程式設計 第27組 期末專案書面報告

B09703083 財金二 何柏翰
B09703120 財金二 黃照軒
B09303029 經濟二 陳怡安
B08303044 經濟三 鄭念觀
R07227206 心理碩 王建又

一、簡介

我們這組的期末專案主題是運用 <graphics.h> 這個C++ 的 headerfile 來實作我們的遊戲。我們的期末專案包含Reaction time, Sequence memory, Aim trainer, Number memory, Typing等五個心理學小遊戲、包含五個小遊戲的主畫面與音樂系統，讓玩家測試自己的反應力、記憶力、打字速度等。

Reaction time:
這個遊戲可以量測使用者的視動速度 (visuomotor) 表現，相對Aim trainer更為簡單。

Sequence memory:
這個遊戲考驗玩家的視空間工作記憶 (visuospatial working memory)。

Aim trainer:
這個遊戲可以量測使用者的視動速度 (visuomotor) 表現。

Number memory:
這個遊戲可以量測玩家的工作記憶 (working memory)。

Typing:
這個遊戲可以量測使用者的視動速度 (visuomotor) 表現。而typing需要判斷與協調表現。

二、演算法與程式碼

（一）Main Page（何柏翰）

mainPage主要是由<graphics.h> library構建而成，功能是顯示心理學測驗中所有小遊戲的列表，在點擊mainPage上的按鍵後會自動跳入選取的小遊戲中，亦為跳入不同的函式中，分別為reactionTime(), sequenceMemory(), aimTrainer(), numberMemory()與typing()。其中較為技術性的部分是滑鼠點擊和版面配置的部分。

1. 版面配置:

字和格子以電腦寬度的中點作為中心點，分別向外延伸，以避免版面因每台電腦螢幕大小的不同、或是字型大小的不同而發生錯位的情況。

2. 滑鼠點擊:

主要是藉由使用者按滑鼠時的滑鼠位置來判斷是否有按到按鍵，這部分是由ismouseclick()偵測使用者是否按價滑鼠左鍵，getmouseclick()偵測使用者點擊的位置，clearmouseclick()刪除滑鼠左鍵的狀態。

3. 開啟windows視窗:

使用windows.h library中GetSystemMetrics函數，讀取螢幕長寬並使用intwindow來開啟全螢幕視窗，全螢幕的長寬是取自GetSystemMatrix。

（二）Reaction time（鄭念觀）

此部分將測試使用者的反應時（reaction time, 簡稱 RT），是指給予使用者刺激之後到反應開始所需要的時間，即刺激-反應的時間間隔。本遊戲一開始是紅色的等待畫面，將維持一段隨機的時間當畫面由紅色轉為綠色時，使用者必須盡可能的以最快的速度按下滑鼠左鍵，畫面顏色轉換到使用者按下左鍵此段時間即為反應時。

接著將介紹本程式的運作模式。本程式主要透過 <graphics.h> 函式庫架構而成。本部分之顯示方式及邏輯皆與其他遊戲相同，不再贅述。

一開始，先設定並顯示起始頁面，並顯示 “click fast as you can”等提示語。接著使用 GetAsyncKeyState(VK_LBUTTON) 偵測使用者是否按下左鍵。當使用者按下左鍵時，遊戲即馬上開始，將畫面轉為紅色，接著呼叫 randomNum 函式，本函式將回傳一個不大於 maxTime 的隨機數字。而遊戲將使用 delay() 顯示紅色畫面不等的時間。該時間結束後，隨即將畫面顯示改為綠色，並開始以<chrono>中的high_resolution_clock::now() 紀錄當前的時間，並以while 迴圈及GetAsyncKeyState(VK_LBUTTON) 等待使用者按下左鍵。當使用者按下左鍵時，再使用high_resolution_clock::now() 取得當前時間，並與開始時間相減，得到使用者的反應時間；按下的同時會使用 Beep() 函數放出點擊音效。最後將使用者的反應顯示畫面中。本次遊戲的反應時間亦會存入 sum 中，最後計算平均反應時間。

當使用者再度按下滑鼠左鍵時，將繼續重複一次遊戲。本遊戲將如此重複 playingTime 次。最後結束時，畫面上將顯示使用者的平均反應時間，最後會回到程式主畫面。

（三）Sequence memory（何柏翰）

Sequence Memory這個小遊戲是由C++ graphics.h的library製作而成的，在遊戲中，我一共建立了兩個函式，分別是SequenceMemorySetting() 與 SequenceMemory。以下進行介紹。

