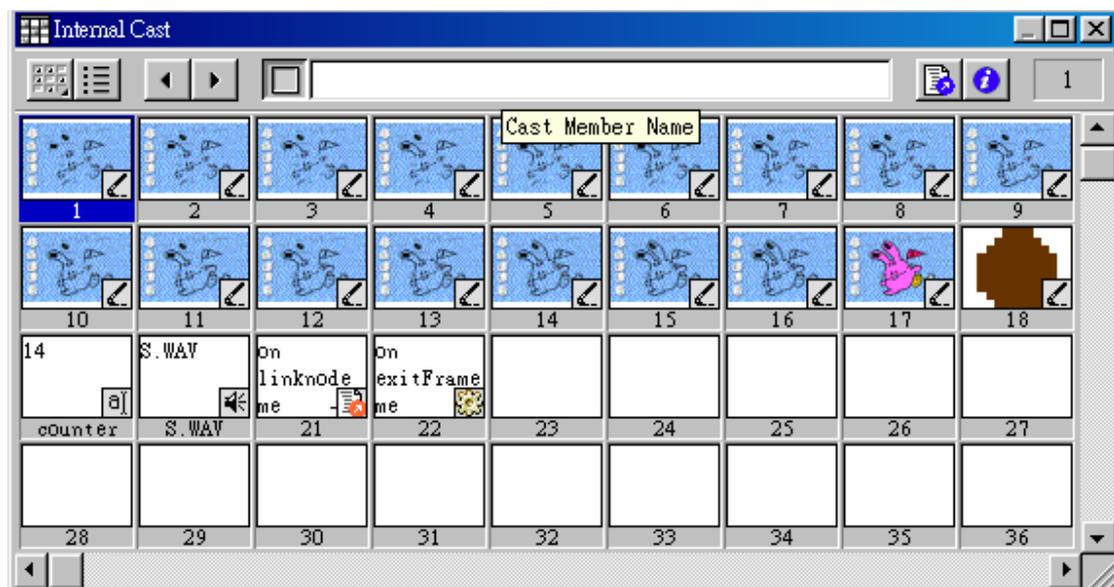


圖 4.25(趣味連連看之 Cast 面板)

### 編排的 Score 面板

Cast	Sprite	放置範圍	X 值	Y 值	Ink 值
Cast1	Sprite1	Frame1~1	256	171	Copy
Cast17	Sprite1	Frame18~45	256	171	Copy
Cast18	Sprite2	Frame1~15	211	136	Matte
Cast18	Sprite3	Frame1~15	128	98	Matte
Cast18	Sprite4	Frame1~15	125	128	Matte
Cast18	Sprite5	Frame1~15	188	157	Matte
Cast18	Sprite6	Frame1~15	185	206	Matte
Cast18	Sprite7	Frame1~15	198	269	Matte
Cast18	Sprite8	Frame1~15	225	286	Matte
Cast18	Sprite9	Frame1~15	245	286	Matte
Cast18	Sprite10	Frame1~15	307	258	Matte
Cast18	Sprite11	Frame1~15	328	207	Matte
Cast18	Sprite12	Frame1~15	316	179	Matte
Cast18	Sprite13	Frame1~15	292	154	Matte
Cast18	Sprite14	Frame1~15	268	135	Matte
Cast18	Sprite15	Frame1~15	189	47	Matte
Cast18	Sprite16	Frame1~15	230	129	Matte
Cast19	Sprite17	Frame1~15	272	25	Copy

表 4-5(趣味連連看之演員參數表)

