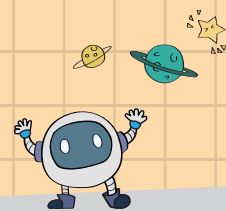




程式語言簡介



職涯導航

根據美國勞工局的研究，西元 2024 年時，美國將有一百萬個與電腦科學相關的工作職缺。

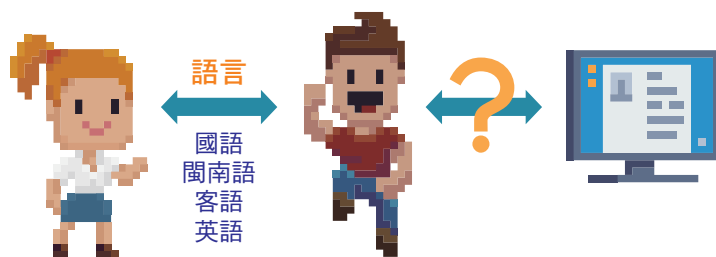
和電腦科學相關的職業有：程式設計師、研發工程師、網頁工程師、遊戲開發工程師、動畫設計師、資料科學家等。

在第 2 章中，我們學習到「演算法是解決問題的方法」，可以利用文字敘述、流程圖或虛擬碼的方式來呈現。但是，若希望利用電腦來實踐演算法，我們還要先學會和電腦溝通。



手腦並用

想一想，人與人溝通需要透過「語言」，那人與電腦之間要如何溝通呢？



1 認識程式語言

資訊科技在生活中的應用層面越來越廣，因此，每個人都應該學習與電腦溝通。程式語言（programming language）是「將演算法的步驟轉換為電腦可執行的指令，並精確地定義在不同情況下應採取何種行動」的一種電腦語言，可說是人類與電腦溝通的語言。

如同人類間存在著各種不同的語言，隨著資訊科技的演進與應用領域的不同，發展出各種程式語言，一般可分為低階語言與高階語言兩大類。

1 低階語言（low-level language）

●機器語言（machine language）

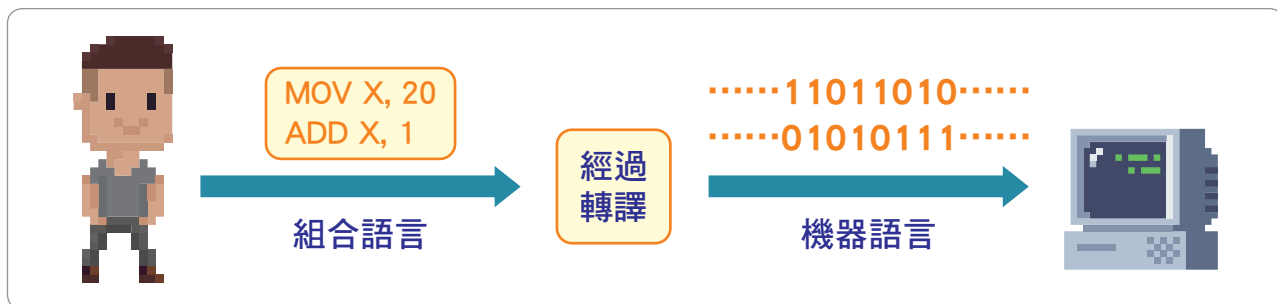
機器語言是最早的程式語言，由於電腦只認得 0 與 1 的訊號，因此機器語言的指令僅由 0 與 1 兩種符號組成（圖 1-3-1）。也因為電腦可以直接看懂機器語言，所以執行速度最快；但是，機器語言的可讀性低、不易維護，且必須針對硬體進行設計，因此不同硬體設備的電腦間無法通用，即所謂的可攜性差。



