編輯大意

一、本書係依據民國107年教育部發布之「十二年國民基本教育課程綱要——國民中學暨普通型高級中等學校——科技領域——資訊科技」編寫而成。

二、全書共一冊，提供普通型高級中等學校使用，共2學分。內容包含：

演算法　程式設計　系統平台　資料表示、處理及分析　資訊科技應用　資訊科技與人類社會

三、本書編寫目的為透過電腦理論與資訊科技實作，訓練學生運算邏輯與系統思考之能力，掌握以資訊科技解決日常生活問題的能力。其中，本書使用貼近生活的實例，透過程式碼的編寫，教導學生解決問題，以期學生具備足夠的練習機會，並激發學生對於資訊科技的興趣，未來得以善用科技知能進行創造、批判、邏輯運算等思考，並培養學生正確使用科技的觀念、態度、工作習慣與能力，成為善用資訊科技、從中反省實踐，且樂於與他人共同創作的數位公民。

四、本書除正文外，另輔以下列三大項目，以期豐富書籍內容，提升學生知能：

科技即戰力

在課文段落之間穿插相關題目，同時輔以參考解答，以利學生即時實作、解題，引導學生檢視自身學習狀況，並建立正確、有邏輯的系統思考方式。

IT 補給站

延伸補充課文中提及的字詞、相關概念或案例解析，充實學生的相關知識，提升學生進行自主學習的動力。

課堂檢測

供學生在各章課程結束後，評量所學之深淺程度，提高學習成效，亦有助於任課教師了解學生的學習狀況。

五、各大章節皆以文字說明為主，流程圖等詳盡生動的圖片描述為輔，藉以在教學過程中加深學生印象；同時也透過圖形或表格式統整，系統化、有效率地建立資訊科技之觀念。

六、本書於內容或編排上若有未盡妥善之處，尚祈各界先進不吝賜教，俾供再版修訂、改進之參考，是幸。

編者　謹誌

目次

主題一 演算法 4

chapter 01 重要資料結構的概念與應用 6
1-1 陣列（Array） 7
1-2 堆疊（Stack）、佇列（Queue） 9
1-3 樹（Tree） 12
1-4 圖（Graph） 30

chapter 02 重要演算法的概念與應用 38
2-1 演算法簡介 39
2-2 常見的解題策略與概念應用 44
2-3 遞迴結構的概念與應用 51
2-4 演算法效能分析 61

主題二 程式設計 68

chapter 03 程式設計概念與實作 70
3-1 程式設計的概念 71
3-2 程式設計工具 72
3-3 變數的概念 79
3-4 變數的運算 83
3-5 結構化程式設計 92
3-6 模組化與函式的概念 114

chapter 04 串列資料結構的程式設計實作 122
4-1 一維串列實作 123
4-2 二維串列實作 129
4-3 串列的應用 131

chapter 05 重要演算法的程式設計實作 142
5-1 排序法實作 143
5-2 搜尋法實作 156
5-3 遞迴實作 162

主題三 系統平台 166

chapter 06 系統平台之運作原理 168
6-1 系統平台的概念 169
6-2 程序管理與資源分配 172
6-3 記憶體的配置與管理 177
6-4 分散式系統（Distributed Systems） 180

chapter 07 網際網路運作原理 186
7-1 網路的架構 187
7-2 通訊技術的演進 193
7-3 通訊協定 196
7-4 網路路由 199

chapter 08 系統平台之未來發展趨勢 204
8-1 AI 人工智慧 205
8-2 雲端運算 209
8-3 IoT 物聯網 214

主題四

資料表示、處理及分析　222

chapter 09 **巨量資料的概念與應用　224**
9-1　巨量資料的概念　225
9-2　巨量資料的蒐集與取得　230
9-3　巨量資料的處理與分析　234
9-4　巨量資料的應用　237

chapter 10 **資料探勘與機器學習的基本概念　242**
10-1　資料探勘　243
10-2　機器學習的概念與應用　251

主題五

資訊科技應用　260

chapter 11 **專案管理與數位協同共創　262**
11-1　專案管理的概念與應用　263
11-2　版本控制的概念　271
11-3　協同共創工具與實作　276

主題六

資訊科技與人類社會　286

chapter 12 **資訊科技合理使用原則　288**
12-1　智慧財產權　289
12-2　創用 CC（Creative Commons）與軟體授權　294

chapter 13 **個人資料的保護　302**
13-1　個人資料保護法　303
13-2　網路上的隱私與保護　308

chapter 14 **資訊科技對人類社會的影響與衝擊　316**
14-1　對就業環境的影響與衝擊　317
14-2　對文明社會的影響與衝擊　320
14-3　對未來的展望　327

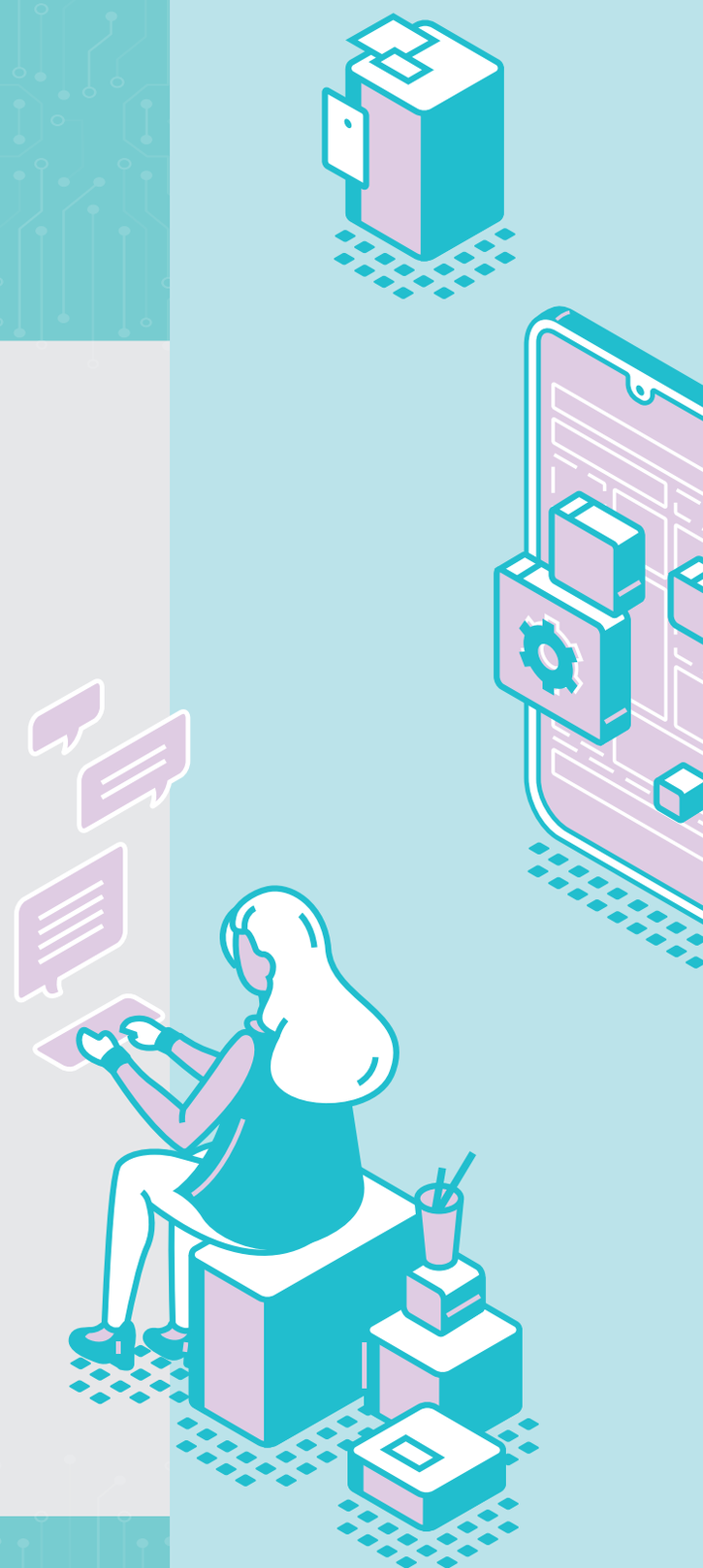
索引表　332

主題一

演算法

chapter 01
重要資料結構的概念與應用 006

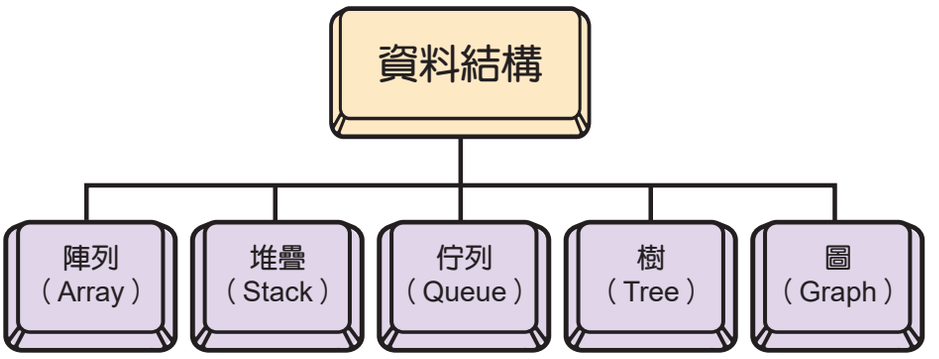
chapter 02
重要演算法的概念與應用 038



chapter 01

重要資料結構的概念與應用

資料結構（Data Structure）主要是用來規劃如何將資料（Data）有組織地存放在記憶體中，以提升其解決特定問題時的執行效能。在這單元中，我們將依序介紹陣列、堆疊、佇列、樹、圖。



1-1 陣列（Array）

在日常生活中，我們經常運用連續的表格記錄相同型態的資料，並藉由表格的呈現，使得資料的記錄或查詢更為簡潔便利，例如：老師的點名單、班級成績單、學生身高體重紀錄表等表格資料。以點名單為例，其中會有座號資料欄位和姓名資料欄位，此即構成基本的**一維陣列**。這份點名單中的座號是每個人在班級中專屬的序號，當我們要查詢某位同學的姓名，只要透過座號，就可以查詢到相對應的姓名資料，因此座號與姓名具有一對一的專一性（圖1-1-1）。