## 完成後之 Score 面板

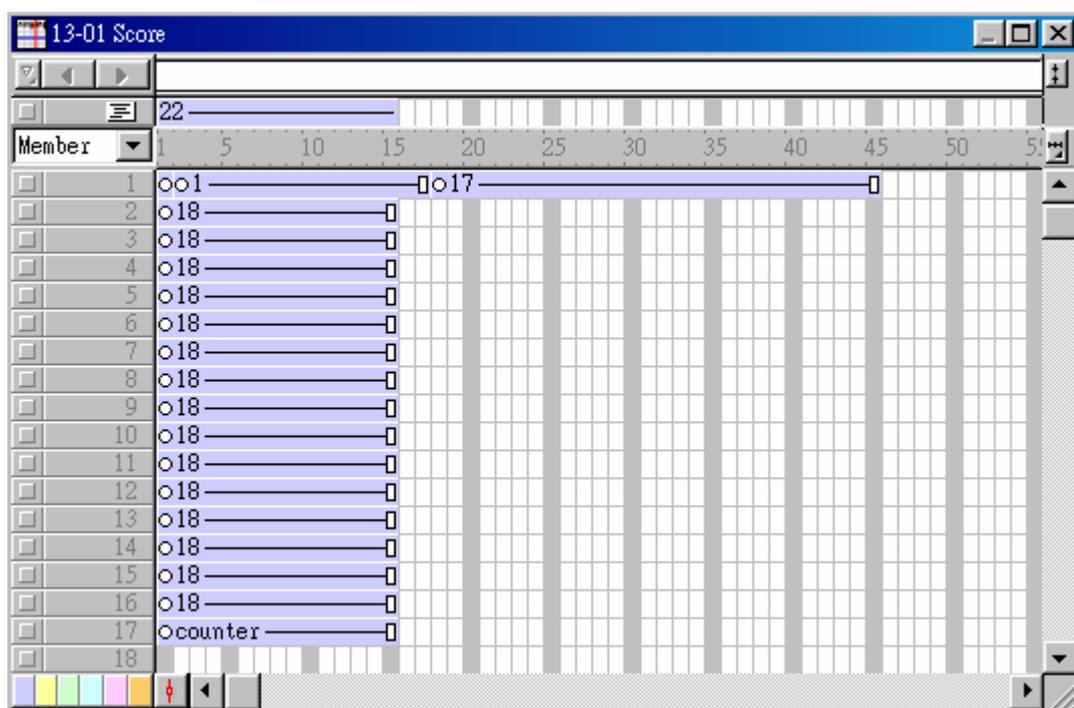


圖 4.26(完成後之 Score 面板)

## Procedure

on linknode me

--定義下面程式碼的名稱為linknode

```
if rollover(the frame+1)=FALSE then go to the frame
```

```
--判斷滑鼠是否經過(the frame+1) Channel上的演員，若是則播放音效
```

```
if rollover(the frame+1)=True then puppetSound "s.wav"
```

```
x=the frame
```

```
--設定x的值等於the frame
```

```
put x-1 into field "counter"
```

```
end linknode
```

### 測驗畫面及回饋畫面



圖 4.27(遊戲初始畫面)



圖 4.28(遊戲中畫面 counter=3)



圖 4.29(遊戲完成畫面)

