

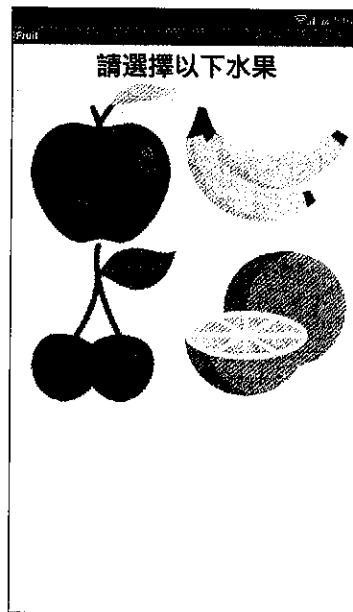
4-2-2 第二類：觸控操作與多媒體互動

本書範例題目內容為認證題型與命題方向之示範，正式測驗試題不以範例題目為限。

201. 水果單字卡 易 中 難

1. 題目說明：

請開啓 **IND02.aia** 專案，設計「水果單字卡」程式。點選畫面上的水果圖示，除了會發出對應的水果英文單字的音效，在下方的標籤也會顯示這個水果的單字。請依下列題意完成作答，並製成封裝檔 **IND02.apk**，作答完成，請將 aia 專案檔及 apk 封裝檔皆下載並儲存於 C:\ANS.CSF\原專案的資料夾內。



<參考圖>

2. 設計說明：

- (1) 專案中已上傳四個水果的圖片及英文單字音效檔案。
- (2) 在標題下加入一個兩欄兩列的 TableArrangement 表格配置。



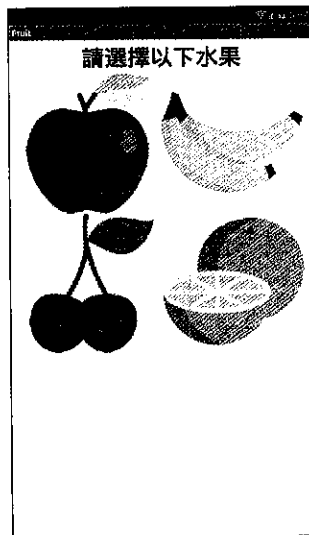
- (3) 在 TableArrangement 中放入已提供的四個 Button 元件，分別設定 image 欄位為四個水果圖片，按鈕皆不顯示文字。
- (4) 設定用來播放音效的 Sound 元件，並依已上傳的音效檔進行對應，對應表如下：

apple.png Apple.mp3	banana.png Banana.mp3
cherry.png Cherry.mp3	orange.png Orange.mp3

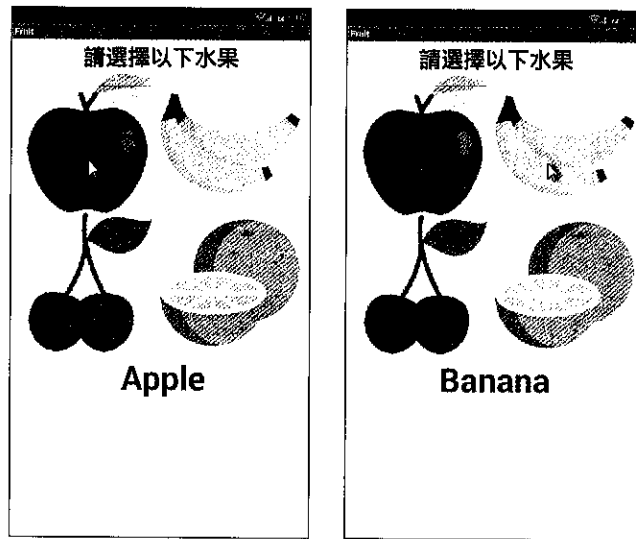
- (5) 點選顯示為水果圖片的按鈕時要能依對應表播放音效，並在下方 Label 元件顯示該水果的英文單字代表正確點選。如點選蘋果圖片按鈕時，播放 **Apple.mp3** 的音效，Label 元件顯示 **Apple**。

3. 執行結果參考畫面：

- (1) 程式初始畫面。



- (2) 點選水果按鈕時要能播放該英文單字的音效，並於下方的標籤顯示目前的英文單字，代表正確點選。



4. 自行測試程式是否達成下列結果：

- (1) 表單上各物件與參考圖相同，屬性設定正確。
- (2) 點選畫面各水果按鈕時，會發出對應的音效檔。
- (3) 點選畫面各水果按鈕時，會於下方的標籤顯示正確的英文單字。

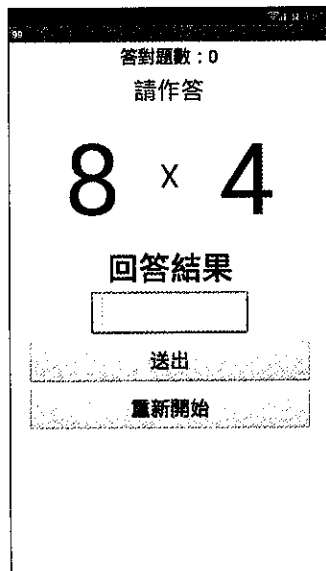
5. 評分項目：

項	目	配 分	得 分
(1)	表單上各物件與參考圖相同，屬性設定正確	4	
(2)	點選畫面各水果按鈕時，會發出對應的音效檔	16	
(3)	點選畫面各水果按鈕時，會於下方的標籤顯示正確的英文單字	10	
總	分	30	

202. 挑戰 99..... 易 中 難

1. 題目說明：

請開啓 **IND02.aia** 專案，設計「挑戰 99」程式。程式一啓動會由程式隨機在畫面中出一個九九乘法的題目，在輸入答案後按下「送出」按鈕，程式會自動比對答案。請依下列題意完成作答，並製成封裝檔 **IND02.apk**，作答完成，請將 aia 專案檔及 apk 封裝檔皆下載並儲存於 C:\ANS.CSF\原專案的資料夾內。



