

## 一、基本概念

1. **硬體 (Hardware)**：指一般的電子、機械設施，用手即可觸碰到之物品。

例如：電腦主機、鍵盤、滑鼠、硬碟、CPU 等。

2. **軟體 (Software)**：

(1) 應用軟體：提供特定功能、解決特定問題之軟體。

(2) 系統軟體：作為 **人與硬體** 之間溝通的橋樑，方便人們設定或管理電腦的硬體設備。

應用軟體：Word、Excel、Powerpoint、Photoshop、Dreamweaver 等

系統軟體：Windows、MacOS、android、iOS、記憶體管理、組譯器、編譯器、直譯器等

3. **韌體 (Firmware)**：作為 **軟體與硬體** 之間溝通的橋樑。**韌體是一種嵌入至硬體內的軟體**，主要用來驅動硬體進行作業。

例如：BIOS、驅動程式。

## 二、程式語言簡介

1. **第一代語言(機器語言)**：由 **0,1** 構成指令，是 CPU 唯一可以執行的語言。

**【優點】** (1) 速度最快：不需翻譯過程，可以直接執行。

**【缺點】** (1) 可攜性較差：不同的機器，可能使用不同的機器語言。

(2) 可讀性差：不僅難學、日後也難以維護。

2. **第二代語言(組合語言)**：用 **助憶碼** 取代 0,1 指令，但需要 **組譯器(Assembler)** 翻譯助憶碼。

**【優點】** (1) 可讀性較機器語言佳：相較機器語言而言，比較容易學習，但仍有難度。

(2) 能充分利用硬體：組合語言與機器語言為一對一的對應。

**【缺點】** (1) 可攜性較差：不同的機器，可能使用不同的組合語言。

(2) 執行效率較機器語言差：需要組譯器翻譯成機器語言。

3. **第三代語言(高階語言)**：又稱為**循序語言**，比較接近人類的表達方式，但需要 **編譯器(Compiler)** 或 **直譯器(Interpreter)** 翻譯程式。

**【優點】** (1) 可讀性佳：容易學習。

(2) 可攜性佳：可使用不同的編譯器或直譯器翻譯給 CPU 執行。

**【缺點】** (1) 執行效率差：需要編譯器或直譯器翻譯成機器語言。

(2) 較佔記憶體空間：程式碼較冗長。

(3) 無法充分利用硬體：與機器語言沒有一對一的對應。

4. **第四代語言(非程序性語言)**：通常應用於大型資料庫，使用者只需要透過資料庫查詢的語法，直接對資料庫下達查詢的指令。例如：SQL(Structured Query Language 結構化查詢語言)。

5. **第五代語言(人工智慧語言)**：又成為**自然語言**，人類口語化的語言，也是程式語言發展的最終目標。

表 1. 程式語言特色比較

	語法	可讀性	執行速度	可攜性	充分利用硬體	翻譯器
機器語言	0,1 組合	③	①	差	可	不需要
組合語言	助憶碼	②	②	差	可	組譯器
高階語言	接近人類	①	③	佳	無	編譯器、直譯器

### 三、編譯器、直譯器：將高階語言翻譯成機器語言

1. **編譯器**：翻譯前，會先檢查程式語法，若無誤則會進行翻譯，最後產生執行檔供執行。

**【優點】** 執行效率佳：每一次執行，不需要重新翻譯。

**【缺點】** 不容易除錯：程式設計師必須自行尋找程式修改，再重新翻譯。

2. **直譯器**：採一邊翻譯、一邊執行的方式，若有錯程式會立即停止，並指出錯誤的程式碼。

**【優點】** 容易除錯：會立即指出錯誤的程式碼。

**【缺點】** 執行效率差：每一次執行，皆需要重新翻譯。

### 四、程式設計的步驟

在寫程式之前，必須釐清問題並找出適合的解決方法，最後將解決問題的**步驟寫出來**或**畫出來**，我們則稱為**演算法**，簡單地說演算法就是一種**解決問題的邏輯思維**。例如：「把蘋果做成一杯蘋果汁」。

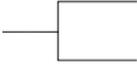
方式一：文字條列化	方式二：圖像化
<ol style="list-style-type: none"> <li>1. 清洗蘋果</li> <li>2. 將蘋果削皮、去籽</li> <li>3. 將經過步驟(2)處理的蘋果放入果汁機</li> <li>4. 在果汁機中加入一定比例的水</li> <li>5. 按下果汁機啟動按鈕</li> <li>6. 將果汁機裡面的蘋果汁倒入玻璃杯中</li> </ol>	

1. 設計演算法的時候，必須掌握五個條件：

條件	說明
輸入(Input)	0 個或多個輸入資料。
輸出(Output)	至少一個輸出結果。
明確性(Definiteness)	清楚、不會模稜兩可。
有限性(Finiteness)	有限步驟之後一定會結束。
有效性(Effectiveness)	能用紙跟筆記算出答案。

## 2. 流程圖符號：

流程圖符號	名稱及意義	使用範例
	開始或結束	開始 結束
	輸入或輸出	印出A 輸入A
	處理	$S=A+B+C$
	決策判斷	$A>B$ 假 真
	迴圈	$I=1, 6, 2$

流程圖符號	名稱及意義	使用範例
	副程式	副程式
	註解	S代表總和
	列印	印出報表
	流向	
	連接	
	人工作業	如：資料收集、人工檢閱
	螢幕輸出	在螢幕上顯示資料
	儲存資料	使用磁碟存取資料