## 4.6 鍵盤打字練習

下圖為”鍵盤打字練習”所需演員，演員編號依序如下

鍵盤打字練習之 Cast 面板

### Cast 面板說明

演員編號 13~38 是字母演員會一直出現在螢幕上，40 為題目字串演員。

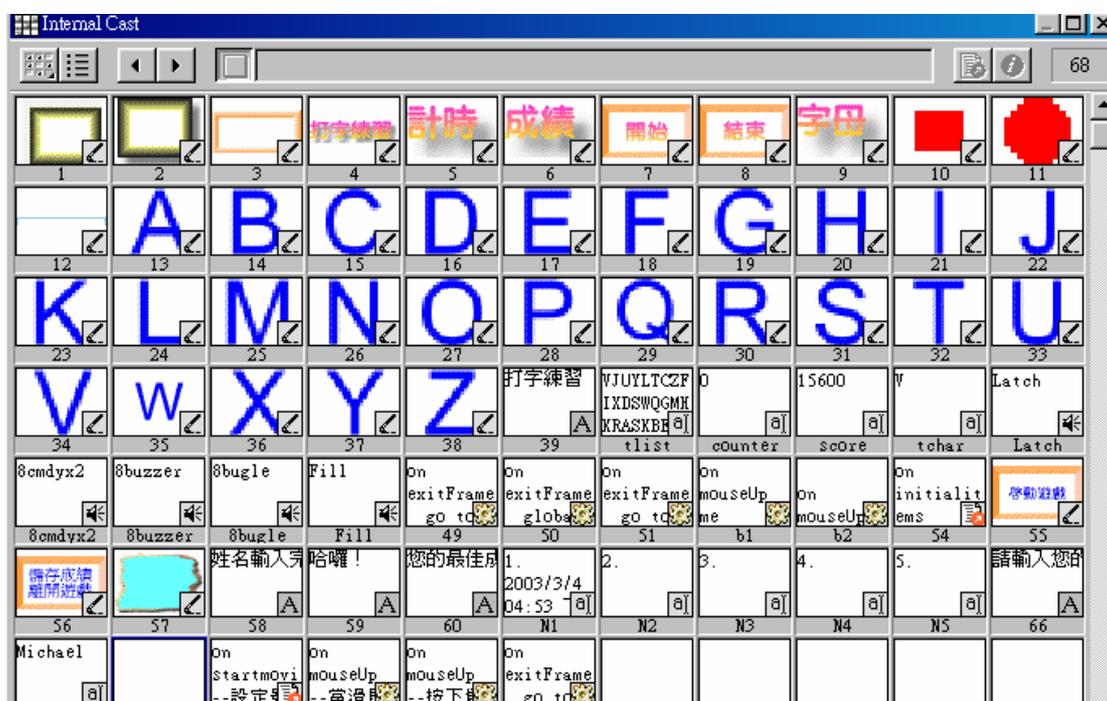


圖 4.30(鍵盤打字練習之 Cast 面板)

### 編排 Score 面板

Cast	Sprite	放置範圍	X 值	Y 值	Ink 值
Cast66	Sprite3	Frame1~1	58	176	Background
Cast67	Sprite4	Frame1~1	254	179	Background
Cast55	Sprite5	Frame1~1	337	396	Copy
Cast58	Sprite6	Frame1~1	175	260	Background
Cast57	Sprite98	Frame30~30	314	246	Background
Cast59	Sprite99	Frame30~30	165	148	Background
Cast67	Sprite100	Frame30~30	217	144	Background
Cast60	Sprite101	Frame30~30	304	147	Background
Cast61	Sprite102	Frame30~30	173	183	Background
Cast62	Sprite103	Frame30~30	173	216	Background
Cast63	Sprite104	Frame30~30	173	244	Background
Cast64	Sprite105	Frame30~30	173	275	Background
Cast65	Sprite106	Frame30~30	173	307	Background
Cast56	Sprite107	Frame30~30	510	430	Copy
Cast68	Sprite1	Frame1~30	2	462	Copy

表 4-6(鍵盤打字練習之演員參數表)

### 完成後之 Score 面板

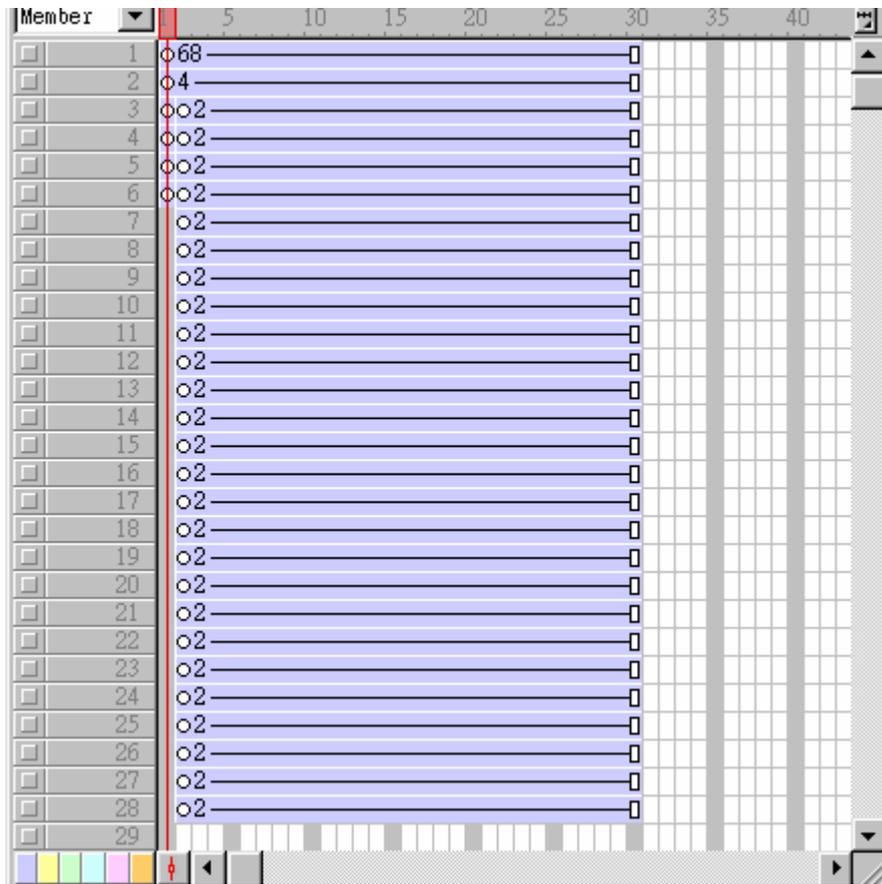


圖 4.31(完成後之 Score 面板之一)