▲ 圖 1-3-1 機器語言與電腦溝通。

●組合語言（assembly language）

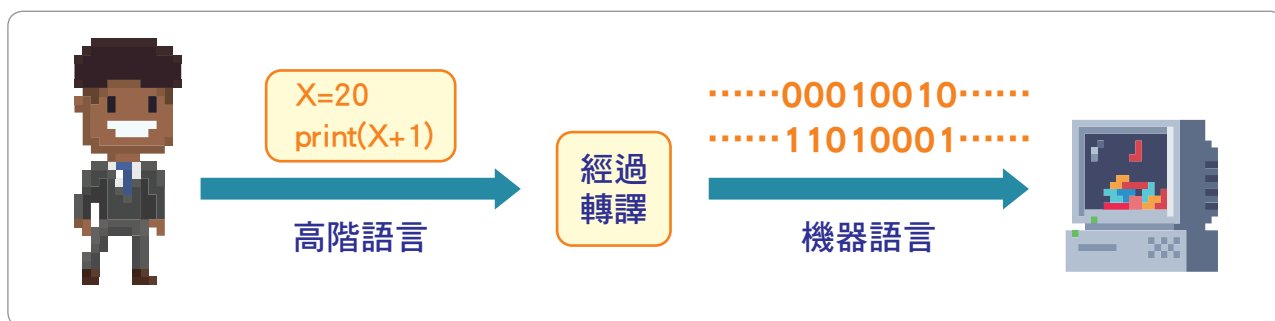
組合語言採用接近人類語言的簡短字串作為指令，比起用 0 與 1 組合的機器語言容易理解（圖 1-3-2），但因為電腦只懂機器語言，因此必須先轉譯為機器語言才能執行，所以執行速度比機器語言慢；組合語言仍屬於低階語言，可攜性差，無法在不同硬體設備的電腦間通用。



▲ 圖 1-3-2 組合語言須轉譯成機器語言才能與電腦溝通。

2 高階語言 (high-level language)

高階語言的語法較接近人類的語言，使用上易於理解，且程式可跨平臺在不同的電腦系統上執行，因可攜性佳而被廣泛應用。常見的高階語言包括 C、C++、Python、Java 等，但由於必須轉譯成電腦能理解的機器語言，因此執行效率會比機器語言低（圖 1-3-3）。