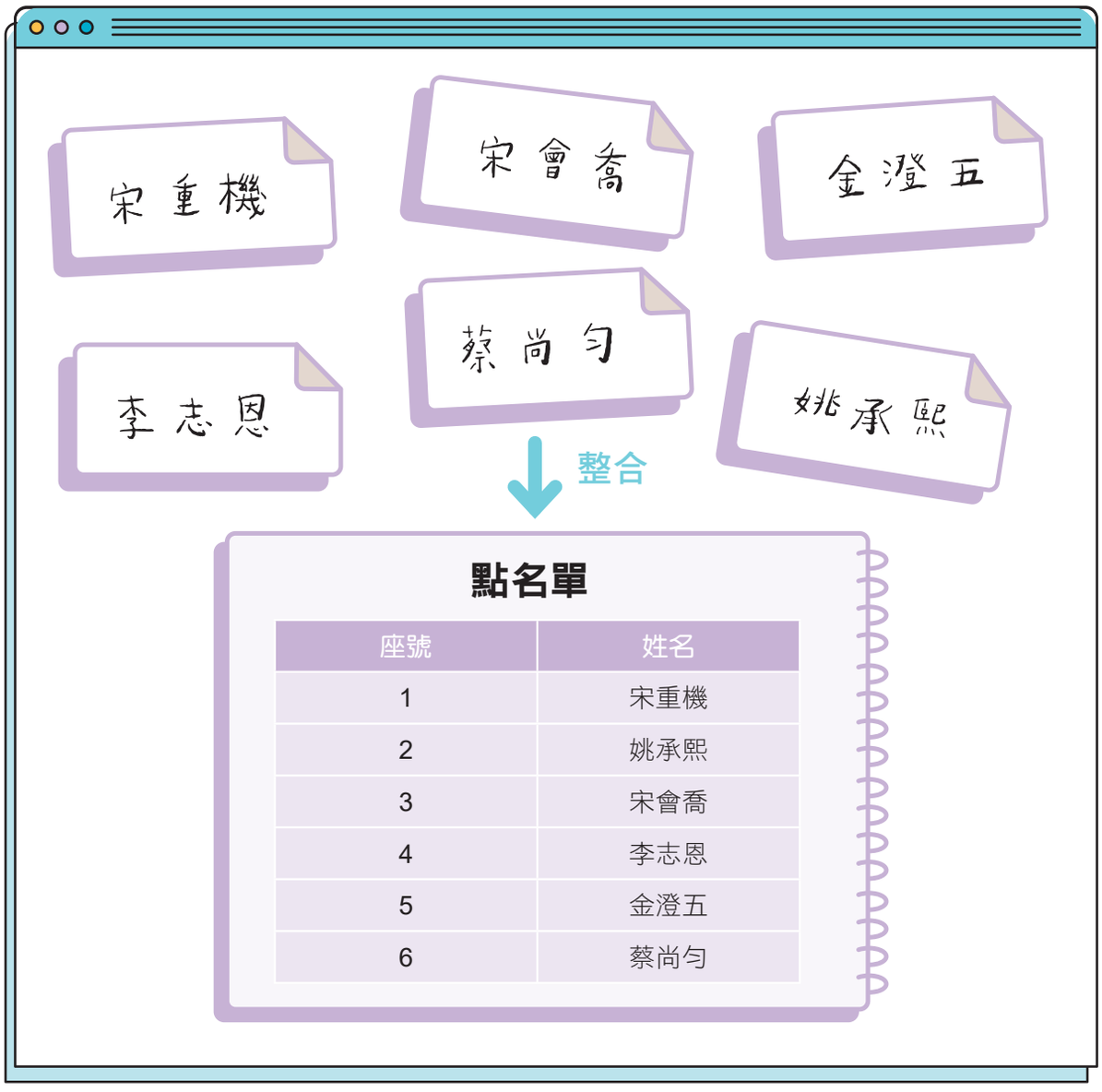


圖1-1-1 數張點名卡整合為使用表格和座號排序的點名單較為簡潔，且可讀性高

陣列是指將一群相同型態的資料放在連續的表格中，並且可以透過**索引值**（序號，Index Value）存取其中的**元素**（資料，Element）。在陣列的應用中，上述這份點名單的座號稱之為索引值，而索引值所查詢到的相對應資料稱之為元素。例如：圖1-1-2這份點名單的「索引值4」，其對應元素就是「李志恩」。

點名單

座號	姓名
1	宋重機
2	姚承熙
3	宋會喬
4	李志恩
5	金澄五
6	蔡尚勻

索引值　　元素

圖1-1-2　點名單中的索引值和元素

倘若有許多相同大小的一維陣列資料，我們可以將其組合成**二維陣列**的型態。例如：圖1-1-3的班級週課表，每天都有6節課程，每一天皆視為一維陣列，再把5天組合成二維平面的陣列，記錄6×5=30節課程資料。若查詢「星期二的第三節」，則可以知道該時段是「資訊課」，因此二維陣列須以兩個索引值查詢相對應的元素值。

二年17班　課表

節次＼星期	星期一	星期二	星期三	星期四	星期五
1	國文	英文	英文	歷史	地理
2	國文	資訊	英文	歷史	地理
3	自主學習	資訊	體育	公民	物理
4	自主學習	地科	體育	公民	物理
5	數學	生物	家政	數學	化學
6	化學	生物	家政	數學	地科

索引值　　索引值

將五個一維陣列，組合成一個二維陣列

共 6×5=30 筆元素

圖1-1-3　班級週課表

1-2 堆疊（Stack）與佇列（Queue）

1-2-1 堆疊

當我們將書本由下而上的放入（Push）箱中並疊起，或從箱中取出（Pop）書本的過程，皆在箱中的最頂端進行書本的增減，這種無論放入或取出皆是在最上層（頂端，Top）發生，並遵守後進先出（Last In First Out，LIFO）的概念，即為**堆疊**（圖1-2-1）。

堆疊在資訊科技當中，實務上的應用例子如瀏覽器中「回到上一頁」的按鈕，可以將每次瀏覽網頁所經過的頁面網址（Push），以堆疊的資料結構儲存於記憶體中。當使用者按下「上一頁」，則將堆疊最上層的網頁（Pop出前一個瀏覽的網址），重新載回瀏覽器的頁面；編輯文件或圖片時，我們也常會運用「還原」的功能，讓文件回到上一個步驟的編輯狀態，藉此還原先前的資料或圖片。

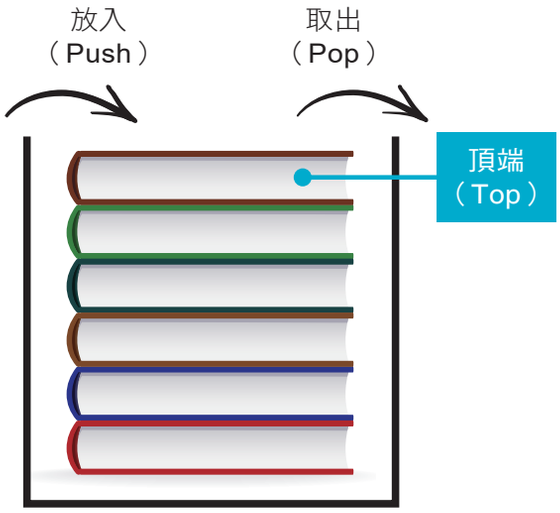
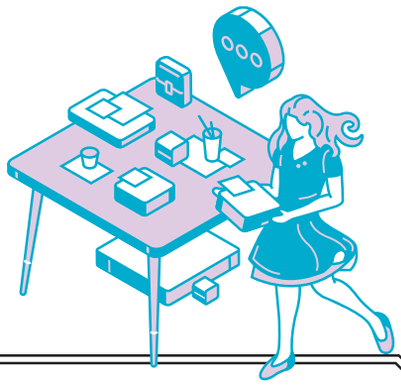


圖1-2-1

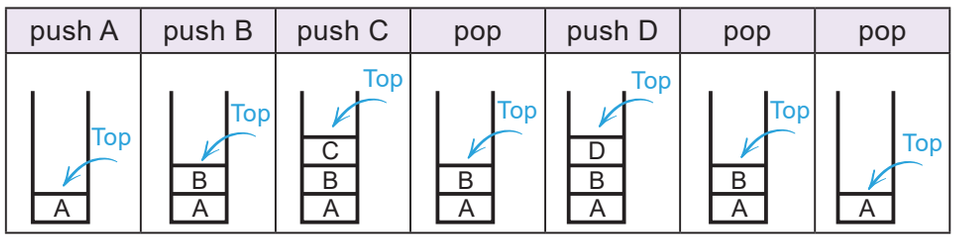
將書本放入箱中，並由頂端進行拿取，此為堆疊的應用。



科技即戰力

請問經過以下指令：「push A，push B，push C，pop，push D，pop，pop」，最後堆疊內部結果為何？

實作解析 1-2-1



答案：最後只剩下「A」在此堆疊內部。

1-2-2 佇列

飛機在機場跑道等候起飛時，機場塔臺常以「進入跑道排隊的先後順序」作為「起飛的先後次序」。首架進入跑道排隊的飛機，就是最優先起飛的架次；而最後進入跑道隊伍尾端者，則需等待前面的架次起飛後才能起飛。此種先進先出（First In First Out，FIFO）的方式，即為**佇列**結構（圖1-2-2～圖1-2-3）。

有別於堆疊的加入與刪除資料都在同一端點進行，佇列加入資料（Enqueue）於尾端（Rear）進行，刪除（Dequeue）資料則於佇列的前端（Front）進行。

而佇列在資訊科技的實務應用如作業系統的工作排程，可以將先載入記憶體等待CPU執行的工作，優先放入CPU運算（Computation）並執行；印表機的任務排程也是相同原理，先送入印表機緩衝記憶體的工作，會優先被列印出來（圖1-2-4）。