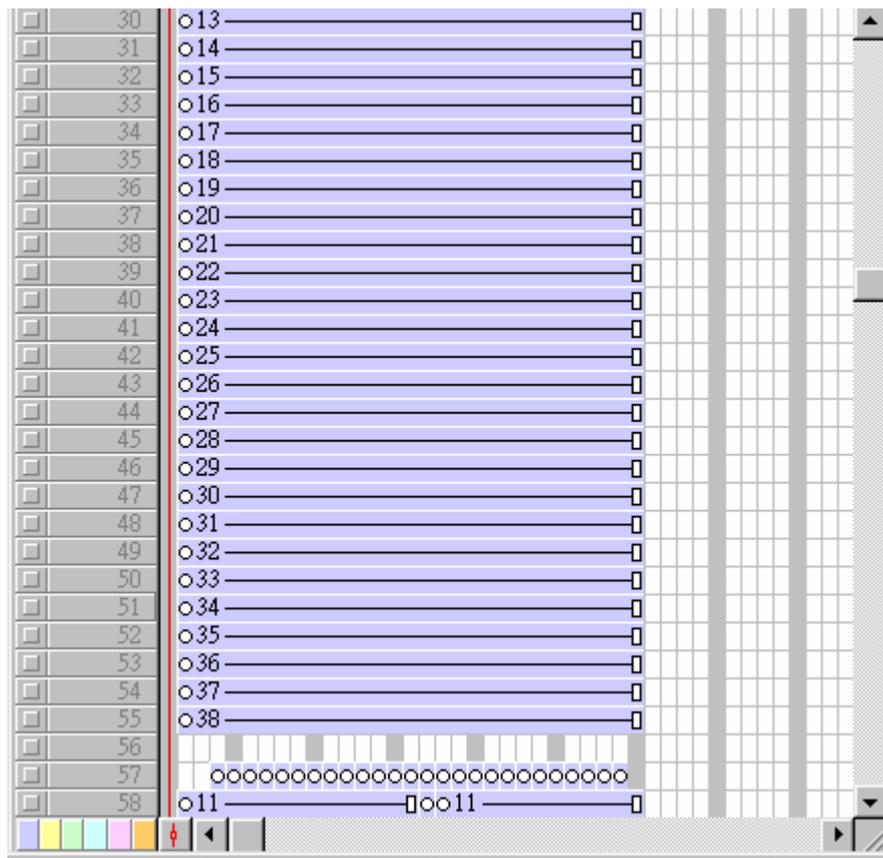


圖 4.32(完成後之 Score 面板之二)