<參考圖>

2. 設計說明：

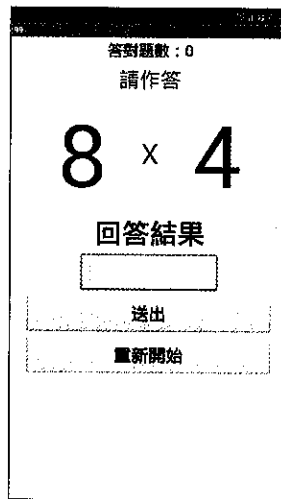
- (1) Label 元件名稱「NO1」為被乘數，Label 元件名稱「NO2」為乘數，被乘數與乘數的數字範圍都限定在 2~9，每次隨機出現，請依據其相乘結果，在下方 TextBox 元件中輸入答案。
- (2) 輸入答案的 TextBox 元件設定只能輸入數字。
- (3) 輸入答案後按下「送出」按鈕，程式會判斷其相乘結果是否正確，如果正確會累計更新上方的答對題數，並修改【回答結果】的 Label 元件，Text 顯示為【答對了】；如果答案錯誤，不更新答對題數，Text 顯示為【答錯了】。



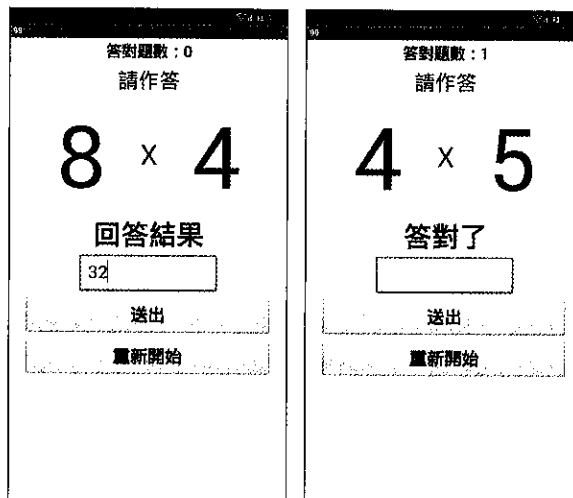
- (4) 答題結果不論對錯，都會清空 TextBox 元件並繼續進行出題。
- (5) 按下「重新開始」按鈕，歸零上方的答對題數，回答結果的訊息回復初始文字，清空 TextBox 元件並重新出題。

3. 執行結果參考畫面：

- (1) 程式起始畫面。

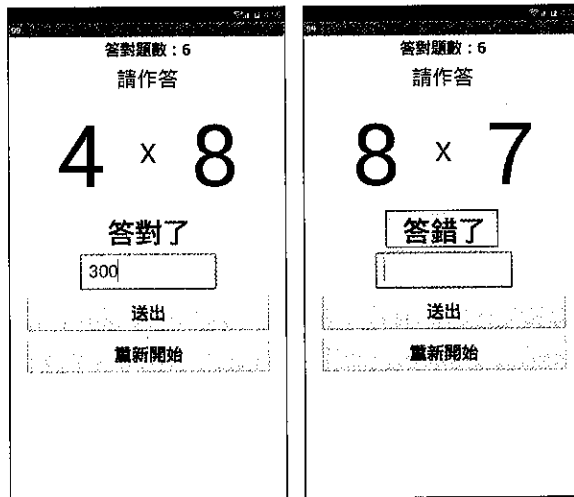


- (2) 輸入答案後送出，程式會判斷其相乘結果是否正確，如果正確會累積更新上方的答對題數，並修改【回答結果】的 Label 元件「result」，Text 顯示為【答對了】，清空 TextBox 元件內容。

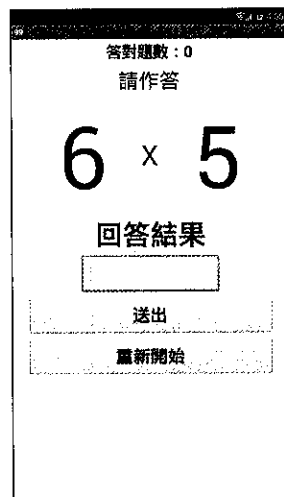




- (3) 如果答案錯誤，答對題數不更新，修改【回答結果】的 Label 元件，Text 顯示為【答錯了】。



- (4) 按下「重新開始」按鈕，上方的答對題數會歸零，下方的訊息會回復初始文字，並重新出題。



4. 自行測試程式是否達成下列結果：

- (1) 各物件與參考圖相同，屬性設定正確。
- (2) 輸入答案的 TextBox 元件設定只能輸入數字。
- (3) 程式起始即自動出題，被乘數與乘數限定 2~9，並隨機顯示。

- (4) 在輸入答案後點選「送出」按鈕，判斷答案正確，更新累計答對題數，result 的 Label 元件顯示【答對了】。
- (5) 如答案錯誤，則答對題數不更新，result 的 Label 元件顯示【答錯了】。
- (6) 按下「重新開始」按鈕，歸零上方的答對題數，回答結果的訊息回復初始文字，清空 TextBox 並重新出題。

5. 評分項目：

項	目	配 分	得 分
(1)	各物件與參考圖相同，屬性設定正確	3	
(2)	輸入答案的 TextBox 元件設定只能輸入數字	2	
(3)	程式起始即自動出題，被乘數與乘數限定 2~9，並隨機顯示	5	
(4)	在輸入答案後點選「送出」按鈕，判斷答案正確，更新累計答對題數，result 的 Label 元件顯示【答對了】	10	
(5)	如答案錯誤，則答對題數不更新，result 的 Label 元件顯示【答錯了】	5	
(6)	按下「重新開始」按鈕，歸零上方的答對題數，回答結果的訊息回復初始文字，清空 TextBox 並重新出題	5	
總	分	30	



203. 小畫家 易 中 難

1. 題目說明：

請開啓 **IND02.aia** 專案，設計「小畫家」程式，選擇不同的顏色按鈕，會改變畫筆顏色，點選畫面中央的畫布 Canvas 會出現圓圈，並可依手指移動軌跡畫出線條。請依下列題意完成作答，並製成封裝檔 **IND02.apk**，作答完成，請將 aia 專案檔及 apk 封裝檔皆下載並儲存於 C:\ANS.CSF\原專案的資料夾內。

