

習題解答

Adventure 01

一、選擇題

1. A 2. B 3. A 4. A 5. A 6. D 7. C 8. B 9. B 10. A

二、填充題

- | | |
|------------------------|----------|
| 1. 2.5、5 | 2. 0、2.5 |
| 3. D3、D5、D6、D9、D10、D11 | 4. 0、1 |
| 5. 數位 | 6. 類比 |
| 7. 數位、數位 | 8. 類比、數位 |
| 9. D14~D19 | 10. 數位 |

Adventure 02

一、問答題

從偏好設定找尋草稿碼簿路徑位置，一般都在「我的文件」當中的「Arduino」資料夾，例如「C:\Users\你的名稱\Documents\Arduino」。

安裝的方式很簡單，首先在 Arduino 的資料夾裡做一個「tools/ArduBlockTool/tool」三層的資料夾，然後把 Ardublock-beta-20140702.jar 直接放入「tools/ArduBlockTool/tool」中 tool 的資料夾裡就行了。

Adventure 03

一、選擇題

1. A 2. B 3. C 4. A 5. D 6. A 7. B 8. A 9. C 10. D

二、填充題

- | | | |
|-------------|---------|-----------|
| 1. 數位 | 2. 整數 | 3. 0~1023 |
| 4. 副函式 | 5. 1 | 6. 數位引腳 |
| 7. 設置數位 pin | 8. 類比引腳 | 9. 設置類比腳位 |
| 10. 公分 | | |

三、問答題

1. 第一個方塊：如果「條件滿足」區塊內的判斷式是真的（判斷成立時），就執行「然後執行」區塊內的程式碼。

第二個方塊：如果「條件滿足」區塊內的判斷式是真的（判斷成立時），就執行「然後執行」區塊內的程式碼。如果「條件滿足」區塊內的判斷式不是真的，就執行「否則執行」區塊內的程式碼。

2. 第一個方塊：times：想要執行的次數。

commands：要執行的程式。

如果想要程式執行幾次就可設定幾次，但是它是從第 1 次開始到 2、3、4 依序執行。

第二個方塊：這方塊跟上面不同的地方是可以自己設定起始數字和結束數字，還有每次循環的次序，可依需要做調整，可每次 1 個或 2 個或 3 個…等等跳著執行。

3. LED 燈長腳先接 1 個 220 歐姆的電阻，接杜邦線再接 S 腳，短腳接杜邦線接 G 腳。

Adventure 04

一、選擇題

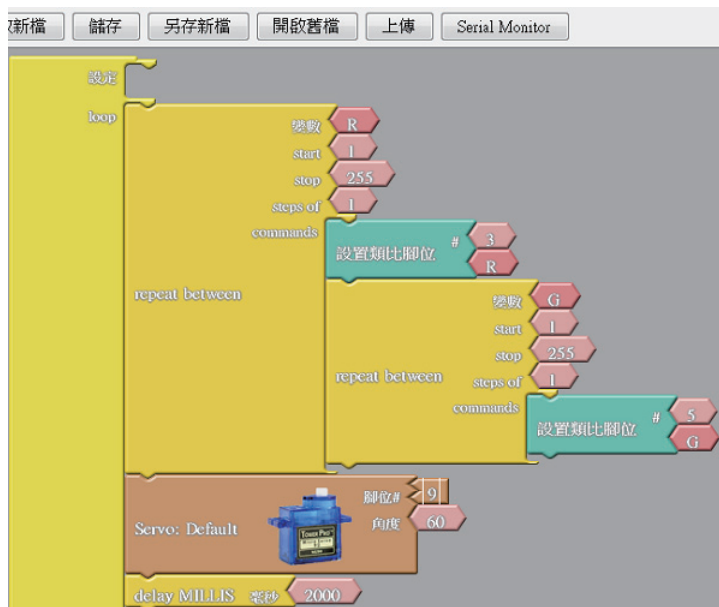
1. C 2. A 3. B 4. B 5. A 6. B 7. A 8. C 9. D 10. A