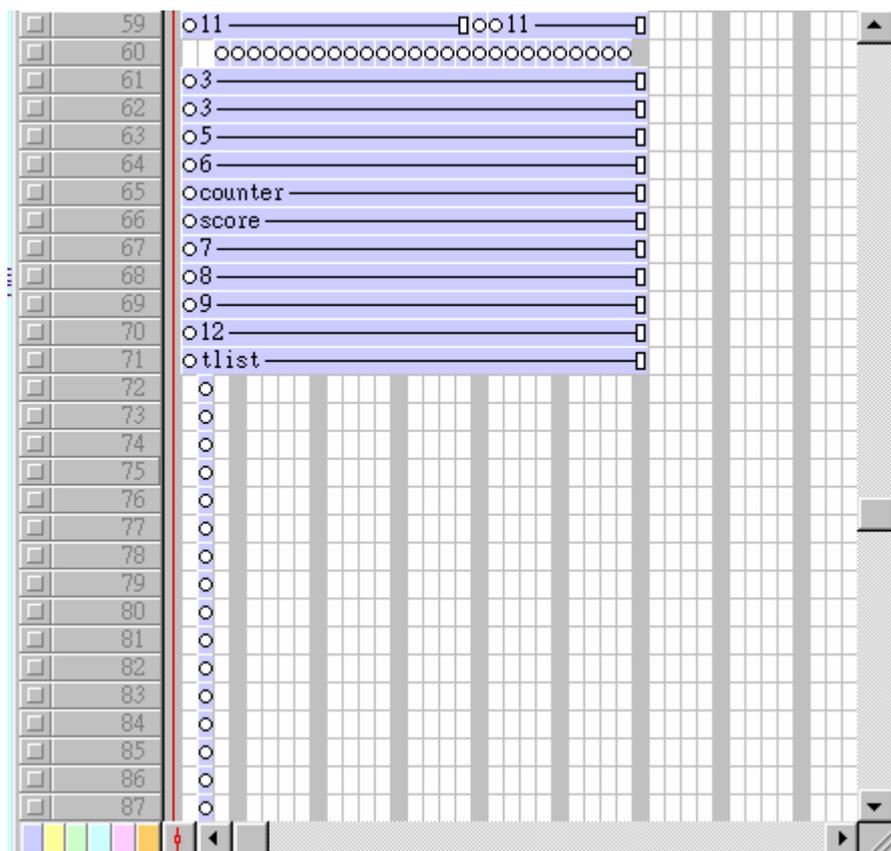


圖 4.33(完成後之 Score 面板之三)

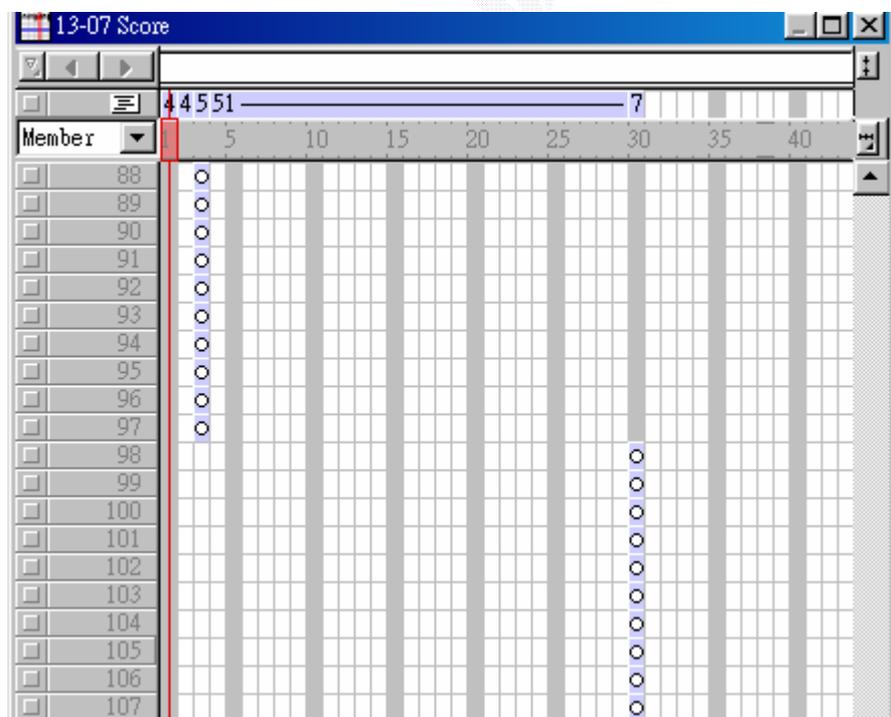


圖 4.34(完成後之 Score 面板之四)