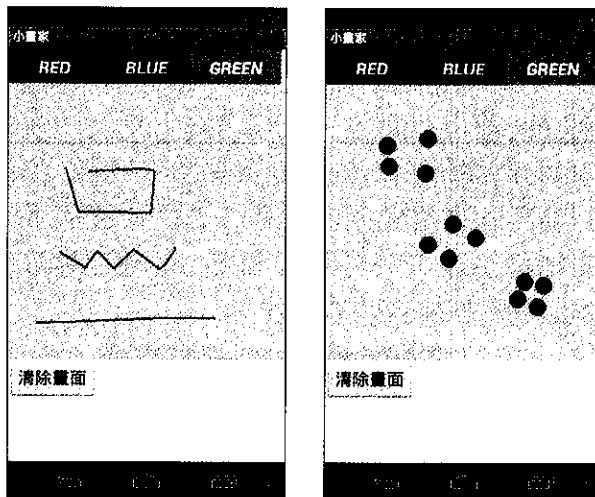


<參考圖>

2. 設計說明：

- (1) 頁面中三個 Button 元件，Text 由左至右分別為【RED】、【BLUE】與【GREEN】，底色分別設定為紅色、藍色與綠色。這三個按鈕需水平排列，Height 皆設定為 40 pixels，Width 為 105 pixels。
- (2) Button 元件下方為一個 Canvas 元件，底色為淺灰色 (Light Gray)，Height 設定為 320 pixels，Width 為 Fill parent。
- (3) 在畫布上點擊畫面時，會依選定的顏色畫出半徑為 10 大小的實心圓圈。
- (4) 在畫布上拖拉時，會根據手指軌跡及選定的顏色畫出線條。
- (5) 點選「清除畫面」按鈕，正確清除目前畫布上所有的畫圖圖案。

3. 執行結果參考畫面：



4. 自行測試程式是否達成下列結果：

- (1) 表單上各物件與參考圖相同，屬性設定正確。
- (2) 點擊畫面時，依選定的顏色畫出半徑為 10 大小的實心圓圈。
- (3) 在畫布上拖拉時，依所選擇的顏色及手指軌跡畫出線條。
- (4) 顏色可正確切換。
- (5) 點選「清除畫面」按鈕，正確清除畫面。

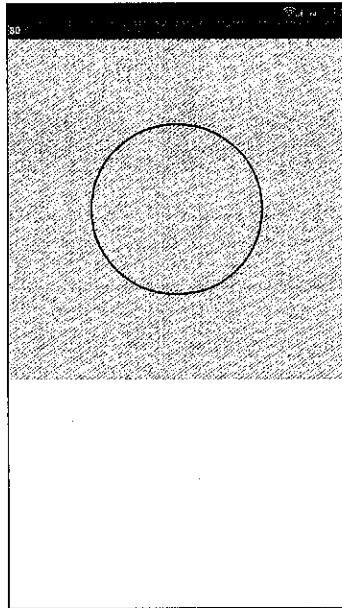
5. 評分項目：

項	目	配 分	得 分
(1)	表單上各物件與參考圖相同，屬性設定正確	5	
(2)	點擊畫面時，依選定的顏色畫出半徑為 10 大小的實心圓圈	8	
(3)	在畫布上拖拉時，依所選擇的顏色及手指軌跡畫出線條	8	
(4)	顏色可正確切換	6	
(5)	點選「清除畫面」按鈕，正確清除畫面	3	
總	分	30	

204. 動態畫圓 易 中 難

1. 題目說明：

請開啓 **IND02.aia** 專案，設計「動態畫圓」程式，當手指接觸畫面時，會以 Canvas 中心點與觸碰點距離為半徑畫出一個圓，移動手指時，圓也會跟著變大變小；當手指離開畫面時，圓就消失。請依下列題意完成作答，並製成封裝檔 **IND02.apk**，作答完成，請將 aia 專案檔及 apk 封裝檔皆下載並儲存於 C:\ANS.CSF\原專案的資料夾內。



<參考圖>

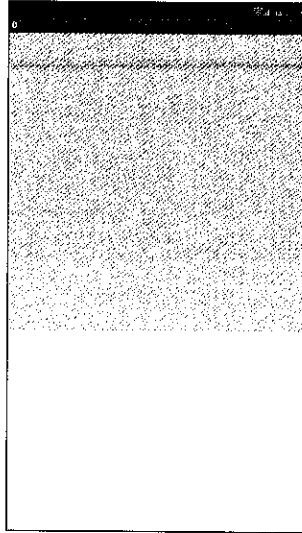
2. 設計說明：

- (1) 圓半徑方程式為 $\sqrt{(x - x1)^2 + (y - y1)^2}$ 。其中 $x、y$ 為觸碰點座標， $x1、y1$ 為 Canvas 中心座標，也就是(160, 160)。
- (2) 手指接觸畫面，將根據觸碰點與 Canvas 中心點距離畫圓，並將圓半徑顯示於 Screen 的標題 (Title)。
- (3) 移動手指時，圓也會跟著變大變小，同樣將圓半徑顯示於 Screen 的標題 (Title)。
- (4) 手指離開畫面，圓消失，Screen 的標題也歸零。

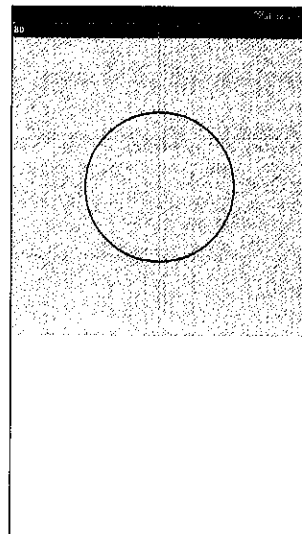
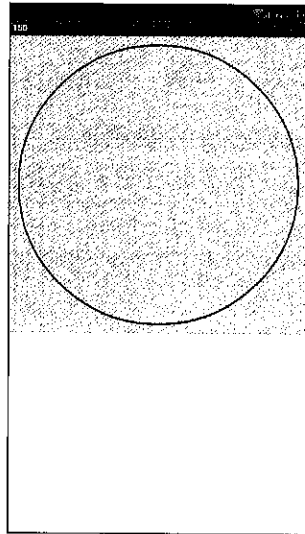