SequenceMemorySetting() –
設定起始的遊玩畫面，即為內含6*6個正方形格子的大正方形，這些圖形由<graphics.h>中的幾何圖形所構建而成。

SequenceMemory –
遊戲的主程式，其中又分為三個較重要的部分

1. 滑鼠左鍵點擊:

使用graphics.h 的ismouseclick()函式來判斷使用者是否點擊滑鼠左鍵，並用getmouseclick()並用getmouseclick()函式來讀取滑鼠目前所在位置，以判斷使用者是否有點擊到正確的位置，最後用clearmouseclick()來清除游標點擊的紀錄。

2. 正方形格子閃燈:

使用rand()讓格子以隨機的方式亮藍燈，而亮藍燈則是藉由graphics.h中的floodfill()函式來使格子顏色轉藍。另外，當使用者點擊正確的格子時，同樣也是使用floodfill()函式使格子顏色轉白，反之，錯誤時轉紅。

3. 音效:

在遊戲過程中，在使用者點擊格子時，會有音效產生，主要是由beep函數來製成，每次點擊會增加50HZ音頻。

（四）Aim trainer（王建又、鄭念觀）

首先，透過sprintf函式、setcolor函式、settextstyle函式，及outtextxy函式，設定並顯示起始頁面的遊戲提示語。其後，以帶有WM_LBUTTONDOWN引數（左鍵是否按下）的ismouseclick函式置入while迴圈，以反覆偵測使用者是否有點擊滑鼠左鍵；如有點擊滑鼠左鍵，即跳出迴圈，並以clearmouseclick函式改變ismouseclick函式的回傳值、以cleardevice函式清除圖形模態上的字句，並進入遊戲迴圈（game loop）中。

在此之前，先設計三個全域函式（global function）：void target(int x, int y)、int randomPos(int screenRang)以及void bee(int location)。Target函式以setcolor函式、circle函式、line函式畫出同心圓標（參數x、y用以設定同心圓圓心）randomPos透過rand函式，產生隨機數字，並由參數screenRang來限制回傳值的範圍，以產生適用於target函式的引數；而在bee函式中，透過呼叫 Beep函式來發出聲音（〈小蜜蜂〉）。

遊戲迴圈主要由一個內含while迴圈的for迴圈所架構而成。在for迴圈開始前，先以srand函式設定亂數種子。在for迴圈開始之後，先以randomPos函式分別隨機產生變數x與y的值，並以此二變數作為引數，呼叫target函式以便在顯示器上畫出同心圓標的。之後，透過high_resolution_clock::now之成員函式回傳當前的Unix時間。接著進入while迴圈，分別以ismouseclick函式和getmouseclick函式分別確認使用者是否有使用左鍵點擊顯示器，以及擊點位置是否有在同心圓標的內；若有，則跳脫while迴圈，並再回傳當前的Unix時間，並與之前所得時間相減，得出反應時間（reaction time）。另外，離開while迴圈後，呼叫bee函式，發出聲音以作為遊戲的另一反饋。
跳脫for迴圈後，將累積的反應時間與迴圈次數相除，即可取得平均反應時間（average reaction time）。

最後，一樣以sprintf函式、setcolor函式、settextstyle函式，以及outtextxy函式，設定並顯示終止頁面的遊戲提示語，並以ismouseclick函式偵測點擊；若點擊則呼叫closegraphic函式，以結束圖形模態。

（五）Number memory（陳怡安）

數字廣度的演算法主要分為兩部分：一個是顯示數字的頁面，另一個是使用者輸入的頁面。

l. 顯示數字的頁面

我們讓數字長度從 1 開始（設定level = 1），若使用者回答正確就讓數字長度增加 1；設定一個布林變數，當使用者回答正確時持續執行此程式。

適用 while loop，讓當使用者沒有回答錯誤時會持續偵測使用者是否點擊螢幕以開始遊戲，當使用者點擊螢幕，第 level 回合的遊戲開始。

首先，先建立一個長度為 level 的整數陣列 rn，將用來儲存接下來產稱的隨機數字，以利後續與使用者輸入的數字比較，在每次產生一個隨機數字的時候顯示於螢幕上，結束顯示第 level 個數字後，到轉移到輸入頁面，也就是呼叫 answeringPage() 函數。
answeringPage() 函數結束後，根據回傳之 correct，若仍為 true 前往下一 level；若為 false，告知使用者其位在的 level ，並結束此遊戲，回到主頁。

```
顯示題目數字頁面()
        頁面設計
        int level = 1; // 設定 level （也就刺數字長度）從1開始跑
        bool correct = true;
        while(correct)
                if(使用者點擊螢幕)
                        int* rn = new int[level];
                        for(1-level)
                                rn = rand()%10; // 依據 level 大小決定產生幾個隨機數字（0-9）

                        answeringPage(); // 跳至回答頁面，讓使用者輸入答案
                        if( correct == false) // 也就是使用者回答錯誤
                                break;

        顯示使用者的 level
```