## Procedure

```
on initialitems
```

```
  global alpha
```

```
  alpha="ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHI J KLMNOPQRSTUVWXYZABCDEFGHI J  
  KLMNOPQRSTUVWXYZABCDEFGHI J KLMNOPQRSTUVWXYZABCDEFGHI J KLMNOPQRSTU  
  BCDEFGHI J KLMNOPQRSTUVWXYZABCDEFGHI J KLMNOPQRSTUVWXYZABCDEFGHI J  
  STUVWXYZABCDEFGHI J KLMNOPQRSTUVWXYZABCDEFGHI J KLMNOPQRSTUVWXYZ"
```

```
end
```

```
on randomitems
```

```
  global alpha
```

```
  n=0
```

```
  repeat while n<260
```

```
    n=n+1
```

```
    ran=random(26)+integer((n-1)/26)*26
```

```
    x=char ran of alpha
```

```
    y=char n of alpha
```

```
    put x into char n of alpha
```

```
    put y into char ran of alpha
```

```
  end repeat
```

```
end
```

```
on getitems
```

```
  global alpha,t_position
```

```
  t_position=t_position+1
```

```
  put chars(alpha,t_position,t_position) into field "tchar"
```

```
  put chars(alpha,t_position,t_position+25) into field "tlist"
```

```
  sprite(chartonum(field "tchar")+7).visible=TRUE
```

```
end
```

```
on invisible_lighting
```

```
  n=71
```

```
  repeat while n<97
```

```
n=n+1  
sprite(n).visible=FALSE  
end repeat  
end
```

## 測驗畫面及回饋畫面

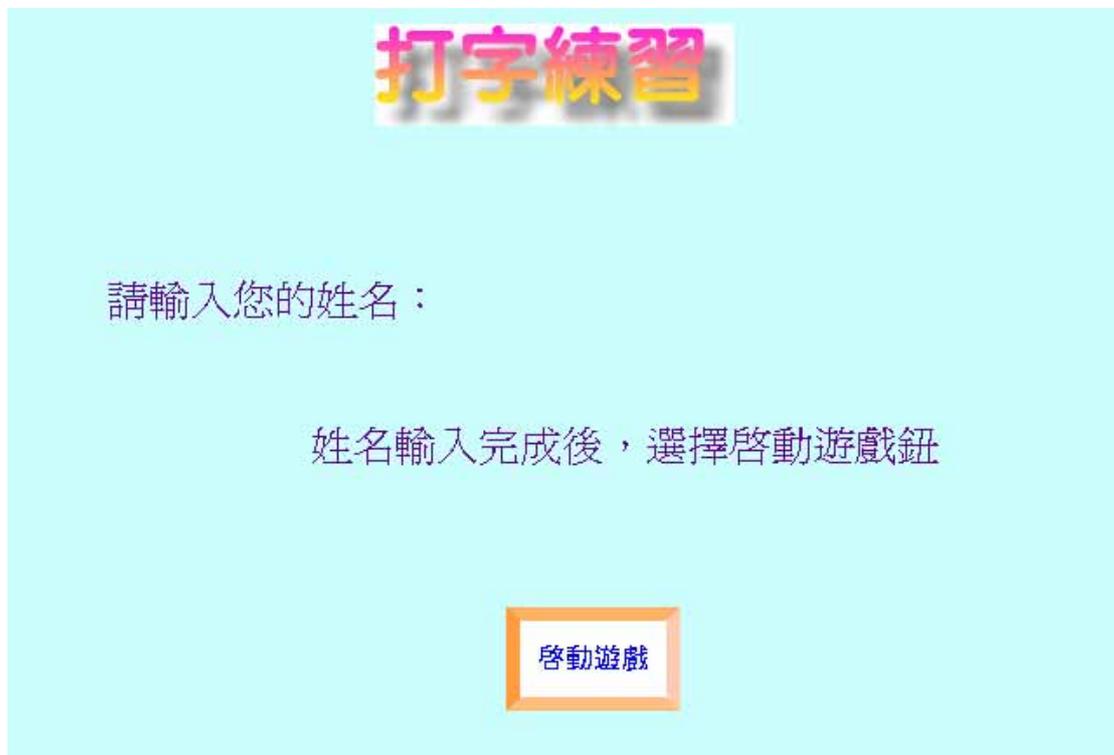


圖 4.35(遊戲初始畫面登入姓名)



圖 4.36(遊戲開始畫面)



圖 4.37(遊戲結束畫面~排行版)

## 第五章 系統評估

### 5.1 記憶大考驗

1. 優點: 這個遊戲可以使小朋友們不會感到枯燥乏味，雖然題型都一樣，但題目一直在變化，藉由遊戲的過程中，來增加瞬間記憶力，進而增加運用頭腦的機會。
2. 缺點: 因為當初在定儲存圖形的時候定死了，所以儲存圖案的格子數目顯的太少，相對的使得題型的變化量不大，而且不可以由玩家自己選擇難易度。