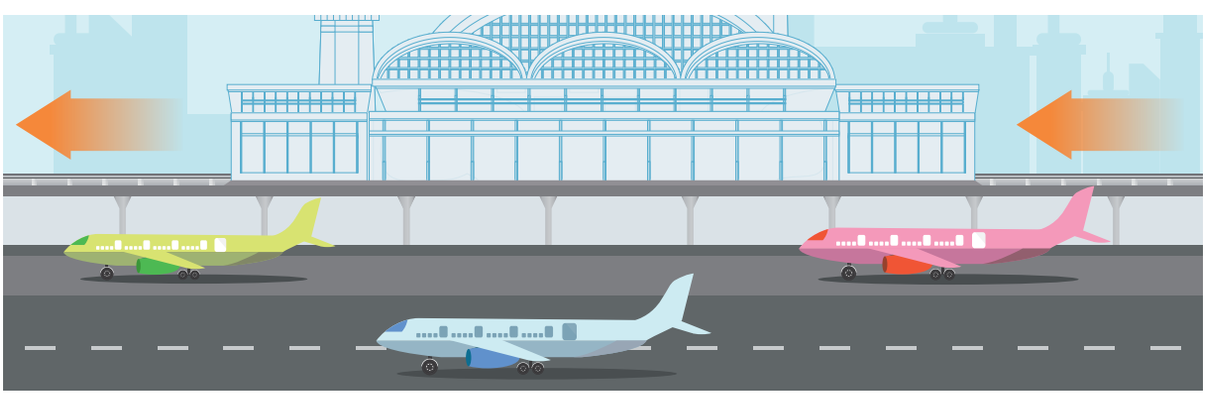


圖1-2-2 機場塔臺指揮跑道上的飛機起飛是「先進先出」的概念

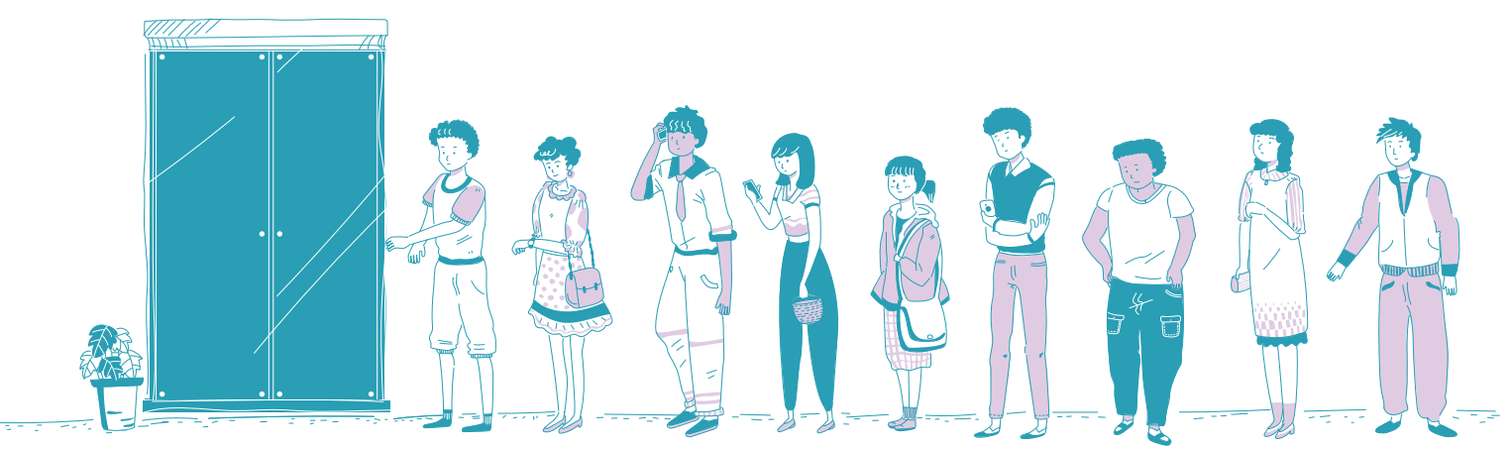


圖1-2-3 排隊隊伍遵守「先進先出」的規則

圖1-2-4 印表機的任務排程也是遵守佇列的規則，以送交時間的先後次序，依序執行列印

科技即戰力

有一佇列經過以下指令：「加入 A，加入 B，加入 C，刪除，加入 D，刪除，刪除。」請問最後該佇列的內部結果為何？

實作解析1-2-2

Step1. 加入A

Front A Rear

Step2. 加入B

Front AB Rear

Step3. 加入C

Front ABC Rear

Step4. 刪除

Front BC Rear

Step5. 加入D

Front BCD Rear

Step6. 刪除

Front CD Rear

Step7. 刪除

Front D Rear

（此Queue的最終結果）

1-3 樹（Tree）

樹狀結構是一種仿效自然界樹木生長模式，由最高層級開始逐層分支、擴散開來的階層架構。在生物學中的「生物分類法」就是一種樹狀結構的應用，以界、門、綱、目、科、屬、種的分支作為樹狀延伸的依據，例如：動物界、脊索動物門、哺乳綱、食肉目、貓科、豹屬、豹種（圖1-3-1）。由此可知，我們可以運用樹狀結構的概念，將一群目標資料透過特定規則進行分類，並儲存於樹狀結構當中。

而在資料結構中的樹則和自然界的樹木生長方向相反：資料結構的樹，最上方的層級為樹根，並且向下分支擴散，形成「樹根在最上層，樹葉在最底層」的倒立樹狀結構（圖1-3-2）。如微軟Windows作業系統的檔案儲存結構（圖1-3-3）、組織架構圖（圖1-3-4）、單淘汰制賽程表、字典樹（Trie，圖1-3-5）等，皆為樹狀結構的應用。