3. 執行結果參考畫面：

(1) 程式初始畫面。

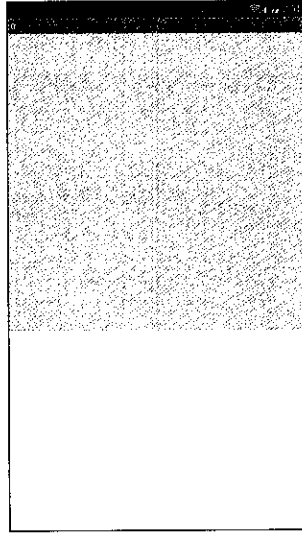


(2) 手指接觸畫面，將根據觸碰點與 Canvas 中心點距離畫圓，並將圓半徑顯示於 Screen 的標題 (Title)，如下圖左 (半徑為 150) 與下圖右 (半徑為 80)。





(3) 手指離開畫面，圓消失，Screen 的標題也歸零。



4. 自行測試程式是否達成下列結果：

- (1) 表單上各物件與參考圖相同，屬性設定正確。
- (2) 手指接觸畫面，根據觸碰點與 Canvas 中心點距離畫圓。
- (3) 移動手指時，圓也會跟著變大變小。
- (4) 畫圓時顯示圓半徑於 Screen 的標題 (Title)。
- (5) 手指離開畫面，圓消失，Screen 的標題也歸零。

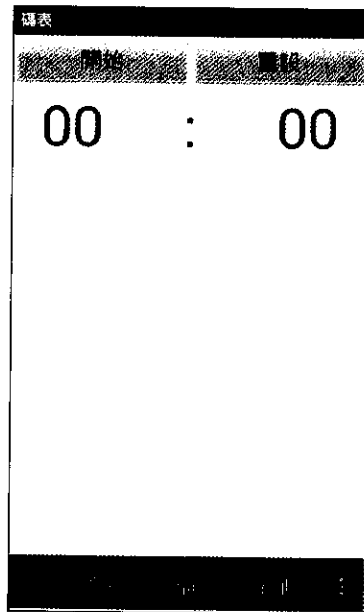
5. 評分項目：

項	目	配分	得分
(1)	表單上各物件與參考圖相同，屬性設定正確	4	
(2)	手指接觸畫面，根據觸碰點與 Canvas 中心點距離畫圓	8	
(3)	移動手指時，圓也會跟著變大變小	8	
(4)	畫圓時顯示圓半徑於 Screen 的標題 (Title)	6	
(5)	手指離開畫面，圓消失，Screen 的標題也歸零	4	
總分		30	

205. 碼表..... 易 中 難

1. 題目說明：

請開啓 **IND02.aia** 專案，設計「碼表」程式，可進行碼表的計時，並重新歸零後再次計時。請依下列題意完成作答，並製成封裝檔 **IND02.apk**，作答完成，請將 aia 專案檔及 apk 封裝檔皆下載並儲存於 C:\ANS.CSF\原專案的資料夾內。



<參考圖>

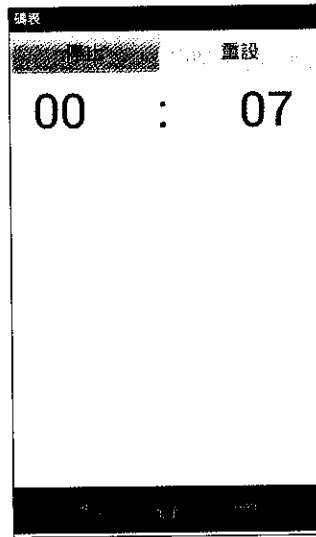
2. 設計說明：

- (1) 設定 Screen1 的 Title 為【碼表】。
- (2) 按下「開始」按鈕，開始碼表計時，並在按下按鈕後，Text 改變為【停止】，此時「重設」按鈕不可作用。
- (3) 按下「停止」按鈕，Text 改變為【開始】，設定「重設」按鈕為可作用。
- (4) 按下「重設」按鈕，設定時間歸零。

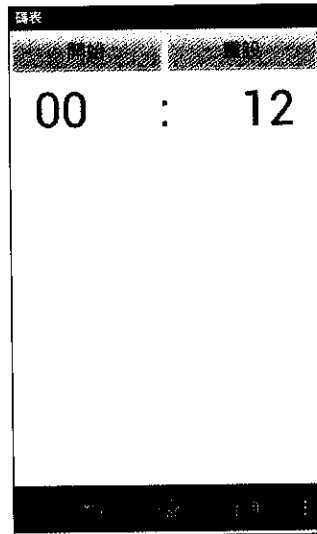


3. 執行結果參考畫面：

- (1) 按下「開始」按鈕，開始碼表計時，並在按下按鈕後，Text 改變為【停止】，此時「重設」按鈕不可作用。



- (2) 按下「停止」按鈕，Text 改變為【開始】，設定「重設」按鈕為可作用。



4. 自行測試程式是否達成下列結果：

- (1) 按下「開始」按鈕，進行計時並變更 Text 為【停止】，計時的同時無法按下「重設」按鈕。
- (2) 按下「停止」按鈕，Text 改變為【開始】，設定「重設」按鈕為可作用。
- (3) 計時期間，時間正確顯示。
- (4) 按下「重設」按鈕，將時間歸零。

5. 評分項目：

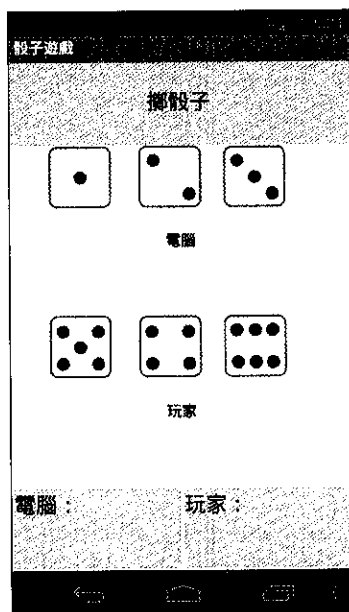
項 目	配 分	得 分
(1) 按下「開始」按鈕，進行計時並變更 Text 為【停止】，計時的同時無法按下「重設」按鈕	5	
(2) 按下「停止」按鈕，Text 改變為【開始】，設定「重設」按鈕為可作用	5	
(3) 計時期間，時間正確顯示	15	
(4) 按下「重設」按鈕，將時間歸零	5	
總 分	30	