II. 讓使用者輸入之頁面

根據傳入之 level ，利用 for loop 讓使用者輸入 level 次，也就是 level 個數字答案，當使用者輸入第 i 個某數字為 n，檢查此數字是否等於整數陣列儲存之第 i 個數字，若否，設定 correct 為 false。輸入數字時，使用者輸入的數字會顯示於畫面。若到輸入的最後一個（第 level 個）數字，correct 不曾改變仍為 true，則提示使用者點擊螢幕進行下一 level 遊戲。

```
answeringPage()
        頁面設計
        for(i = 1-level)
                while(輸入次數小於等於題目數字長度)
                        if(使用者點擊某數字)
                                if(輸入的這個數字不等於題目給定的數字)
                                        correct = 0; // 設定 correct = false

        if(correct == 1) // 若迴圈皆跑完，答案沒有錯誤時
                顯示提醒使用者點擊螢幕玩下一等級
```

（六）Typing（黃照軒、鄭念觀、陳怡安、何柏翰）

一開始我們覺得這應該是一個簡單的遊戲，不過後來我們遇到一些問題。首先，我們最一開始是用 cin 讀入玩家輸入的字母，但是我們發現 cin 無法適用在視窗模式；換句話說，cin 意味著console in，但是當螢幕出現視窗時，玩家並不是打在console上，所以這個方法行不通。

於是，我們試著改變方法，我們運用 <windows.h> 中的 getAsyncKeyState() 函數。這個函數輸入一個狀態，如’A’或是一些鍵盤代碼。如果鍵盤狀態是輸入狀態，則回傳true（非零的整數），若鍵盤狀態非輸入狀態，則回傳false。但是，這個函數在這個遊戲中並不好用，因為這代表我們需要寫26個，甚至更多個類似的結構來接收玩家輸入的東西。

此外，我們外圈是while迴圈，電腦會不斷偵測鍵盤狀態，但是在玩家按下另一個鍵前，getAsyncKeyState() 函數會記下玩家先前的鍵盤狀態，因此不斷在視窗上印出同樣的字母。

於是，我們換了一個函數，我們使用 <conio.h> 中的 getch() 函數。這個函數會接收玩家按下的按鈕，並回傳按下按鈕的ASCII code。這個辦法解決了的while迴圈與重複印出相同字母的問題。最終，我們可以讓玩家看到他打的字，也可以讓玩家刪除先前打的字，是一大突破。

最後，我們再將玩家打的句子與原本的句子比較，若相同，則顯示花費秒數，若不相同，則顯示” You made some mistakes”。

三、心得

何柏翰：

這次的期末專案我學到了兩件很重要的事情，第一件事情是團體合作，在一開始時，大家都不知道要做什麼遊戲比較好，還有要用什麼函式庫來建立遊戲比較好。雖然後來我們選擇了用graphics.h做遊戲，不過因為大家都沒有用c++做小遊戲的經驗，因此大家都是從零開始學起，一步一步互相扶持著彼此，最後終於完成了這個遊戲。儘管在現在看起來是多麼陽春的東西，不過卻是我們每一天熬夜討論、熬夜打程式的成果，在一開始，我從沒有想過我們真的能做出一個遊戲! 所以從這次的期末專案中，我了解到1個人雖然渺小，但團結起來可以完成一個艱鉅的任務，因此我要感謝每一位組員的付出。

第二件事情是自學的態度，從小到大，我常常都是完成老師或教授所指定的任務，這些任務大部分都可以用師長教導我們的東西來達成，不需要自己上網查資料來完成任務，但是這次的期末專案，不僅僅是需要用到上課所學，也要用到課本以外的知識，因此我需要自己查論壇、看網路上教學影片，和組員們一起從零開始、互相討論問題來學習這些不是上課內容的範圍。所以我也從這堂課中學到自學的重要性，相信這個技能不管是在大學抑或是出社會後都是不可或缺的能力。

黃照軒：

我們用的是很古老的<graphics.h>，所以介面看起來相對許多新的library，套件更為簡陋。不過，這也讓我覺得寫出這些東西，並且無私分享的人很偉大。

在這次專案中，我覺得自學的能力很重要，因為一開始，我們想要做遊戲，因此想用GUI，但是這些在這門課都沒有教到，所以我們需要自己查詢各種文件、影片等等。因為這是我們不熟悉的東西，所以常常卡關，但是很幸運的，團隊中優秀的隊友們讓我們好幾次遇到危機都化險為夷，並且學到一些背後的原理。