圖1-3-1 生物分類法

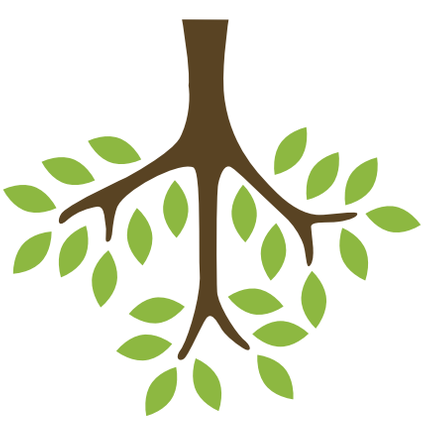


圖1-3-2 倒立的樹狀結構

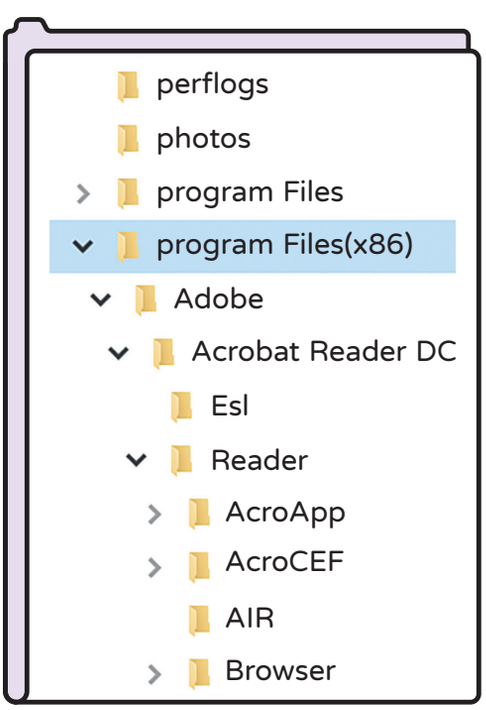


圖1-3-3 微軟Windows作業系統的檔案儲存結構

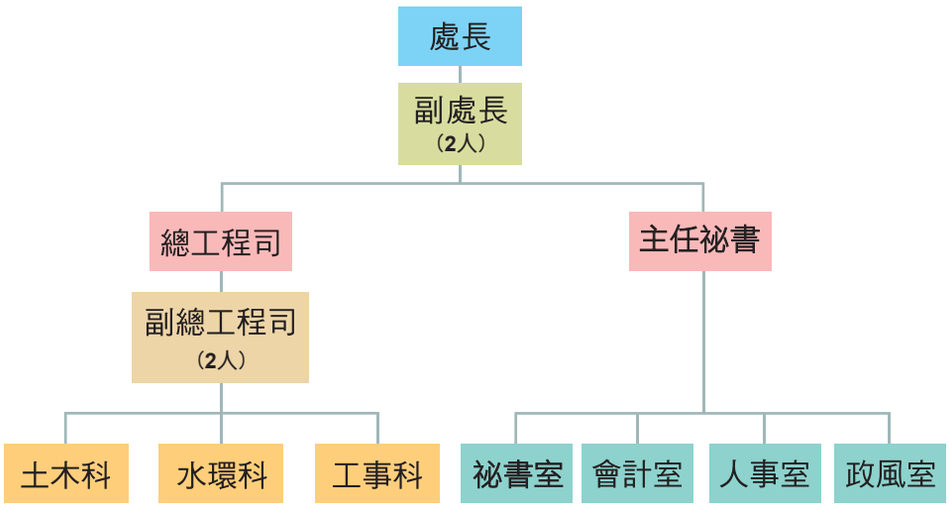


圖1-3-4 組織架構圖

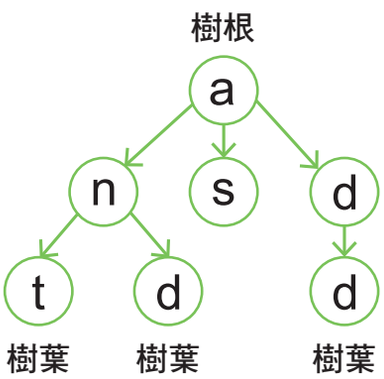


圖1-3-5 字典樹（各字母間的連接箭頭代表搜尋時的單一方向性）

在資料結構中，**樹**是由**節點**（Nodes）和**邊**（Edges）相連形成的資料結構（圖1-3-6），且至少由一個根節點（Root）所形成的集合。

每個節點可以衍生出0個以上的子節點。如左圖1-3-6，節點A有2個子節點（B、C）與其相連。

而根節點以外的節點亦可分為多個沒有交集且各自獨立的子集合，每個子集合又可稱為子樹。例如：在圖1-3-6中，節點B、D與節點C、H皆為節點A的子樹。

樹的任兩節點相通，且只有一條路徑，亦不會產生**循環路徑**（Cycle）。如下圖1-3-7的節點B、C若相連，則節點A、B、C會產生封閉循環。

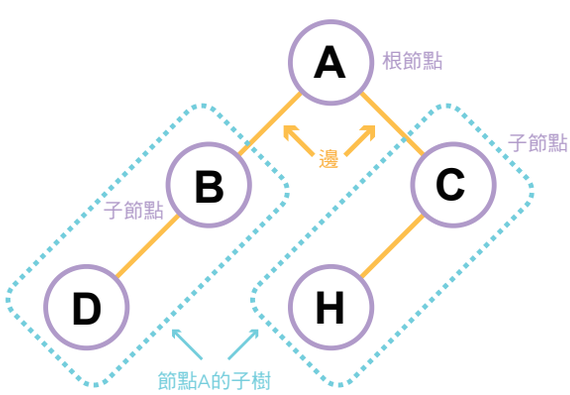


圖1-3-6 節點與邊的關係

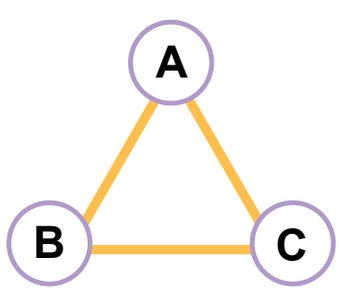


圖1-3-7 樹不會產生循環，亦即不會產生起點與終點相同的迴路。

1-3-1 樹的專有名詞

資料結構的樹有多個專有名詞，讓我們透過圖 1-3-8 分別介紹如下：

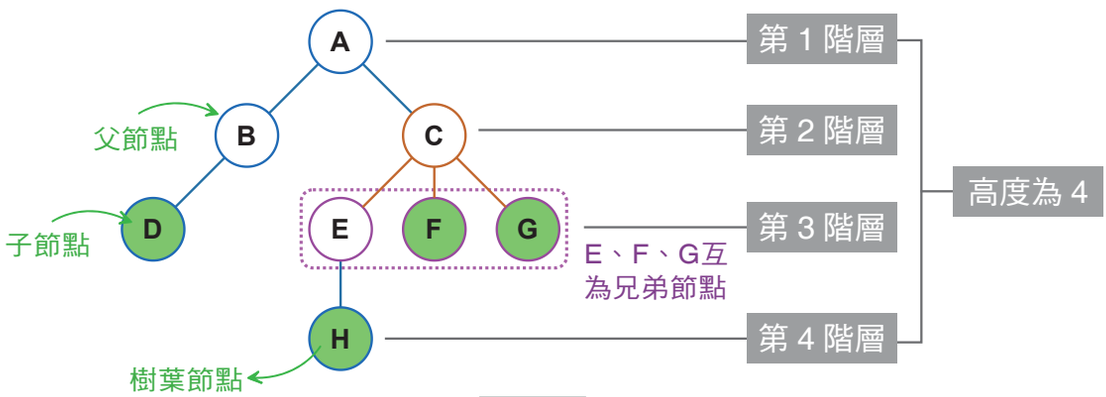
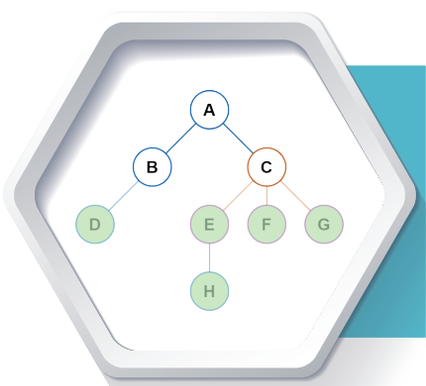
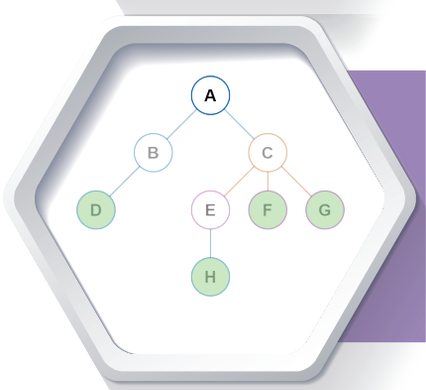


圖1-3-8 樹



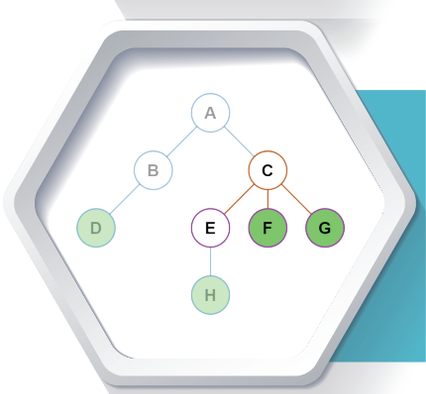
父（Parent）節點

由該節點向上回溯的上一層節點。如節點B、C的父節點皆為A。



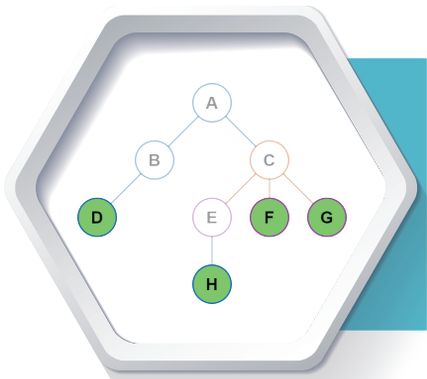
樹根節點（Root）

在樹的最頂層且沒有父節點的節點。如節點A。



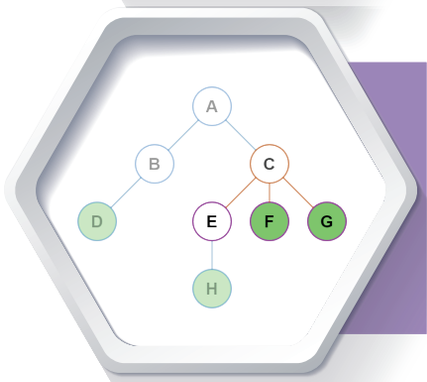
子節點（Child）

由該節點所衍生出的下一層節點。如節點C的子節點為節點E、F、G。



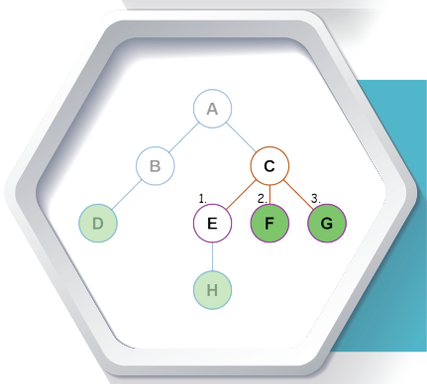
樹葉（Leaf）節點

在樹的最底層，且沒有子節點的節點。如節點D、H、F、G。



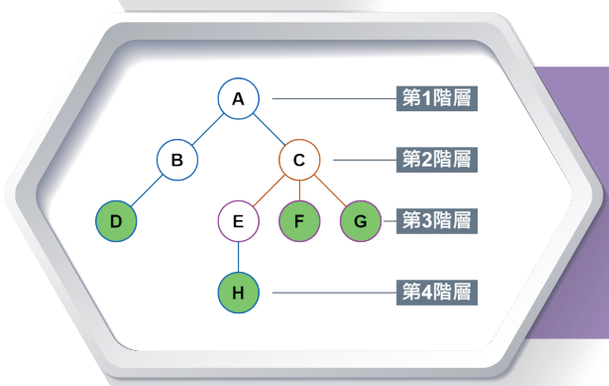
兄弟（Sibling）節點

具有相同父節點的所有節點。如節點E、F、G的父節點皆為C，故E、F、G互為兄弟節點。



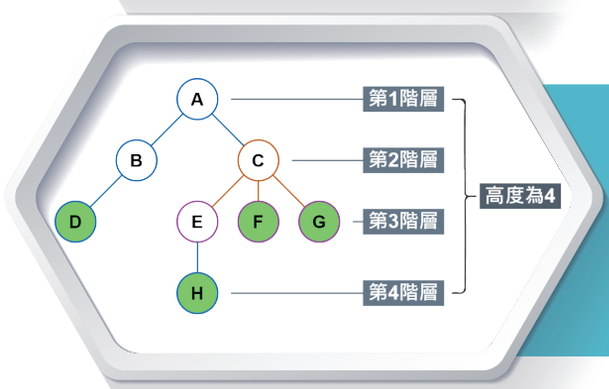
節點分支度（Degree）

該節點所擁有的子節點數。如節點C的分支度為3。



階層（Level）

根節點為第1階層，而其所衍生的子節點為第2階層，依此類推。如H的所在位置為第4階層。



高度（Height）

此樹的最大階層數，又稱深度（Depth）。如圖1-3-8樹的高度為4。

1-3-2 二元樹（Binary Tree）

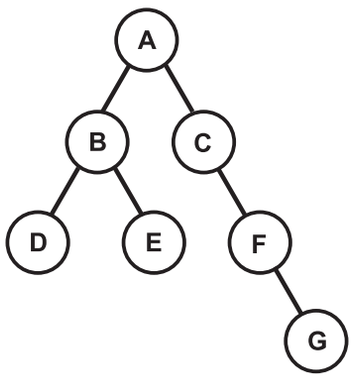


圖1-3-9 二元樹示意圖

樹依據不同的分支度，可分為許多類型，例如：二元樹（圖1-3-9）、四元樹（圖1-3-10）、八元樹等。而在樹狀結構的應用中，「**二元樹**」的應用較為廣泛，由於其「最大分支度為2」的特性，可應用在排序（Sort）、搜尋（Search）、編碼等方面，例如：搜尋演算法之一的**二元搜尋樹**（Binary Search Tree）以及資料壓縮所使用到的**霍夫曼樹**（Huffman Tree）等；而四元樹、八元樹等，本質上仍可以運用二元樹進行實作。以下就讓我們以二元樹實作四元樹作為範例（圖1-3-11）。

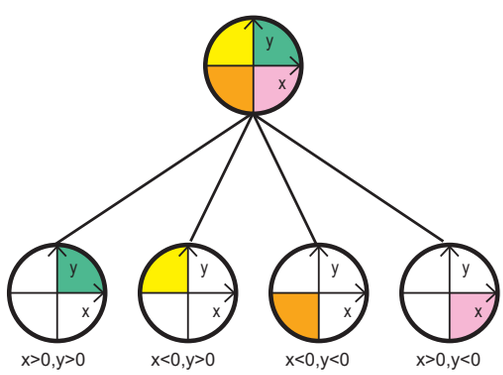


圖1-3-10 四元樹範例。四元樹可應用於平面座標的象限分類，恰好可以區分四個不同象限。

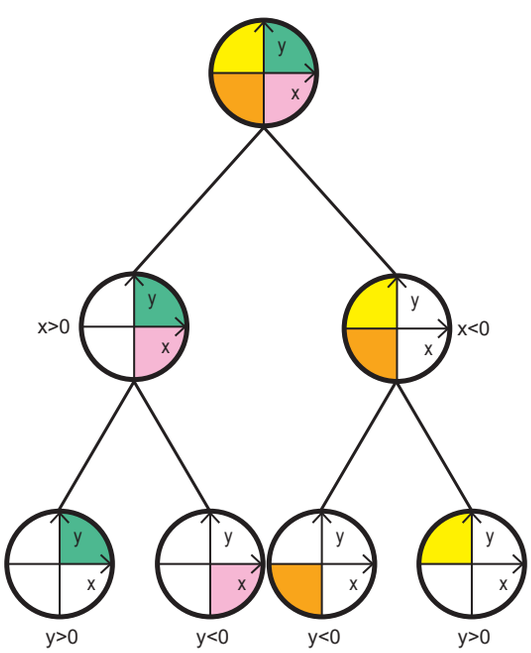


圖1-3-11 運用二元樹實作四元樹，可以先以x>0或x<0劃分為兩部分。由這兩部分分別向下，再以y>0或y<0劃分，最後同樣可形成四個象限的分配。

二元樹

1. 二元樹即為樹中每一個「節點」的子節點，數量為0～2個。
2. 二元樹除了根節點外，可以分成兩個獨立子樹，稱為「左子樹」和「右子樹」。以圖1-3-9為例，左子樹（圖1-3-12）、右子樹（圖1-3-13），分別如下圖所示，亦符合二元樹。