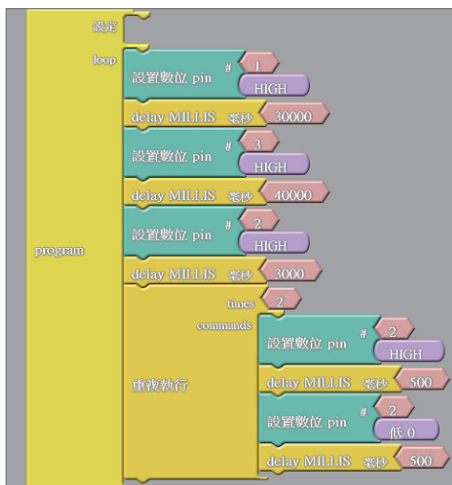
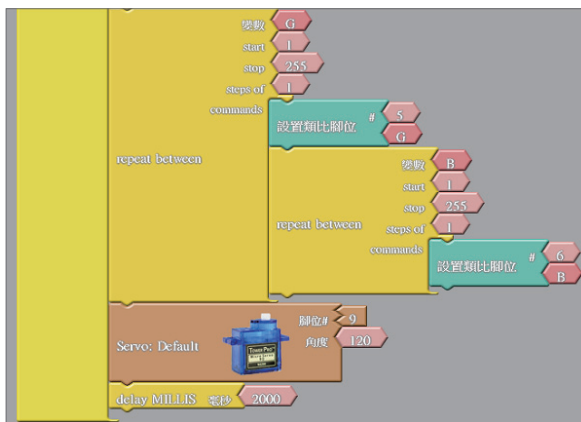
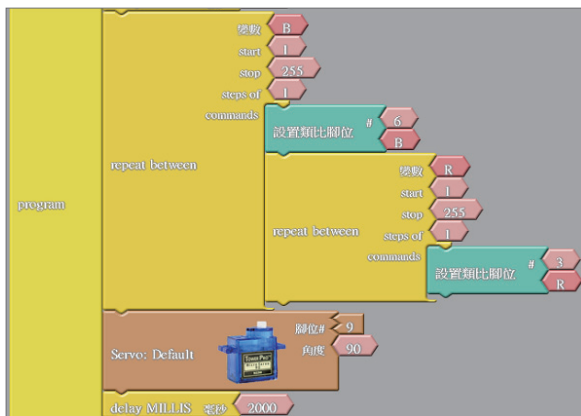
二、填充題

- | | | |
|-----------|--------------|--------------|
| 1. 列數 | 2. 行數 | 3. 逆時針 |
| 4. 順時針 | 5. 一二相 | 6. 一相，二相，一二相 |
| 7. 四個時間 | 8. 接線互換，高低電位 | 9. 八個時間 |
| 10. PWM 波 | | |

三、問答題

1. random（亂數方塊）可以隨意產生 0~1024 之間的一個數值。
- 2.





4 Ardublock + App Inventor 2 輕鬆學：玩積木寫程式，輕鬆進入 Arduino 的創意世界

Adventure 05

一、選擇題

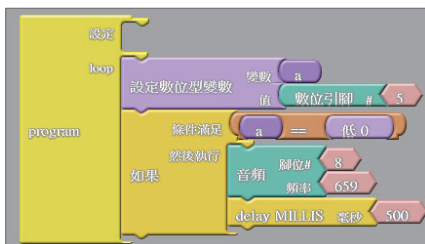
1. B 2. B 3. B 4. A 5. A 6. B 7. A 8. D 9. B 10. A

二、填充題

- | | |
|--------------------------------|------------------------------------|
| 1. 0 | 2. 0、1 |
| 3. serial println、serial print | 4. 電阻 |
| 5. 類比 | 6. $5\text{v}/1024=0.0049\text{v}$ |
| 7. 0.0049 | 8. 10.0 |
| 9. 數位、類比 | 10. 類比 |