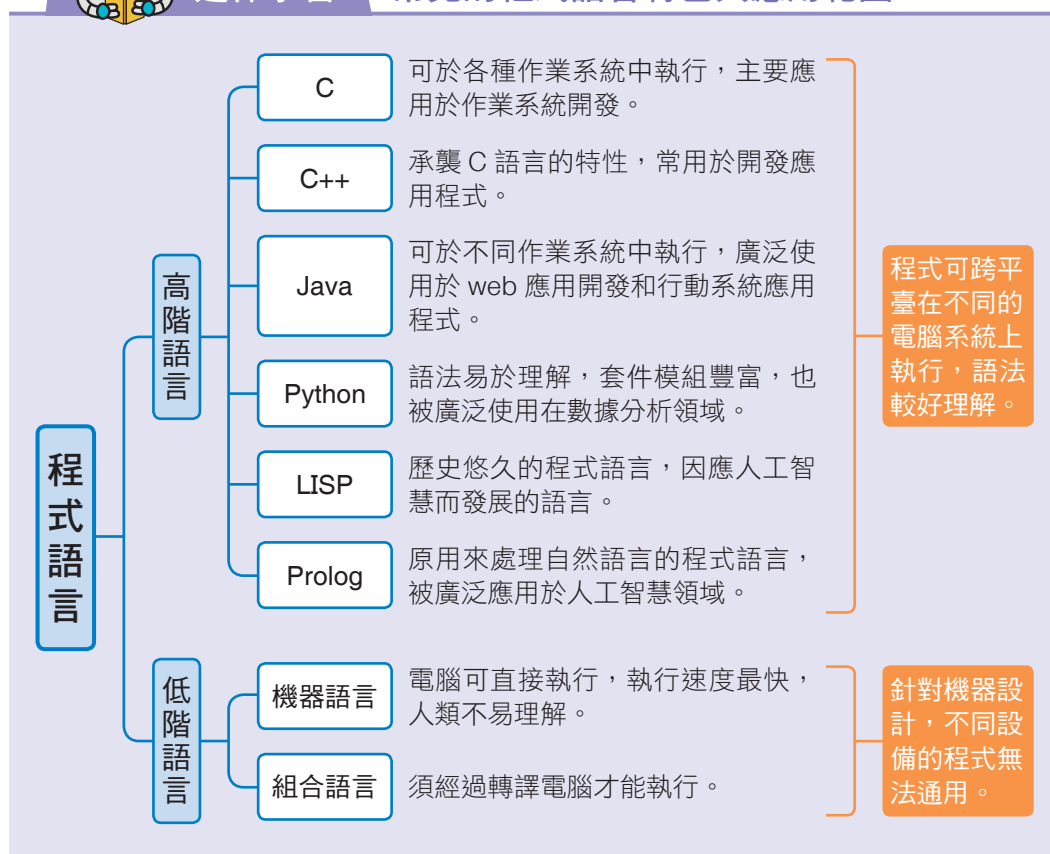


▲ 圖 1-3-3 高階語言比較容易理解，但也須轉譯成機器語言才能與電腦溝通。



延伸學習

常見的程式語言特色與應用範圍



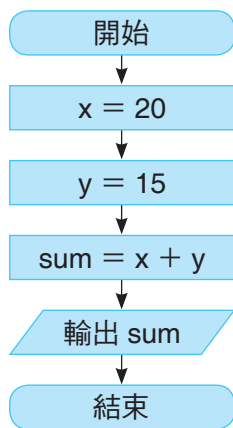
2 認識 Scratch

目前各領域廣泛應用的程式語言多為文字式程式語言，但語法較為複雜，因此學習難度高。為了降低學習門檻，我們可以選擇視覺化程式設計工具，例如 Scratch、Blockly、App Inventor 等，來學習程式的邏輯，作為未來進入文字式程式設計的基礎（圖 1-3-4）。



知識快遞

視覺化程式設計工具又稱「積木式」或「圖形化」程式設計工具，讓初學者避開文字式程式語法的學習障礙，經由積木的組合，專注於程式邏輯的學習。



A 文字式 (以 C 語言為例)

```
#include<stdio.h>
int main() {
    int x=20;
    int y=15;
    int sum=x+y;
    printf("%d",sum);
}
```

B 視覺化 (以 Scratch 為例)



▲ 圖 1-3-4 除了傳統文字式程式語言之外，現在更有視覺化程式設計工具，方便我們從演算法進入程式設計的世界。

本書選用 Scratch 作為學習撰寫程式的工具，只要將程式積木拖曳、組合，就能完成程式設計（圖 1-3-5）。我們可以使用 Scratch 來創作互動式故事、遊戲、動畫、音樂，還能在官方網站上分享自己的創意作品。