此外，我也發現備份的重要。有幾次，我們為了修正一些小問題而大改程式碼，這差點讓我們之前打的程式失去作用，幸好後來搶救成功，也修正了問題。

另外，我也學到「不要輕言放棄」。我負責typing遊戲時，遇到的種種困難幾乎讓我想放棄，但是組員的即時救援與不斷嘗試，終於實現這個小遊戲。當玩家打的字母可以正確顯示在螢幕上時，我真的非常感動。雖然這個感想似乎很老套，但是這真的是我在專案中深刻體會到的事。

鄭念觀：

這次的期末專題，我主要負責 reaction time 的設計，及其他部分的改良。我們使用很古老的 graphics.h 來進行介面的設計，雖然介面看起來簡潔單調，少了些許的色彩，但這不代表設計跟實作起來比較簡單，就像 CString 沒有比 C++String 簡單好用。

專題過程中，我們面臨的困難比想像多上許多，許多是因爲graphics.h 的功能不多且年代久遠，參考資料不多。例如用來偵測滑鼠點擊的GetAsyncKeyState (VK_LBUTTON) 及 ismouseclick(WM_LBUTTONDOWN) ，我們花費不少時間摸索這兩個函數的使用方式。

另外我們也花費很多精力在不同部分的整合與改良。例如，我們發現整合後不同遊戲會相互影響，在前一個遊戲點擊左鍵會導致下一個遊戲出現無法預測的結果。在這個過程中，我了解到製作一個程式專題的運作模式，除了個人部分的設計撰寫之外，最困難的是將不同部分整合為一，與小組成員測試、找出問題並討論解決方案、優化再測試，不斷循環...

最後，我要感謝我的組員們，在專題的過程中給我很多機會表達意見，還容忍我調皮的個性，像是把 aim training 的點擊聲改成小蜜蜂。他們是我在台大三年來合作過最愉快且順利的夥伴，感謝他們盡心盡力的一同完成這次的專案，希望以後還能一起合作的機會。

王建又：

在專案之初，其實和夥伴想了一些可能很需要建立GUI的專案方案，但當時不確定哪個函式庫是以我們的程度來說相對可行的。而後因為門檻較低的緣故，方才選擇使用版本老舊的<graphics.h>函式庫。或許是和編譯器版本不相容的關係，開始寫的時候才發現，寫完的程式碼沒辦法進行編譯，即使上論壇尋找解決辦法也是一籌莫展。爾後只好在無法編譯debug的情況下寫完第一個版本，在讓可以編譯的夥伴編譯並debug；所幸演算法說不上複雜，小遊戲的這部分還算是開發順利。或許在有一次這樣的機會，可能先與老師或助教討論當前為人所熟知的函式庫，會是訂下題目後要做的第一件事。

陳怡安：

這次期末專案製作我主要負責撰寫 Number Memory 的程式碼，而整個遊戲的生成也是歷經了不少波折。從一開始摸索古老套件 graphics.h 的過程，因為是全新的東西花了特別久的時間去熟悉；中間自己負責程式碼撰寫遇到的困難，可能一個小錯就要查詢好幾個網站才得以解決；一直到最後大家把程式碼合併，互相幫對方的程式碼更優化，在這個過程中常常修了一個 bug 又產生了下一個，一直重複再重複，但其中也很多 bug，讓我們有很有趣的討論，是自己寫程式所沒有的。

而製作這次專案的過程也是帶給我在程式方面很多的初體驗，一開始是因為使用 windows 才支援的套件而在 mac 上安裝雙系統的初體驗，後來也體驗到了合併每個人程式碼的初體驗，而整體來說更是我設計的第一個遊戲！因此，能製作這樣的專案讓我格外興奮，更不用說能和其他組員一起成功製作完成這個專案我真的非常高興，也很有成就感。我們組員都很 carry，也能這次合作讓我特別感謝他們，是一次氣氛很棒的合作經驗！

四、分工

專案可以大致切分成六個部分：

一、Main page 由何柏翰負責

二、Reaction time 由鄭念觀負責

三、Sequence memory 由何柏翰負責

四、Aim trainer 由王建又負責，鄭念觀支援

五、Number memory 由陳怡安負責

六、Typing 由黃照軒負責，何伯翰、鄭念觀、陳怡安支援