三、問答題

- 第一個：**顯示 message 訊息，按一下可自己輸入訊息。
換行 true：可換行輸出資料
換行 false：不換行輸出資料
 - 第二個：**顯示 message 訊息，按一下可自己輸入訊息。
只有換行輸出資料，Serial println 方塊和 Serial print 方塊可用於 Arduino
輸出資料給手機，另一個 Arduino 或電腦。
- 如果想要關掉蜂鳴器只要按 RESET 按鈕就可以了。



Adventure 06

一、選擇題

1. D 2. B 3. D 4. A 5. C 6. C 7. A 8. B 9. A 10. A

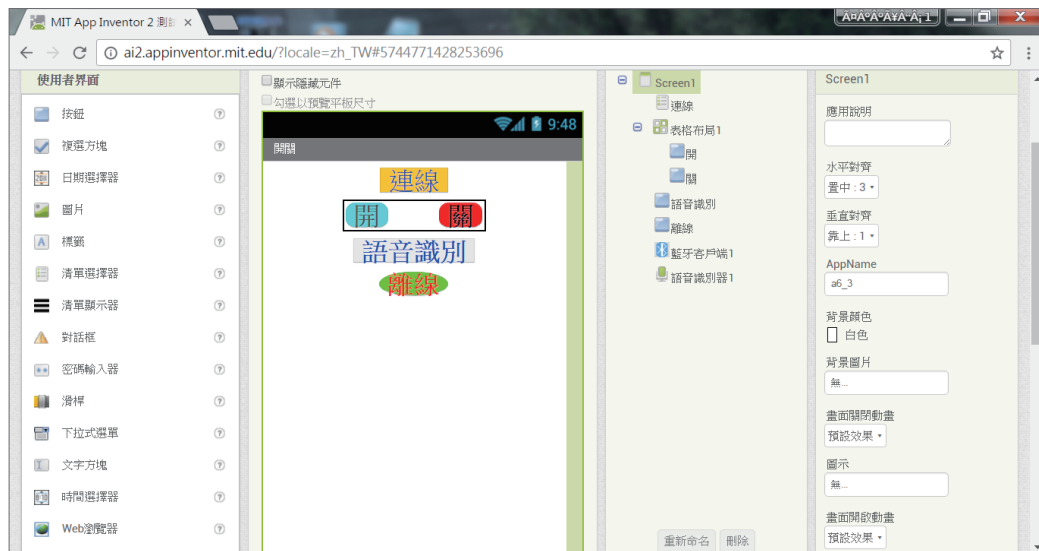
二、填充題

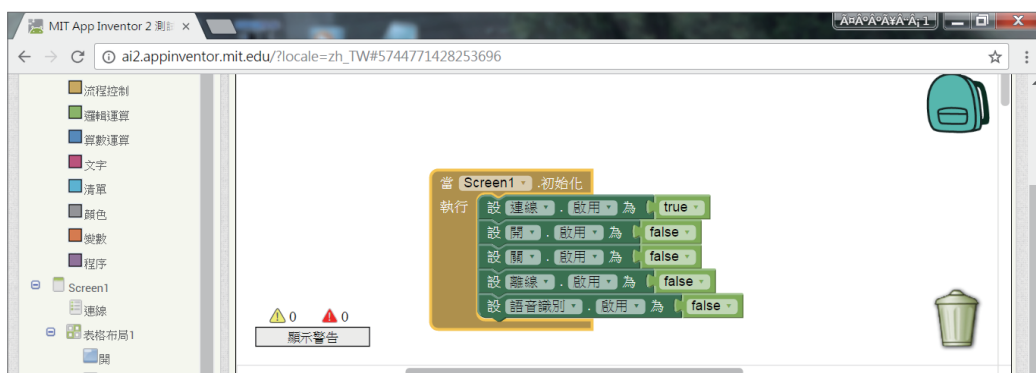
- | | |
|----------|--------------|
| 1. 我的專案 | 2. 外觀編排、程式設計 |
| 3. 清單選擇器 | 4. apk |
| 5. abp | 6. 不明來源 |
| 7. 設定 | 8. RX、TX |
| 9. D0、D1 | 10. false |