206. 骰子遊戲 易 中 難

1. 題目說明：

請開啓 **IND02.aia** 專案，設計「骰子遊戲」程式，電腦與玩家各自持有三個骰子，按下「擲骰子」按鈕，畫面上的六個骰子會隨機顯示點數，並計算玩家與電腦所得到的點數。請依下列題意完成作答，並製成封裝檔 **IND02.apk**，作答完成，請將 **aia** 專案檔及 **apk** 封裝檔皆下載並儲存於 **C:\ANS.CSF\原專案**的資料夾內。

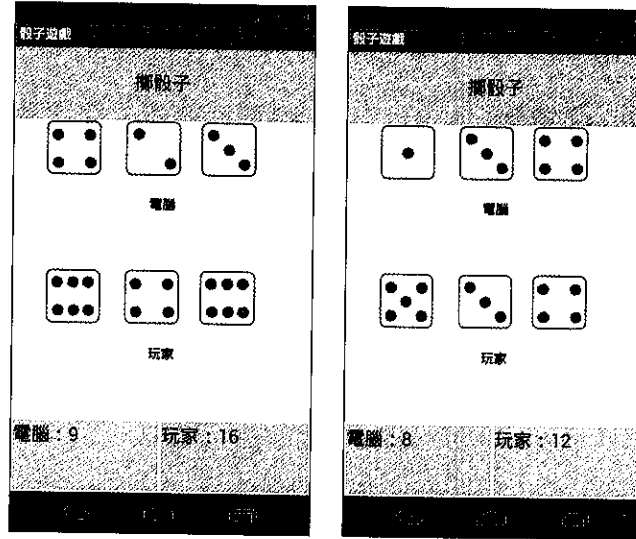


<參考圖>

2. 設計說明：

- (1) 專案中已提供六張圖片，分別代表骰子一~六點。
- (2) Screen 頁面中有兩個 Canvas 元件，在畫布中以電腦和玩家做區分，分別加入三個 ImageSprite 元件表示骰子的圖案，一共六個 ImageSprite 元件，顯示如參考圖所示。
- (3) 每一個骰子都能亂數表示一點到六點的圖案。
- (4) 按下「擲骰子」按鈕，畫面上六個骰子會隨機表示數字，並於下方以 Label 元件分別顯示所得到的骰子點數。

3. 執行結果參考畫面：



4. 自行測試程式是否達成下列結果：

- (1) 按下「擲骰子」按鈕，畫面上六個骰子會隨機表示一點到六點的圖案。
- (2) 必須能正確統計電腦與玩家得到的骰子點數。

5. 評分項目：

項	目	配 分	得 分
(1)	按下「擲骰子」按鈕，畫面上六個骰子會隨機表示一點到六點的圖案	12	
(2)	正確統計電腦得到的骰子總點數	9	
(3)	正確統計玩家得到的骰子總點數	9	
總	分	30	

207. 相片藝廊 易 中 難

1. 題目說明：

請開啓 **IND02.aia** 專案，設計「相片藝廊」程式，可切換功能按鈕或滑動螢幕，進行相片的展示。請依下列題意完成作答，並製成封裝檔 **IND02.apk**，作答完成，請將 aia 專案檔及 apk 封裝檔皆下載並儲存於 C:\ANS.CSF\原專案的資料夾內。



<參考圖>

2. 設計說明：

(1) 專案中已提供本題所需之素材及圖片清單，圖片順序為：

1—bee 蜜蜂	2—beetle 金龜子	3—butterfly 蝴蝶
4—Cerambycidae 天牛	5—dragonfly 蜻蜓	6—grasshopper 蚱蜢
7—ladybug 瓢蟲	8—mantis 螳螂	9—stagbeetle 鍬形蟲

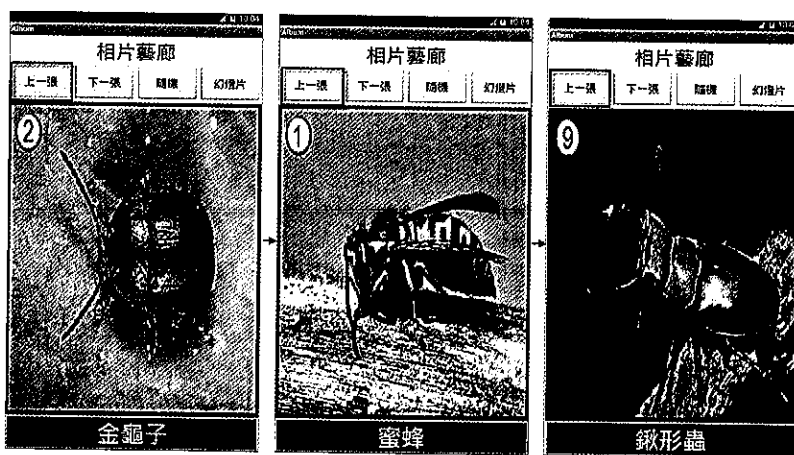
(2) 程式起始，Canvas 元件即讀取編號 1 的圖片 (**bee.jpg**)，最下方 Label 元件顯示昆蟲對應的名稱。

(3) 點選「上一張」按鈕，或由左往右滑動螢幕畫面，切換上一個編號昆蟲圖片及名稱。

- (4) 點選「下一張」按鈕，或由右往左滑動螢幕畫面，切換下一個編號昆蟲圖片及名稱。
- (5) 點選「隨機」按鈕，由 1~9 號隨機顯示昆蟲圖片及名稱。
- (6) 點選「幻燈片」按鈕，以目前正在顯示的圖片為起點，每隔 1 秒鐘自動切換下一張昆蟲圖片及名稱，並能依順序循環播放。

3. 執行結果參考畫面：

- (1) 點選「上一張」按鈕，或由左往右滑動螢幕畫面，圖片編號會遞減 (3→2→1→9)，當編號到 1 時，再上一張就變成 9 號圖片，圖片與名稱對應正確顯示。



- (2) 點選「下一張」按鈕，或由右往左滑動螢幕畫面，圖片編號會遞增 (8→9→1→2)，當編號到 9 時，再下一張就變成 1 號圖片，圖片與名稱對應正確顯示。