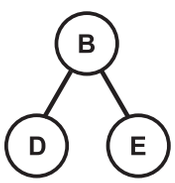


圖1-3-12 左子樹

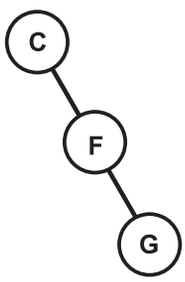


圖1-3-13 右子樹

二元搜尋樹

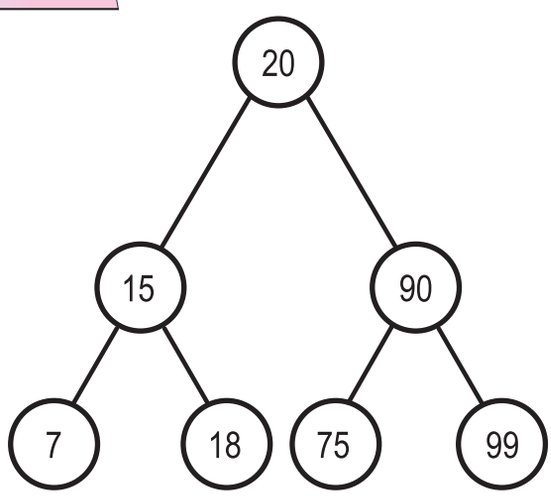


圖1-3-14 二元搜尋樹示意圖

「**二元搜尋樹**」是二元樹的一種延伸應用，其目的在於提供資料的新增、刪除與搜尋等操作。並且會依據下列規則，將數字依序排列，形成二元搜尋樹：

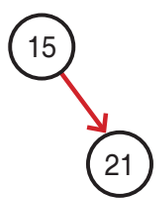
1. 二元搜尋樹中的所有節點皆儲存不同的數值，每一個節點數值必不相同。
2. 二元搜尋樹中，左子節點的數值必小於該所屬父節點的數值。
3. 二元搜尋樹中，右子節點的數值必大於該所屬父節點的數值。
4. 每一父節點所衍生的左、右子樹，皆符合二元搜尋樹的規則。

範例1-3-1　二元搜尋樹的建立

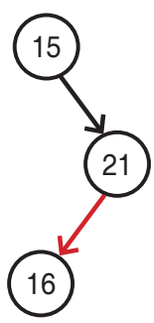
將右列數值資料建立二元搜尋樹：15、21、16、7、3、14、25、12
（二元搜尋樹的建立有其順序性，故採用帶有箭頭的邊連結節點。）

1.首先，放入資料15
作為樹根節點

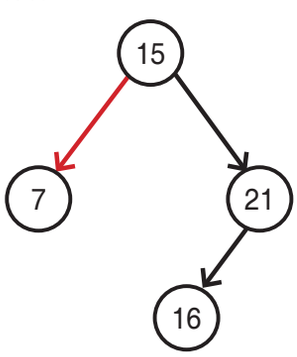
2.放入資料21
（21 > 15，往右子樹放）



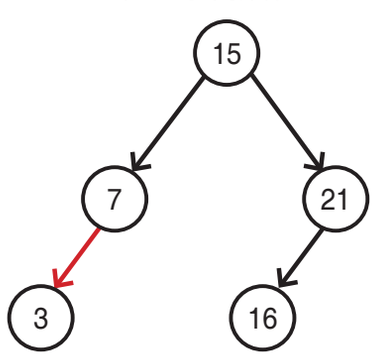
3.放入資料16
（16 > 15，往右子樹放；
16 < 21，往左子樹放）



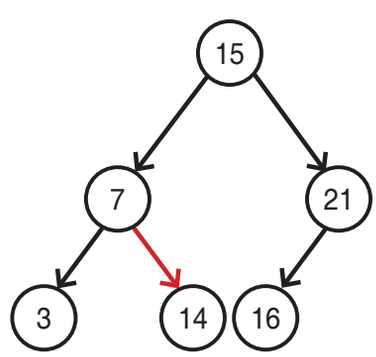
4.放入資料7
（7 < 15，往左子樹放）



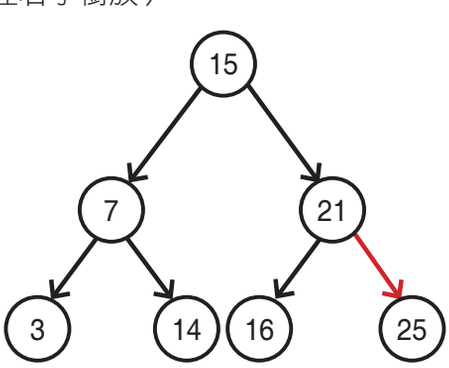
5.放入資料3
（3 < 15，往左子樹放；
3 < 7，往左子樹放）



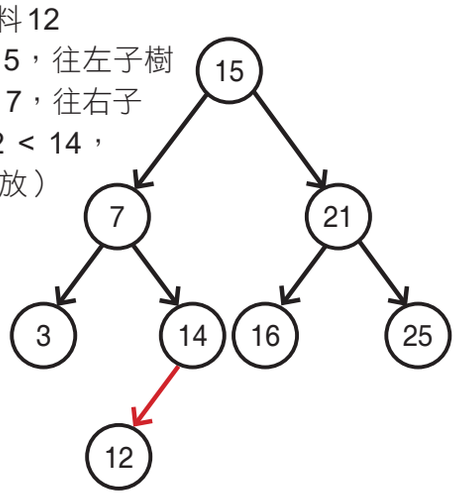
6.放入資料14
（14 < 15，往左子樹放；
14 > 7，往右子樹放）



7.放入資料25
（25 > 15，往右子樹放；25 > 21，
往右子樹放）



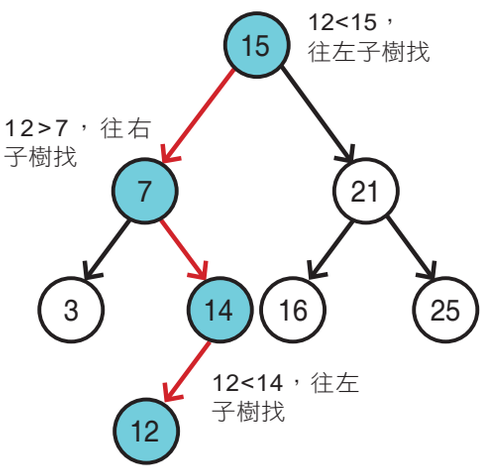
8.放入資料12
（12 < 15，往左子樹
放；12 > 7，往右子
樹放；12 < 14，
往左子樹放）



範例1-3-2 二元搜尋樹的搜尋

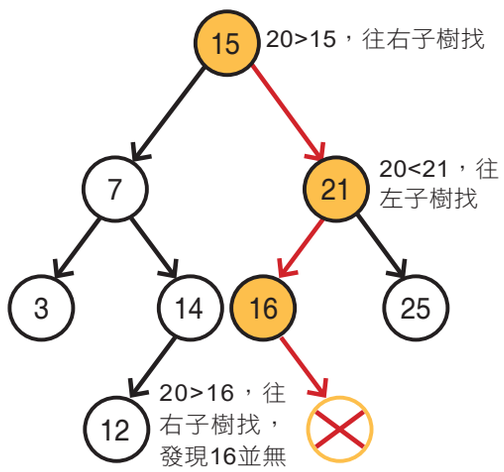
運用範例 1-3-1 建立之二元搜尋樹搜尋下列數值：

1.搜尋數值 12



12=12（經過第四次反覆比較）

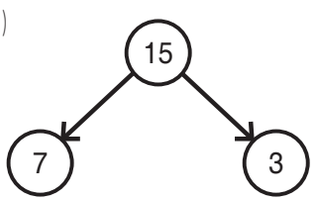
2.搜尋數值 20



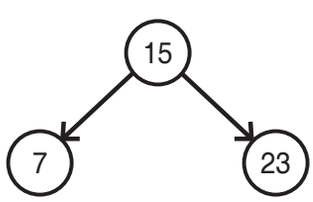
科技即戰力

請問下列圖形是否為二元搜尋樹？

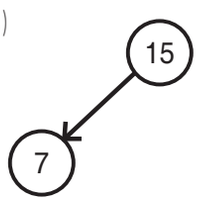
(1)



(2)



(3)



實作解析1-3-1

1.否。（右子樹 3 小於樹根 15）
2.是，符合二元搜尋樹條件。
3.是，右子樹可以為空集合。

霍夫曼編碼（Huffman Coding）

我們在國中階段學過的ASCII編碼，是屬於固定長度編碼方式；而霍夫曼編碼是屬於非固定長度編碼方式的一種，常用於字元（char）的編碼、資料的壓縮等用途，其運用資料出現頻率的差異性，進而降低資料儲存的空間。霍夫曼編碼運用字元出現頻率所產生的霍夫曼樹，即是二元樹的應用之一，其三步驟流程如下：

我們先將下列字串作為目標資料，並且以**ASCII碼**及**霍夫曼編碼**的方式比較兩者的差異性：

a a a a a a b b b b c c c c c d d e

1.以ASCII碼為編碼規則

字元	a	b	c	d	e
二進位表示	1100001	1100010	1100011	1100100	1100101

ASCII碼每個字元皆以7個位元（bits）表示，目標資料有6個a、4個b、5個c、2個d、1個e，共(6+4+5+2+1)=18個字元，故7(bits)×18(字元)=126位元