三、問答題

- (1) 打包 apk 並顯示二維條碼，再由 QRCode 掃描下載到手機。
(2) 打包 apk 並下載到電腦。

2







將滑鼠放在返回結果上等一下，就會出現求返回結果的方塊。

3. 將 D5、D8 設為高電位。

由讀取序列讀取手機傳來的訊號並把他設給字元變數 ch。

當字元變數 ch 等於 n 時 D5 上的燈會亮。

當字元變數 ch 等於 f 時 D8 上的燈會亮。

顯示 ch 內容。

延遲 1 秒以便觀察。

Adventure 07

一、選擇題

1. A 2. C 3. A 4. B 5. D 6. C 7. C 8. A 9. D 10. D

二、填充題

- | | |
|--------------------------------|----------|
| 1. serial println、serial print | 2. TX RX |
| 3. - 負 | 4. 繼電器 |
| 5. 文字 | 6. 計時器 |
| 7. 文字 | 8. 文字 |
| 9. 大 | 10. 表格 |

三、問答題

1. 程式說明：

設一個整數變數 it，把變數 it 設為 90，讓 Arduino 上的伺服馬達在開機時，先固定到 90 度。

設一個字元變 ch，用來接收手機傳入的訊息，每當接收一次 ch 等於 r 時，it 變數會加 10，然後讓馬達轉到 it 的角度，當 it 大於 170 時，將 it 固定在 170，因為伺服馬達最大角度只有 180，所以設定到 170 度。

每當接收一次 ch 等於 l 時，it 變數會減 10，然後讓馬達轉到 it 的角度，當 it 小於 10 時，將 it 固定在 10，因為伺服馬達最小角度只有 0，所以設定到 10 度

用 serial println 將角度顯示，並傳到手機。

2. (1) Screen1 先初始化→用→Screen1

拉出 Screen1 初始化，因為要連線所以先開啟清單選擇器。

清單選擇器啟用為 true，其他為 false。

(2) 使手機與藍牙連線：

a. 連線準備選擇→用→清單選擇器

用清單選擇器選擇我們配對成功的藍牙裝置，並與藍牙連線。

b. 當連線完成後，關閉連線，啟動其他按鈕→用→清單選擇器

當連線完成時把連線啟用設為關閉，其它設為開啟。

3. 當語音辨識器辨識完成時會把語音內容轉換成文字，放入返回結果內。
當藍牙有連線時，將返回結果內的文字，設給語音控制，使語音控制按鈕上會出現我們說的話。
用語音識別器判別輸入的語音是否為我們設定的字元。
如果字元相同，就傳送特定字元到 Arduino。
4. 這裡是利用計時器每 1000 毫秒自動呼叫藍牙把 Arduino 測到的溫度傳到手機上，從變數中拉出一個初始化全域變數名稱改為 tem，並接上空白的字元方塊。這樣就可以把變數的資料型態設為文字 (serial 傳送的資料型態為文字)，然後把訊息設給 tem 這個變數，把變數設給文字方塊顯示出來。
如果變數 >37.5，就會呼叫音效發出 bb 聲，比較方塊要用比較文字，不可以用算數運算，因為藍牙傳送的資料型態是文字。
5. 將 D5、D6、D7 拉升為高電位，讓流水燈熄滅。
讀取位址為 10 的儲存空間的資料，並顯示出來 (Arduino 的儲存位址為 0~1023)。
讀取位址為 10 的儲存空間的資料，放入變數 a 內。
當 a 大於 4 時，將 a 歸 0。
Arduino 每從開一次就讓 a 加 1。
將 a 的內容重新寫入位址為 10 的儲存空間。
重新讀取位址為 10 的儲存空間的資料，並顯示出來。
顯示出來的目的是為了觀察它的變化，這樣在設計程式時更容易找出有問題的地方。
Loop 區塊的程式：
如果按重置開關 1 次，a 等於 1 就亮接 D5 的 1 個燈。
如果按重置開關 2 次，a 等於 2 就亮接 D5、D6 的 2 個燈。
如果按重置開關 3 次，a 等於 3 就亮接 D5、D6、D7 的 3 個燈。