4. 自行測試程式是否達成下列結果：

- (1) 程式起始，Canvas 元件即讀取編號 1 的圖片 (**bee.jpg**)，最下方 Label 元件顯示昆蟲對應的名稱。
- (2) 點選「上一張」按鈕時，或由左往右滑動螢幕畫面，圖片編號遞減，圖片與名稱對應正確顯示。
- (3) 點選「下一張」按鈕，或由右往左滑動螢幕畫面，圖片編號遞增，圖片與名稱對應正確顯示。
- (4) 點選「隨機」按鈕，隨機顯示 1~9 號的圖，圖片與名稱對應正確顯示。
- (5) 點選「幻燈片」按鈕，以目前正在顯示的圖片為起點，每隔 1 秒鐘自動切換下一張昆蟲圖片及名稱，並能依順序循環播放。

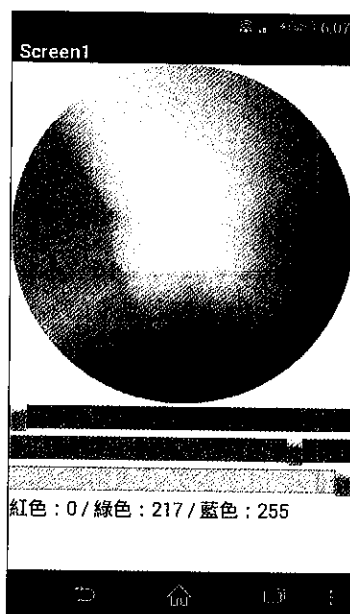
5. 評分項目：

項	目	配 分	得 分
(1)	程式起始，Canvas 元件即讀取編號 1 的圖片 (bee.jpg)，最下方 Label 元件顯示昆蟲對應的名稱	2	
(2)	點選「上一張」按鈕，圖片編號遞減，圖片與名稱對應正確顯示	4	
(3)	由左往右滑動螢幕畫面，圖片編號遞減，圖片與名稱對應正確顯示	4	
(4)	點選「下一張」按鈕，圖片編號遞增，圖片與名稱對應正確顯示	4	
(5)	由右往左滑動螢幕畫面，圖片編號遞增，圖片與名稱對應正確顯示	4	
(6)	點選「隨機」按鈕，隨機顯示 1~9 號的圖片，圖片與名稱對應正確顯示	6	
(7)	點選「幻燈片」按鈕，以目前正在顯示的圖片為起點，每隔 1 秒鐘自動切換下一張昆蟲圖片及名稱，並能依順序循環播放	6	
總	分	30	

208. 取得像素顏色..... 易 中 難

1. 題目說明：

請開啓 **IND02.aia** 專案，設計「取得像素顏色」程式。點擊 Canvas 元件畫面上的任一處都會將該處的颜色解析結果顯示於畫面下方的 Label 元件。畫面上的三個 Slider 元件也會根據紅色、綠色與藍色的強度來移動到指定位置。請依下列題意完成作答，並製成封裝檔 **IND02.apk**，作答完成，請將 aia 專案檔及 apk 封裝檔皆下載並儲存於 C:\ANS.CSF\原專案的資料夾內。



<參考圖>

2. 設計說明：

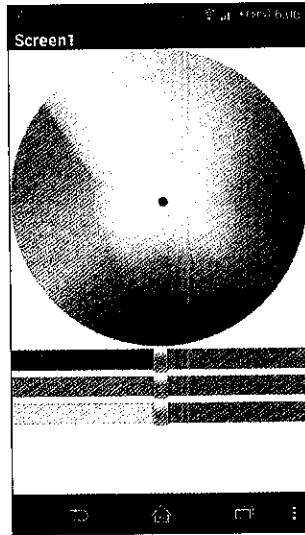
- (1) 在點擊 Canvas 元件任一處時，會在點擊處顯示像素顏色數值（Canvas 元件的 `getPixelColor` 指令）。另外還會將該處的颜色資料解析出紅色、綠色與藍色等三原色顯示於 Label 元件。
- (2) 頁面中三個 Slider 元件，可用來顯示颜色強度，依序分別將其 `ColorLeft` 欄位改為紅色、綠色與藍色。最小值（`MinValue`）為 0，最大值（`MaxValue`）為 255，以對應颜色強度的範圍。Slider 的指針位置（`ThumbPosition`）會根據點擊處的颜色解析結果移動到對應的位置。



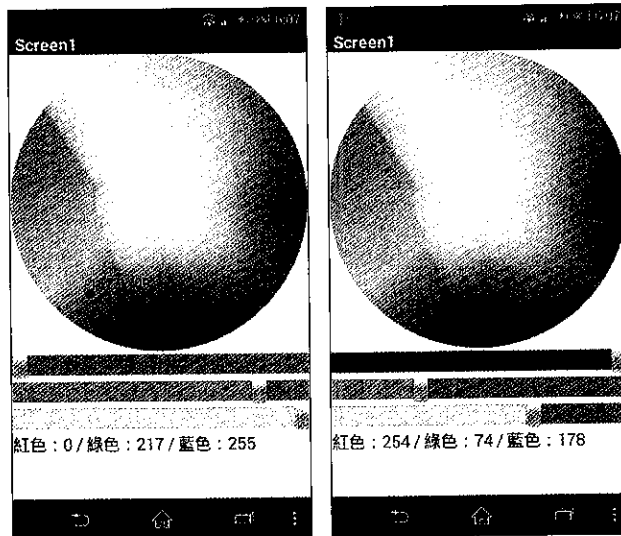
- (3) 最下方 Label 元件，可用來顯示顏色解析結果，Text 欄位須清空，程式執行時會把點擊處的三原色數值顯示出來，顯示格式為【紅色：R / 綠色：G / 藍色：B】，RGB 分別代表點擊處的顏色解析結果。