2.以霍夫曼編碼為規則

⑴先將目標資料的相異字元出現的頻率，由大到小表列。

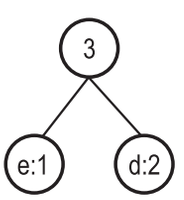
字元	a	c	b	d	e
出現頻率	6	5	4	2	1

⑵開始根據出現頻率列表，逐步建立霍夫曼樹。

STEP 1

挑選頻率最小的e、d作為左右節點，並加總頻率作為兩者樹根，且將此樹根列入頻率表排序。

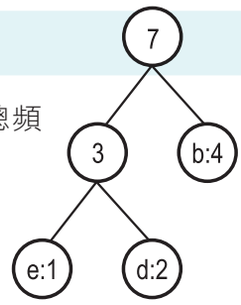
字元	a	c	b	
出現頻率	6	5	4	3



STEP 2

挑選頻率最小的3、b作為左右節點，並加總頻率作為兩者樹根，且將此樹根列入頻率表排序。

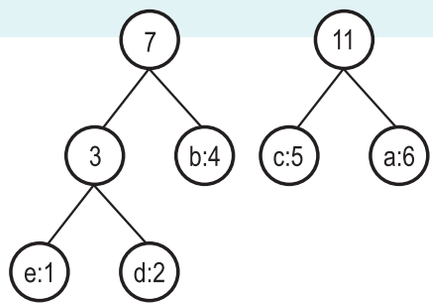
字元		a	c
出現頻率	7	6	5



STEP 3

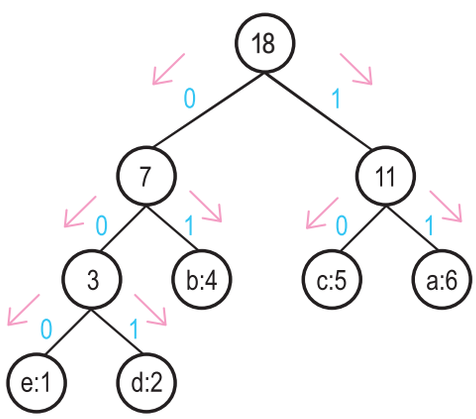
挑選頻率最小的c、a作為左右節點，加總頻率作為兩者樹根，並將此樹根列入頻率表排序。

字元		
出現頻率	11	7



STEP 4

將最後的7、11作為左右節點，加總頻率作為兩者樹根，並將所有通往左子節點的線段編上0，通往右子節點的線段編上1。



(3)再以建立的霍夫曼二元搜尋樹，制定新的編碼規則。

字元	a	c	b	d	e
編碼	11	10	01	001	000
出現頻率	6	5	4	2	1

共需2×6＋2×5＋2×4＋3×2＋3×1=39位元

科技即戰力

Q1.請比較上述例子中，分別以ASCII編碼與霍夫曼編碼兩種方式編碼後，何者較能節省儲存空間？

Q2.為何此種編碼方式較能節省儲存空間呢？

Q3.除了如JPEG圖片檔案格式是使用霍夫曼編碼外，還有哪些檔案格式也是運用這樣的技術呢？

二元樹的走訪

二元樹的走訪（或稱追蹤），其目的是為了造訪二元樹中的每一個節點，且依據根節點的造訪順序，共可分成三種走訪方式，分別如下所示：

01 前序走訪方式（Preorder Traversal）
依序先拜訪樹根節點，而後拜訪左子樹，最後拜訪右子樹，而每次拜訪的子樹又是依據相同追蹤方式，拜訪每一個節點。

02 中序走訪方式（Inorder Traversal）
依序先拜訪左子樹，而後拜訪樹根節點，最後拜訪右子樹，而每次拜訪的子樹又是依據相同追蹤方式，拜訪每一個節點。

03 後序走訪方式（Postorder Traversal）
依序先拜訪左子樹，而後拜訪右子樹，最後拜訪樹根節點，而每次拜訪的子樹又是依據相同追蹤方式，拜訪每一個節點。

中序走訪方式的結果須結合前序或後序走訪的結果才能還原二元樹，若只有前序或後序走訪結果則無法還原二元樹。

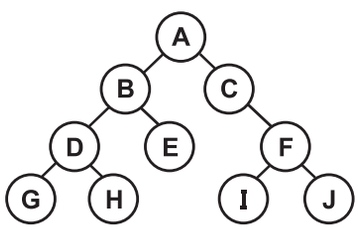


圖1-3-15 二元樹範例

以圖1-3-15為例，進行下列二元樹的走訪：

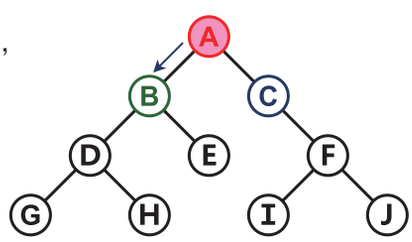
1.前序追蹤：由樹根節點優先輸出，而後輸出左子節點，再輸出右子節點。樹根的輸出順序在最前面，故稱為前序追蹤。