小叮嚀

本書畫面使用 Scratch3.0 版本。由於 Scratch 會不定時改版更新，因此最新版的介面、翻譯，可能與本書畫面略有不同。



▲ 圖 1-3-5 只要經由積木組合，就能完成 Scratch 程式設計；右側舞臺區呈現的就是程式執行的結果。

1 Scratch 操作介面

A 選項列

物件可編輯的項目。

1. 「角色」可編輯的項目：



2. 「舞臺」可編輯的項目：



B 程式積木區

點選左側的「積木類別」，
右方會列出相關的程式積木。

註 增加擴展積木

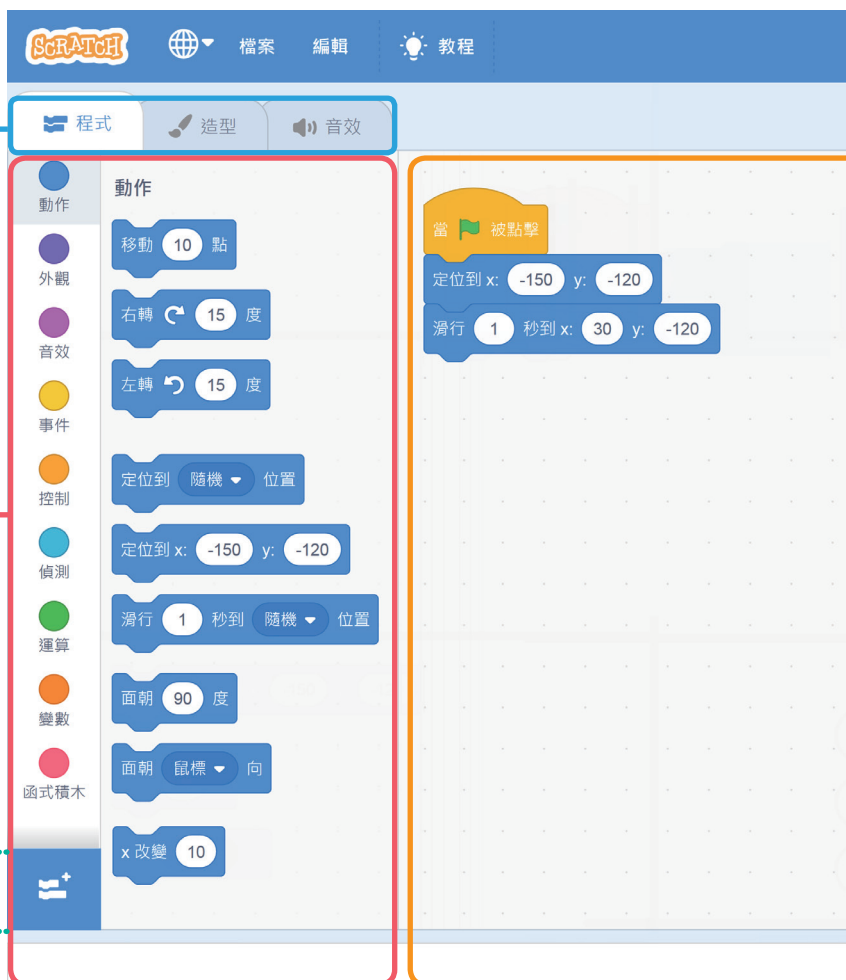
點擊 可添加擴展項目到左側類別區，例如：

- 1. 可以控制角色發出聲音、音樂相關的積木。
- 2. 可以讓角色在舞臺上繪圖的積木。

還有更多延伸工具，或與硬體設備互動的功能，請自己探索看看！



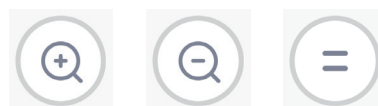
! 連結 Scratch 首頁，點擊「創造」，即可進入此介面。
<https://scratch.mit.edu/>