3. 執行結果參考畫面：

- (1) 程式初始畫面，黑點位於 Canvas 元件中央，滑桿也位於中央。



- (2) 點擊 Canvas 元件後，黑球移動到點擊位置，並於點擊位置顯示數值。滑桿顯示與顏色解析皆正確。



4. 自行測試程式是否達成下列結果：

- (1) 點擊 Canvas 元件後，黑球可移動到點擊位置，並於點擊位置顯示數值。
- (2) 滑桿在點擊 Canvas 元件後，可根據該點顏色強度移動到指定位置。
- (3) 在點擊 Canvas 元件後，可進行顏色解析，並以 Label 元件顯示點擊位置紅色、綠色與藍色等三原色強度。

5. 評分項目：

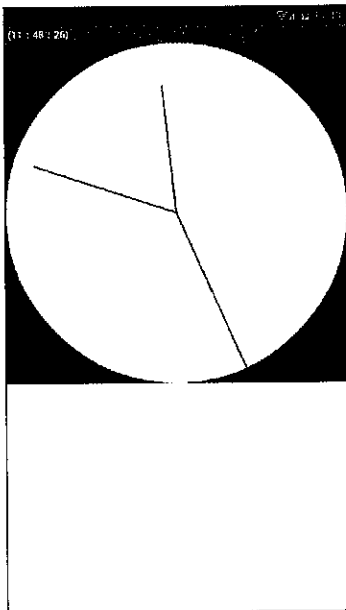
項 目	配 分	得 分
(1) 點擊 Canvas 元件後，黑球可移動到點擊位置，並於點擊位置顯示數值	10	
(2) 滑桿在點擊 Canvas 元件後，可根據該點顏色強度移動到指定位置	10	
(3) 在點擊 Canvas 元件後，可進行顏色解析，並以 Label 元件顯示點擊位置紅色、綠色與藍色等三原色強度	10	
總 分	30	



209. 指針式時鐘 易 中 難

1. 題目說明：

請開啓 **IND02.aia** 專案，設計「指針式時鐘」程式。Canvas 元件會自動根據系統時間資訊，繪製類比式時鐘，需有秒針、分針與時針，並將時間顯示於 Screen 的 Title 狀態列。同時 Canvas 元件的背景顏色會每分鐘隨機更換。請依下列題意完成作答，並製成封裝檔 **IND02.apk**，作答完成，請將 aia 專案檔及 apk 封裝檔皆下載並儲存於 C:\ANS.CSF\原專案的資料夾內。



<參考圖>

2. 設計說明：

- (1) Screen1 頁面中，有一個 Canvas 元件，Canvas 畫布元件的背景顏色會每分鐘隨機更換，顏色可自訂。並會將系統時間以 (HH : MM : SS) 的格式顯示於 Screen 的 Title 狀態列。
- (2) Canvas 元件需要根據 Clock 元件取得的時間資訊來繪製類比式時鐘，意即有時針、分針與秒針。針對分針與秒針，每次轉動均以 6 度為單位 (360 度 / 60 分或秒)，針對時針每次轉動則以 30 度為單位 (360 度 / 12 小時)，並以錶盤中心為轉動圓心。

(3) 計算指針端點 (x,y) 的公式如下：

a. $x = \text{圓心 } x \text{ 坐標} + \sin(\text{指針方向角}) \times \text{指針長度}$

b. $y = \text{圓心 } y \text{ 坐標} + \cos(\text{指針方向角}) \times \text{指針長度}$

(4) 計算秒針、分針與時針的公式如下：

a. 秒針公式：

$$x = 160 + \sin(\text{秒數} \times 6) \times 160$$

$$y = 160 - \cos(\text{秒數} \times 6) \times 160$$

b. 分針公式：

$$x = 160 + \sin(\text{分數} \times 6) \times 140$$

$$y = 160 - \cos(\text{分數} \times 6) \times 140$$

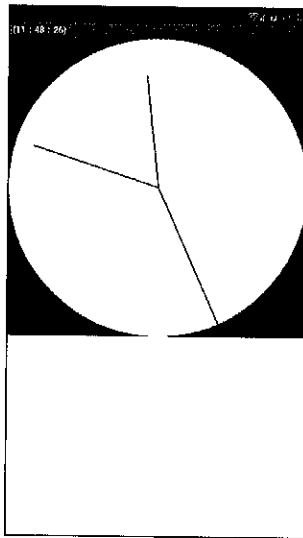
c. 時針公式：

$$x = 160 + \sin((\text{小時}/12 \text{ 的餘數}) \times 30 + (\text{分數}/2 \text{ 的商})) \times 120$$

$$y = 160 - \cos((\text{小時}/12 \text{ 的餘數}) \times 30 + (\text{分數}/2 \text{ 的商})) \times 120$$

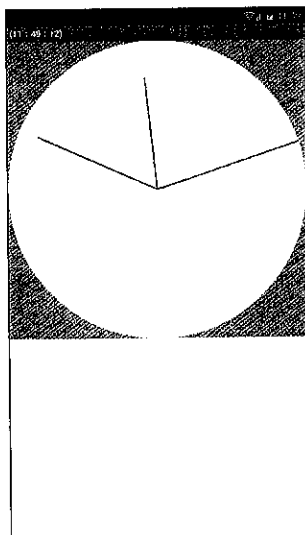
3. 執行結果參考畫面：

(1) 程式初始畫面，秒針、分針與時針會正確根據時間轉動。





(2) Canvas 背景顏色會每分鐘隨機更換。



4. 自行測試程式是否達成下列結果：

- (1) Canvas 可根據系統時間繪製秒針、分針與時針。
- (2) 系統時間會顯示於 Screen 的 Title 狀態列。
- (3) Canvas 背景顏色會每分鐘隨機更換。

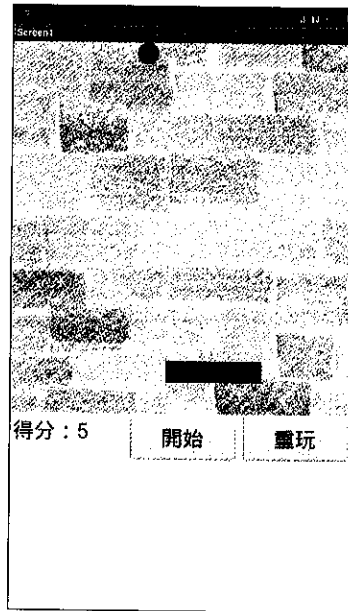
5. 評分項目：

項	目	配 分	得 分
(1)	Canvas 可根據系統時間繪製秒針、分針與時針	10	
(2)	系統時間會顯示於 Screen 的 Title 狀態列	10	
(3)	Canvas 背景顏色會每分鐘隨機更換	10	
總	分	30	