●：輸出值
○：樹根節點
○：左子節點
○：右子節點
○：已輸出節點

step 1

先輸出樹根 A，而後拜訪左子節點，最後再拜訪右子節點。

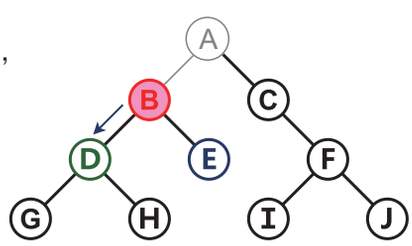
目前輸出 A



step 2

先輸出樹根 B，而後拜訪左子節點，最後再拜訪右子節點。

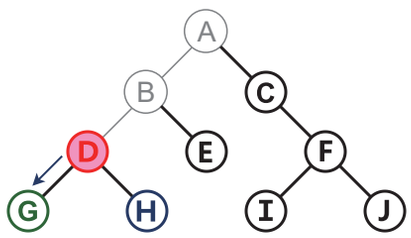
目前輸出 A → B



step 3

先輸出樹根 D，而後拜訪左子節點，最後再拜訪右子節點。

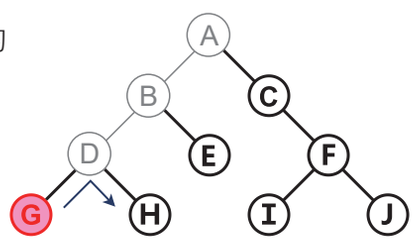
目前輸出 A → B → D



step 4

輸出 D 的左子節點 G。拜訪完 D 的左子樹後，再前往 D 的右子樹。

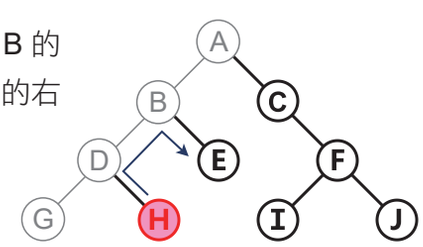
目前輸出 A → B → D → G



step 5

輸出 D 的右子節點 H。當拜訪完 B 的左子樹後，再返回上一層，進入 B 的右子樹。

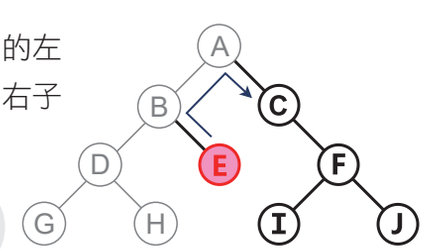
目前輸出 A → B → D → G → H



step 6

輸出 B 的右子節點 E。當拜訪完 A 的左子樹後，再返回上一層，進入 A 的右子樹。

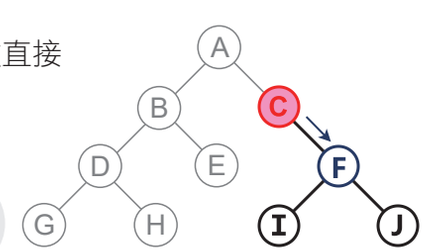
目前輸出 A → B → D → G → H → E



step 7

先輸出樹根 C，因 C 無左子樹，故直接拜訪 C 的右子樹節點。

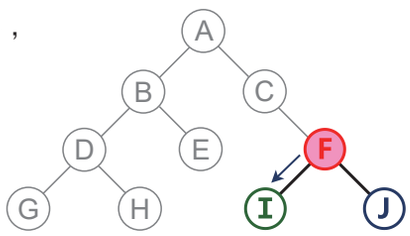
目前輸出 A → B → D → G → H → E → C



step 8

先輸出樹根 F，而後拜訪左子節點，最後再拜訪右子節點。

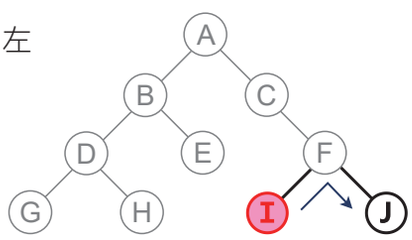
目前輸出 A → B → D → G → H → E → C → F



step 9

輸出 F 的左子節點 I。拜訪完 F 的左子樹後，再前往 F 的右子樹。

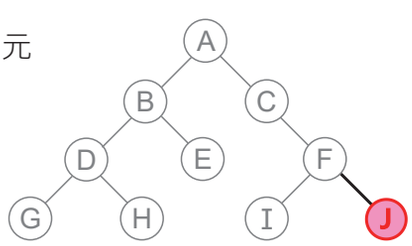
目前輸出 A → B → D → G → H → E → C → F → I



step 10

最後輸出 F 的右子節點 J，完成二元樹的前序走訪。

目前輸出 A → B → D → G → H → E → C → F → I → J

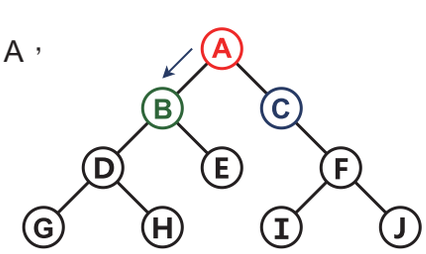


2.中序追蹤：由左子節點優先輸出，而後輸出樹根節點，再輸出右子節點。樹根輸出順序在中間，故稱為中序追蹤。

step 1

先拜訪 A 的左子樹，而後輸出樹根 A，最後再拜訪 A 的右子節點。

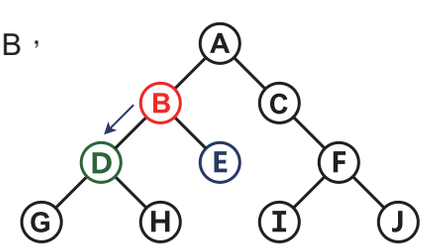
目前輸出 無



step 2

先拜訪 B 的左子樹，而後輸出樹根 B，最後再拜訪 B 的右子節點。

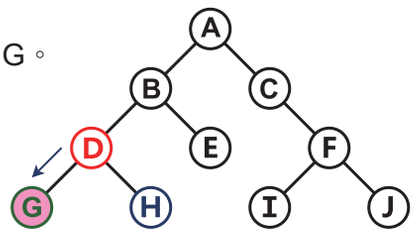
目前輸出 無



step 3

先拜訪 D 的左子樹，輸出左子節點 G。

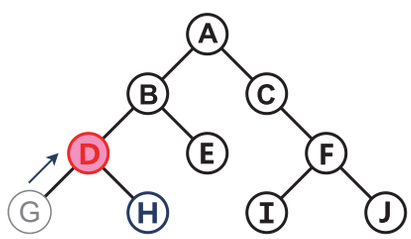
目前輸出 G



step 4

而後輸出樹根節點 D。

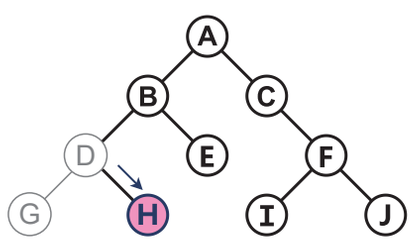
目前輸出 G → D



step 5

再拜訪 D 的右子樹，輸出右子節點 H。

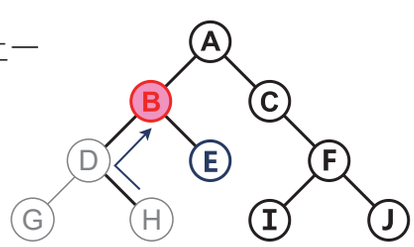
目前輸出 G → D → H



step 6

當拜訪完 B 的左子樹後，再返回上一層，輸出樹根節點 B。

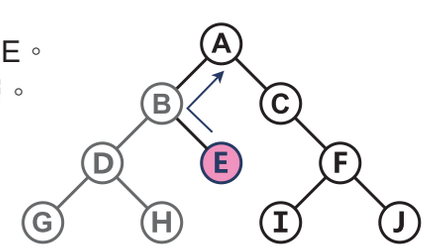
目前輸出 G → D → H → B



step 7

拜訪 B 的右子樹，輸出右子節點 E。
拜訪完 A 的左子樹後，返回上一層。

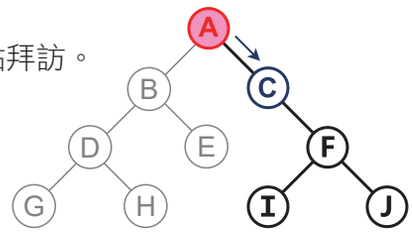
目前輸出 G → D → H → B → E



step 8

輸出樹根A，再進入A的右子樹節點拜訪。

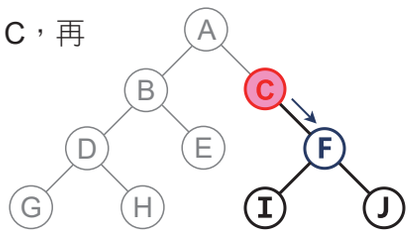
目前輸出 G → D → H → B → E → A



step 9

因 C 無左子樹，故先輸出樹根節點 C，再進入 C 的右子樹節點拜訪。

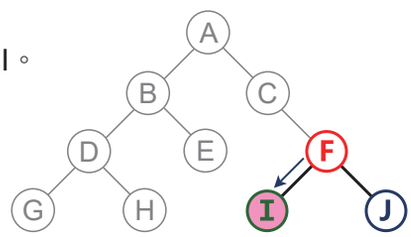
目前輸出 G → D → H → B → E → A → C



step 10

先拜訪 F 的左子樹，輸出左子節點 I。

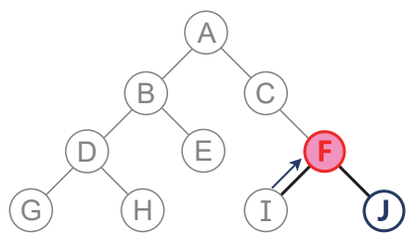
目前輸出 G → D → H → B → E → A → C → I



step 11

而後輸出樹根節點 F。

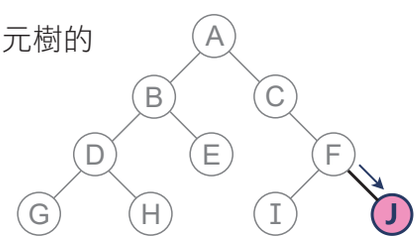
目前輸出 G → D → H → B → E → A → C → I → F



step 12

最後輸出 F 的右子節點 J，完成二元樹的中序走訪。

目前輸出 G → D → H → B → E → A → C → I → F → J

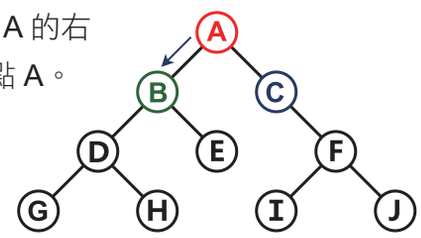


3.**後序追蹤**：由左子節點優先輸出，而後輸出右子節點，最後再輸出樹根節點。樹根輸出順序在最後，故稱為後序追蹤。

step 1

先拜訪 A 的左子節點，而後拜訪 A 的右子節點，最後再返回輸出樹根節點 A。

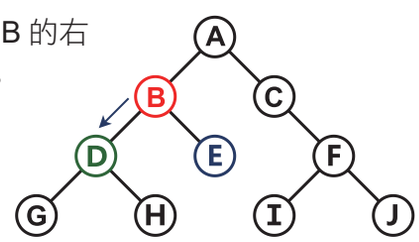
目前輸出 無



step 2

先拜訪 B 的左子節點，而後拜訪 B 的右子節點，最後再返回輸出樹根 B。

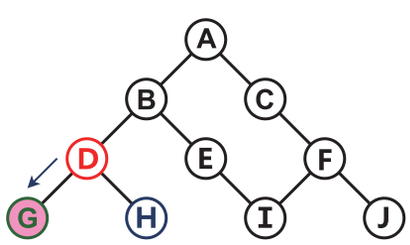
目前輸出 無



step 3

先輸出 D 的左子節點 G。

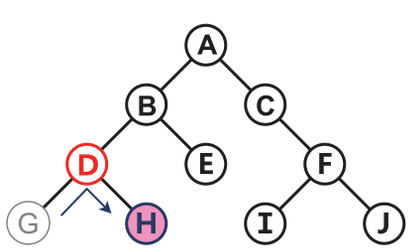
目前輸出 G



step 4

而後輸出 D 的右子節點 H。

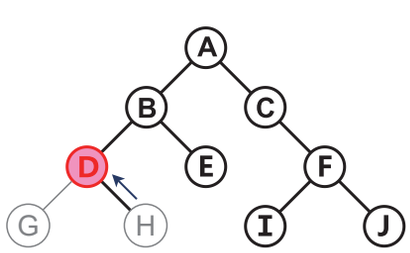
目前輸出 G → H



step 5

輸出樹根節點 D。

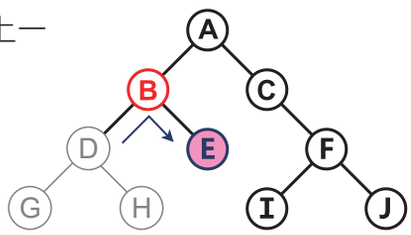
目前輸出 G → H → D



step 6