### 5.2 飛彈射擊遊戲

1. 優點: 這個飛彈射擊遊戲屬於比較活的，充滿機動性，操控性佳，很容易引起小朋友們的興趣，當目標擊中時又有音效的功能，比較不容易玩膩，再加上瞬間要對帶有數字的目標作確認，也要花點腦筋，可以增加反應能力。
2. 缺點: 因為背景畫面停滯，導致飛機看起來像是沒有移動，不能前進飛行，只能在 X 軸上移動，作水平動作，不能在 Y 軸上移動，作 2D 的飛行是一大缺失。

### 5.3 選擇題設計

1. 優點: 這個遊戲專題是針對目的第二項由於目前的小朋友對於數學尤感非常排斥所特別設計的，選擇題是多媒體教學軟體經常應用的技巧，藉著題目引發小朋友們對數學圖形的認識，並可以建立良好的基本幾何圖形觀念。
2. 缺點: 因為題目是屬於 Cast 裡面的演員，所以我們設計出來的題目可能會比較死板，而且只有十四題，也比較少，導致題目一下子就會被重複問到。

### 5.4 填充計算設計

1. 優點: 這一項也是針對數學而做的專題，雖然沒有依學生程度的不同設計出不同的題目，但是可以跳脫時空上的限制，即使小朋友們在家也能輕鬆的學習。由於題目是亂數產生的，所以題目比較活，所以題目比較不會去重複到，而且答對或答錯的回饋畫面都不一樣，還有累積答對數的累計。
2. 缺點: 我們只有設計加法問題，並沒有把減法乘法除法考慮進去，這是比較失敗的地方，使的題目也比較

單調。

## 5.5 趣味連連看

1. 優點:可以讓初期剛學數字的幼兒,學習數字的連續性藉由圖案和數字的呈現來學習。
2. 缺點:圖案還需多變化,還需再加強多設計幾種圖案。

## 5.6 鍵盤打字練習

1. 優點:容易上手藉由眼到手到來增強對鍵盤的熟練度,並認識基本的 26 個英文字母。
2. 缺點:缺少背景動畫,如果背景動畫能做出來就更完美了。

## 第六章 心得感想與展望

### 6.1 心路歷程

當初會想選用這個題目的起因，是因為在大賣場上看到許多的幼教軟體，好像市場不錯，且大人們對小孩的教育都很肯花錢去投資，所以我們試著朝這個方向做做看。這次定的專題題目”互動式幼兒教學軟體製作”。

在製作過程中所碰到的困難包括了：

1. 所用的語言大不相同我們習慣的語言 C，但在這裡是使用 Lingo，所以花了不少時間去翻翻有關 Lingo 的語法。
2. 美工的部分剛開始不知道要如何下手，原本還想要自己做美工圖案，但是沒有那個美工基礎，要用筆手畫出來都有點困難了，更何況是用電腦美工。所以最後我們決定先找一些現成的圖案，再來依劇本的需求角色作篩選。
3. 所用的軟體 DirectX 也是新的嘗試，它的基本功能有點類似 Flash，把演員，舞台作一些時間點上的線性分配。

## 6.2 未來展望

我們做的軟體，我覺得還可以多增添一點背景音樂和音效，因為時間的關係，我們沒再去多找一些遊戲的背景音樂和音效增添在這些遊戲面，還有在遊戲的變化性少的情況下更顯得有些枯燥乏味容易玩膩。這些地方是可以改進的。

在遊戲本身，我覺得飛彈射擊遊戲還可以多加一點飛機 2D 的飛行操作，來增加飛機的操控性和流暢性，不過增添 2D 的功能可能又要多看點書。

希望以後學弟妹可以幫我們完成。

## 6.3 感想

終於如釋重負了，有解脫的感覺，雖然在製作過程中有時會和 parter 意見不合或是時間橋不定，遲到...等造成一些小爭執的種種過程，但是兩個人一起合作完成一件事的感覺還真不錯。不過對於即將畢業還是有些不捨，四年真的好快，再這邊也警惕學弟妹要好好把握四年的大學生活，共勉之。

## 6.4 參考文獻

[1]李天義 陳守莊 ,Lingo 應用大全 ,第三波, 2001,pp.9-1~12

[2]鄧文淵,Flash 5 遊戲總動員,松崗,2001, pp1-30

[3]吳權威,DirectorMX 多媒體與動畫實務,2004,ppA-1~A-8