210. 打磚塊 易 中 難

1. 題目說明：

請開啓 **IND02.aia** 專案，設計「打磚塊」程式，點選「開始」按鈕，球會開始移動，請用手指左右拖拉黑色板子來接球，每次接到球都得 1 分；如果黑色板子漏接球，就結束遊戲並顯示總得分。請依下列題意完成作答，並製成封裝檔 **IND02.apk**，作答完成，請將 aia 專案檔及 apk 封裝檔皆下載並儲存於 C:\ANS.CSF\原專案的資料夾內。



<參考圖>

2. 設計說明：

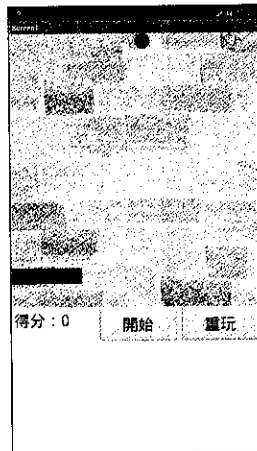
- (1) Ball 元件藍色球起始點請設定位於 Canvas 元件畫面最上方正中央，黑色板子維持專案內預設高度，請設定在畫面左側。
- (2) 點選「開始」按鈕，球會開始四處移動，球的速度 (Speed) 為 30、間隔 (Interval) 為 5、指向 (Heading) 需隨機指定為 225 到 315 之間的整數。
- (3) 黑色板子只可左右拉動，當球碰到板子時，表示接到球得 1 分，會發出音效 **Noink.mp3**，球的指向會改變為：「360 - 目前的指向」，並繼續彈跳。



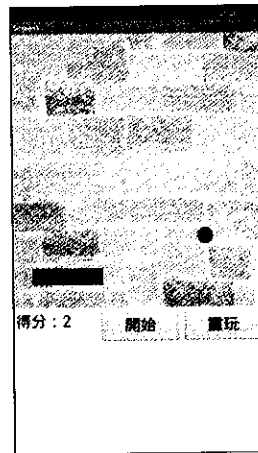
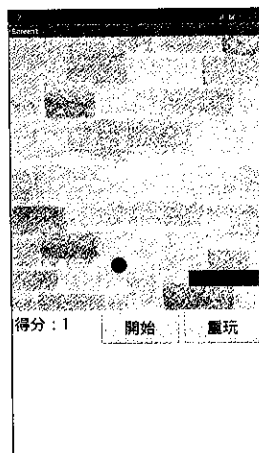
- (4) 如果球碰到 Canvas 元件畫面最下方（即黑色板子漏接球），代表遊戲結束，球停止移動，並發出音效 **Buzzer.mp3**，更新顯示總得分【遊戲結束，您的分數為 n 分】，n 代入總得分。
- (5) 點選「重玩」按鈕會使分數歸零、球回到 Canvas 元件畫面最上方正中央、黑色板子至原預設位置。再次點選「開始」按鈕就能開始遊戲。

3. 執行結果參考畫面：

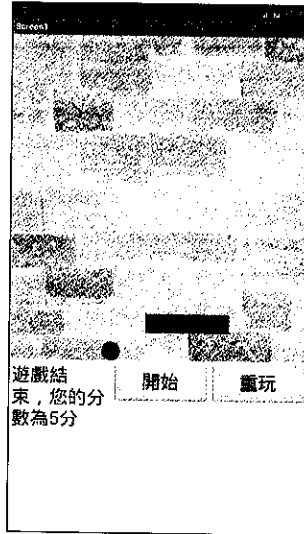
- (1) 程式初始畫面，Ball 元件藍色球起始點預設位於 Canvas 元件畫面最上方正中央，黑色板子則在畫面左側。



- (2) 點選「開始」按鈕，球會開始四處移動，黑色板子只可左右拉動，當球碰到板子時，表示接到球得 1 分，會發出音效 **Noink.mp3**。



- (3) 如果球碰到 Canvas 元件畫面最下方（即黑色板子漏接球），代表遊戲結束，球停止移動，並發出音效 **Buzzer.mp3**，更新顯示總得分【遊戲結束，您的分數為 n 分】，n 代入總得分。



4. 自行測試程式是否達成下列結果：

- (1) Ball 元件藍色球起始點預設位於 Canvas 元件畫面最上方正中央，黑色板子則在畫面左側。
- (2) 點選「開始」按鈕，當球發生碰撞會改變方向，球的速度 (Speed)、間隔 (Interval)、指向 (Heading) 符合要求。
- (3) 黑色板子只可左右拉動，當球碰到板子時，表示接到球得 1 分，會發出音效 **Noink.mp3**，球的指向會改變為：「360 - 目前的指向」，並繼續彈跳。
- (4) 如果球碰到 Canvas 元件畫面最下方（即黑色板子漏接球），代表遊戲結束，球停止移動，並發出音效 **Buzzer.mp3**，更新顯示總得分【遊戲結束，您的分數為 n 分】，n 代入總得分。
- (5) 點選「重玩」按鈕會使分數歸零、球及黑色板子皆回復至原預設位置。



5. 評分項目：

項 目	配 分	得 分
(1) Ball 元件藍色球起始點預設位於 Canvas 元件畫面最上方正中央，黑色板子則在畫面左側	6	
(2) 點選「開始」按鈕，當球發生碰撞會改變方向，球的速度 (Speed)、間隔 (Interval)、指向 (Heading) 符合要求	6	
(3) 黑色板子只可左右拉動，當球碰到板子時，表示接到球得 1 分，會發出音效 Noink.mp3 ，球的指向會改變為：「360 - 目前的指向」，並繼續彈跳	6	
(4) 球碰到 Canvas 元件畫面最下方（即黑色板子漏接球），代表遊戲結束，球停止移動，並發出音效 Buzzer.mp3 ，更新顯示總得分【遊戲結束，您的分數為 n 分】	6	
(5) 點選「重玩」按鈕會使分數歸零、球及黑色板子皆回復至原預設位置	6	
總 分	30	