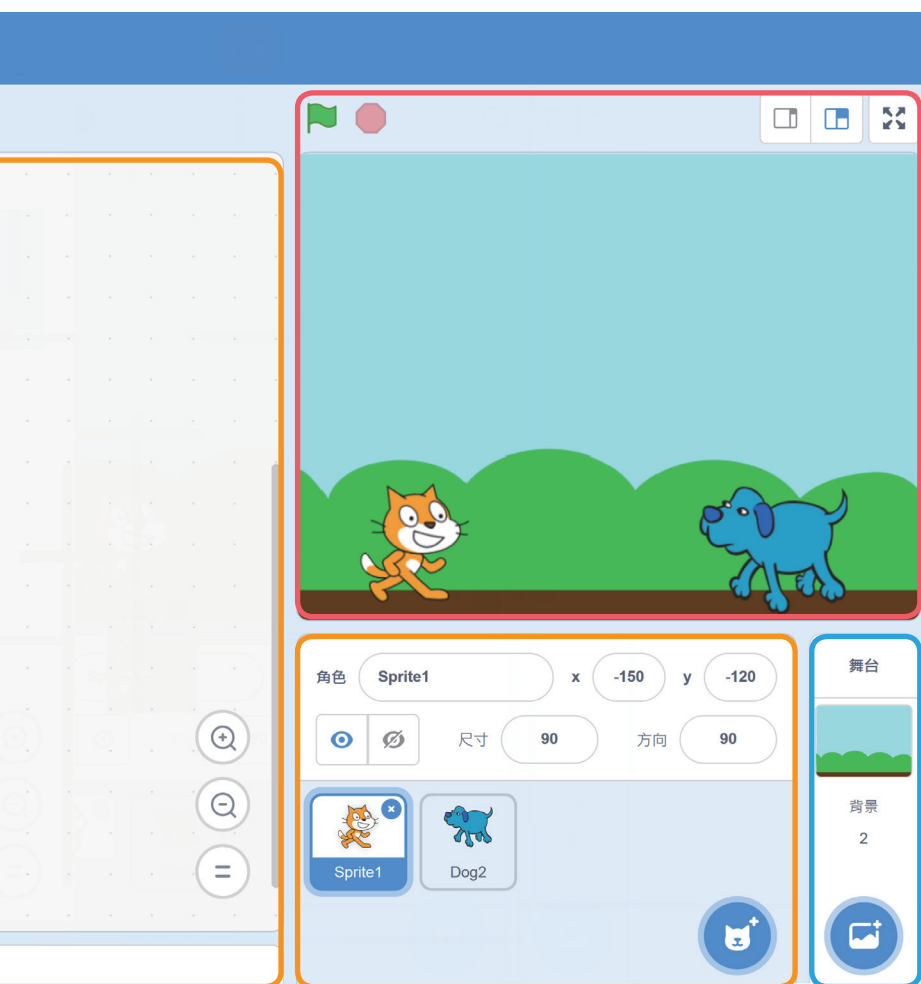


C 程式編輯區

組程式積木的地方。

可視需求控制積木顯示的大小，
方便檢視程式。





F

舞臺區

展示程式執行成果的區域。

-  1. 開始執行程式。
-  2. 停止執行程式。
-  3. 縮小舞臺區。
-  4. 放大舞臺區。
-  5. 全螢幕顯示。

E

舞臺背景區

控制舞台場景變換。

-  1. 點擊背景縮圖，進入背景控制模式。
-  2. 點擊「範例背景」鈕，直接由範例庫新增背景。

D

角色區

本專題使用的所有角色清單。

-  1. 點擊角色縮圖，進入此角色控制模式。
-  2. 點擊「選個角色」鈕，直接由範例庫新增角色。

註 新增角色／舞臺的方式

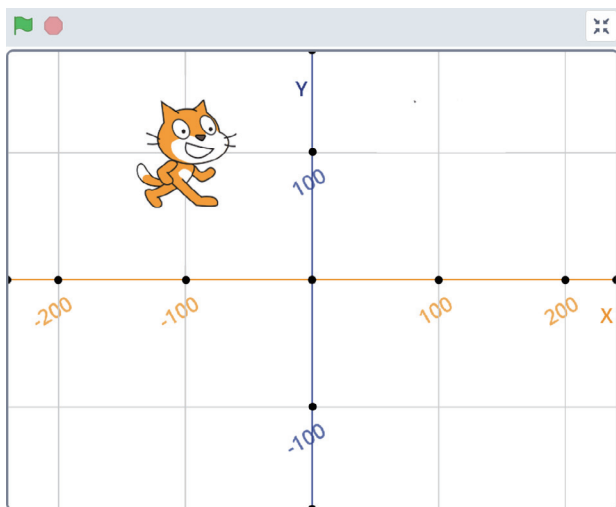
當鼠標停在按鈕上，會出現隱藏選單。

-  1. 從電腦中上傳。
-  2. 隨機挑選。
-  3. 自行繪製。
-  4. 從範例庫挑選。

2 舞臺坐標

舞臺區是展示程式執行結果的地方，而角色在舞臺上的位置可用坐標來表示。

X 軸：介於 -240~240 間
Y 軸：介於 -180~180 間



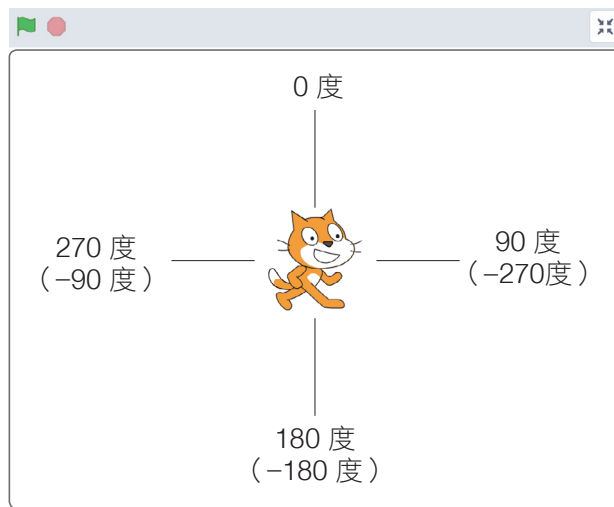
▲ 圖 1-3-6 Scratch 舞臺的坐標。

此時小貓咪的造型中心位於 X：-100、Y：100 的位置，數學上記錄為 (-100, 100)。

3 角色方向

角色面對的方向會決定移動的方向，角色預設為面朝 90 度（面向右方）。

向上（0 度）、向下（180 度）
向右（90 度）、向左（270 度）



▲ 圖 1-3-7 Scratch 角色方向。

此時小貓咪面向 90 度方向，移動時會向右方移動。



手腦並用

如右圖，直接由 Scratch 範例庫中匯入蝙蝠角色（Bat），則：

(1) 若在不調整角色方向的狀況下，程式要角色 **移動 10 點**，則蝙蝠會往舞臺的哪一個方向移動？

- (A) 上方 (B) 下方 (C) 左方 (D) 右方。

(2) 當角色 **面朝 0 度** 時，蝙蝠會呈現什麼狀態？

- (A) (B) (C) (D)