當拜訪完 B 的左子樹後，再返回上一層，輸出 B 的右子節點 E。

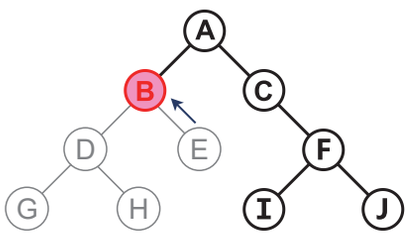
目前輸出 G → H → D → E



step 7

最後再輸出樹根節點 B。

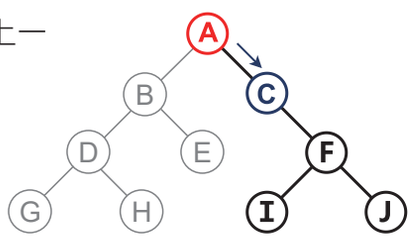
目前輸出 G → H → D → E → B



step 8

當拜訪完 A 的左子樹後，再返回上一層，拜訪 A 的右子節點。

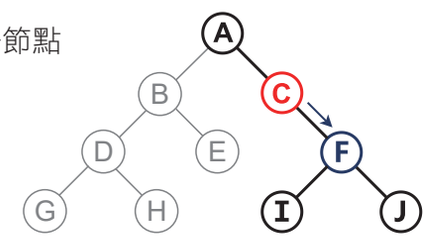
目前輸出 G → H → D → E → B



step 9

因 C 無左子樹，先進入 C 的右子節點拜訪。

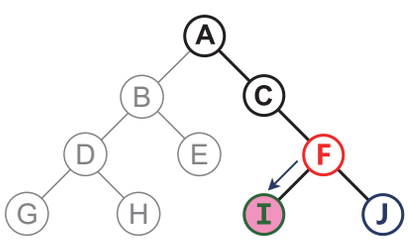
目前輸出 G → H → D → E → B



step 10

先輸出 F 的左子節點 I。

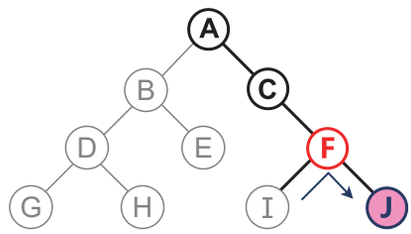
目前輸出 G → H → D → E → B → I



step 11

而後輸出 F 的右子節點 J。

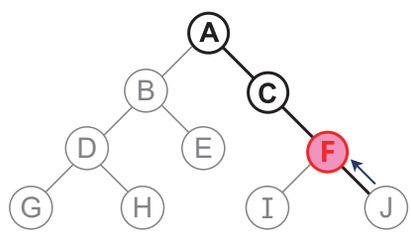
目前輸出 G→H→D→E→B→I→J



step 12

最後再輸出樹根節點 F。

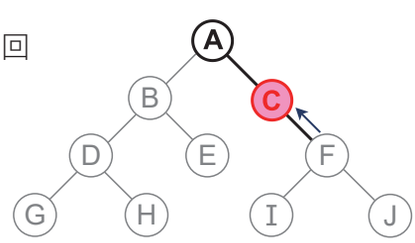
目前輸出 G→H→D→E→B→I→J→F



step 13

當拜訪完 C 的右子樹後，再返回上一層，輸出樹根節點 C。

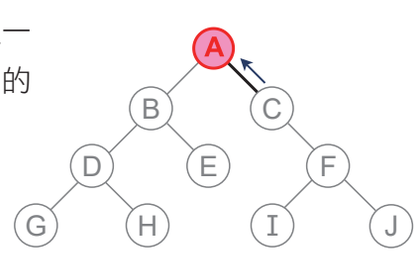
目前輸出 G→H→D→E→B→I→J→F→C



step 14

當拜訪完 A 的右子樹後，再返回上一層，輸出樹根節點 A，完成二元樹的後序走訪。

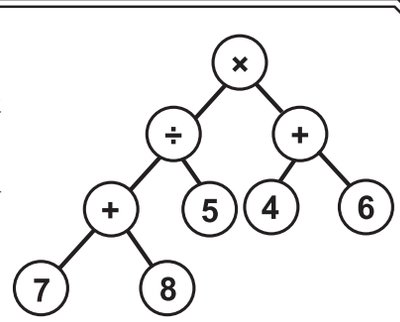
目前輸出 G→H→D→E→B→I→J→F→C→A



科技即戰力

請同學依據「中序追蹤」的規則，計算此圖走訪後的結果。

（將全部節點的數值與運算符號走訪完畢後，再將算式依四則運算規則計算結果。）



1-4 圖（Graph）

在資料結構中，**圖**是由「**頂點**（Vertex，或稱節點）」和「**邊**（Edge）」所構成，可分為**無向圖**和**有向圖**，兩者的差別在於邊的方向性，若邊不存在方向性即為無向圖（如圖1-4-1）；若邊存在方向性則為有向圖（如圖1-4-2）。

而邊也可加上距離、時間、成本、數量等數值作為**權重**（Weight）以表示相鄰兩節點的關係，如圖1-4-3中，有A、B、C、D四座村莊，各村莊之間架設數條自來水管線，「有向邊」跟「權重」就可以用來表示管線水流的方向與距離。

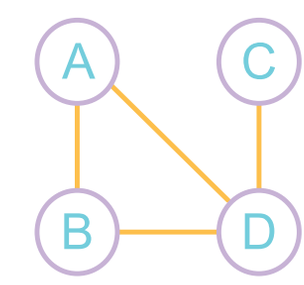


圖1-4-1 邊不存在方向性即為無向圖

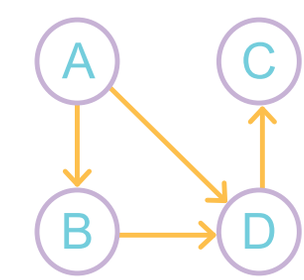


圖1-4-2 邊存在方向性則為有向圖

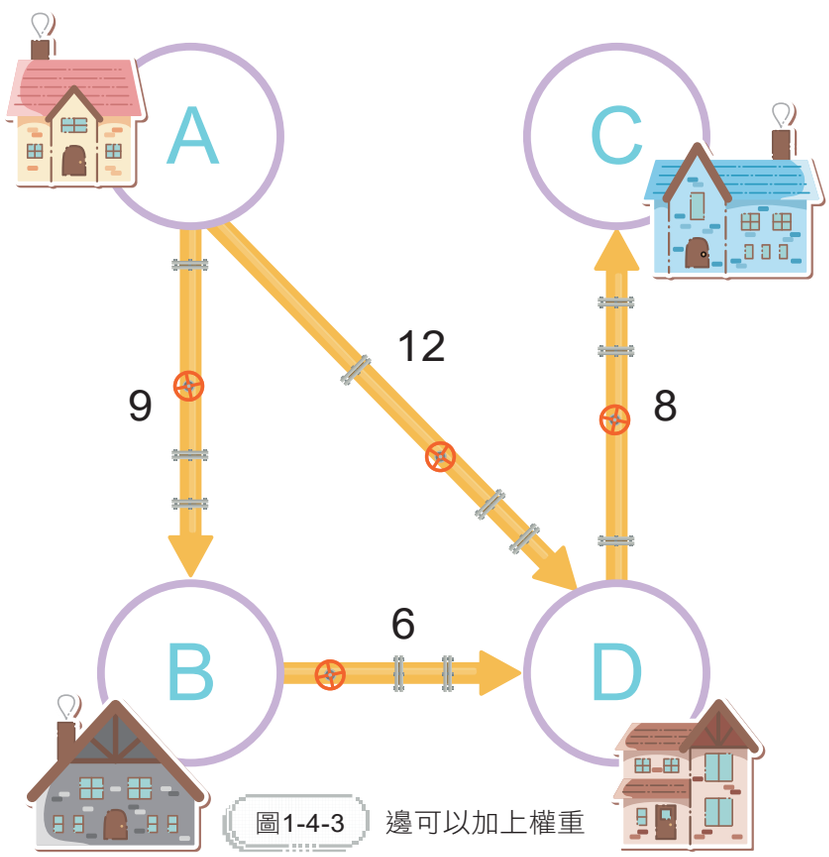


圖1-4-3 邊可以加上權重

可以透過下列二維陣列的方式，記錄與儲存無向圖和有向圖的資料。若有n個節點，則我們需要n×n的二維陣列記錄資料。

下一頁的表1-1為圖1-4-1的無向圖轉為二維陣列儲存，我們以左側的行（紅色區塊），作為節點的出發，上方的欄位（綠色區塊）的節點名稱作為終點，以A行來說，若存在有$\overline{AB}$、$\overline{AD}$，則在該列所相對應的欄位表格中，填入數值1；反之，若不存在有邊的關係，則填入數值0。

透過相同的方式，我們將上圖 1-4-2 和圖 1-4-3 的例子，分別轉為二維陣列的表 1-2 和表 1-3 儲存。透過表 1-1 與表 1-2 的比較，可以發現無向圖（表1-1）的數值在陣列中的45度軸線上呈現對稱，我們將其稱為**對稱矩陣**。而有向圖則因為邊存在方向性，並不一定會呈現對稱矩陣，如表 1-2 所示，只有$\overrightarrow{AB}$，並沒有$\overrightarrow{BA}$。

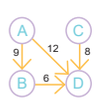
表 1-1 將圖1-4-1的無向圖轉為矩陣方式記錄

	A	B	C	D
A	**0**	1	0	1
B	1	**0**	0	1
C	0	0	**0**	1
D	1	1	1	**0**

表 1-2 將圖1-4-2的有向圖轉為矩陣方式記錄。如圖所示，只有 $\overrightarrow{AB}$，並沒有 $\overrightarrow{BA}$。

	A	B	C	D
A	**0**	1	0	1
B	0	**0**	0	1
C	0	0	**0**	0
D	0	0	1	**0**

表 1-3 將圖1-4-3的有向圖與權值轉為矩陣方式記錄



	A	B	C	D
A	**0**	9	0	12
B	0	**0**	0	6
C	0	0	**0**	0
D	0	0	8	**0**

許多導航軟體的路線規劃功能（圖1-4-4）也會透過在地圖的起點與終點之間，加入各種路線規劃，使得兩地（點）之間可以呈現不同路線（邊）的選擇，以及各自所需的時間（權重）。

圖1-4-4 Google Map的規劃路線功能，可以呈現兩端點（節點）間不同的路線（邊），及其所需時間和里程（權重）供使用者選擇。如圖中的路線A、B、C所呈現的資料。

我們也可以將捷運的站間時間以點和邊加上權重的方式，以圖來呈現，或是將圖轉化為矩陣方式（陣列）來表示或儲存。

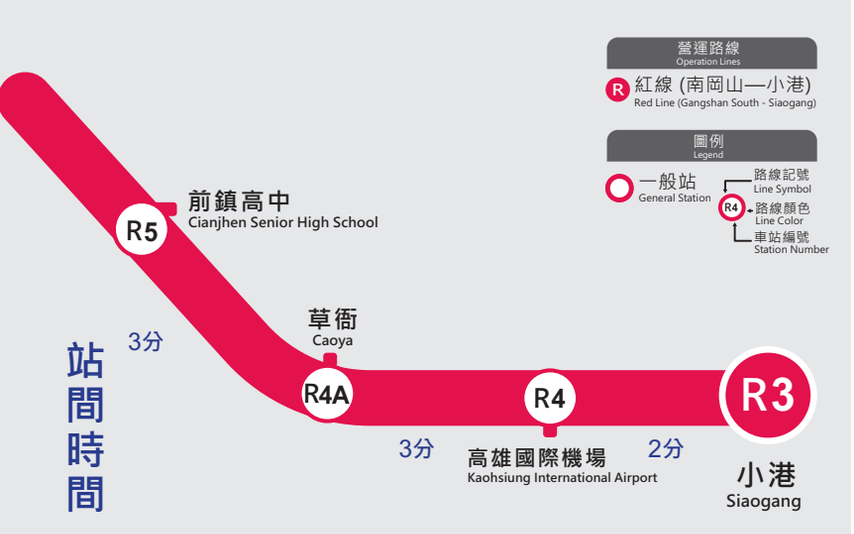


表 1-4 以矩陣表示站間的行駛時間

車站編號	R3	R4	R4A	R5
R3	0	2	5	8
R4	2	0	3	6
R4A	5	3	0	3
R5	8	6	3	0

圖1-4-5 站間的時間區間圖

IT 補給站

六度分隔理論（Six Degrees of Separation）

六度分隔理論又稱為「小世界現象」或「小世界效應」，其概念是由哈佛大學的社會心理學教授史丹利・米爾格蘭（Stanley Milgram，1933～1984）在1967年所提出，此理論是指世界上任何原先不相識的兩個人，只要透過少數朋友作為中介的連接橋梁，就可以互相認識對方。然而，隨著近年來社群網路的蓬勃發展，使得人與人的關係因為網路變得更加密集且錯綜（圖1-4-6），只要按下「加為朋友」的按鈕，即可串起彼此在虛擬世界的關係，Facebook更在其社群媒體上發現好友之間的關係平均只需要3.57人的距離，同學們是否覺得螢光幕上的偶像因此而離我們很近呢？

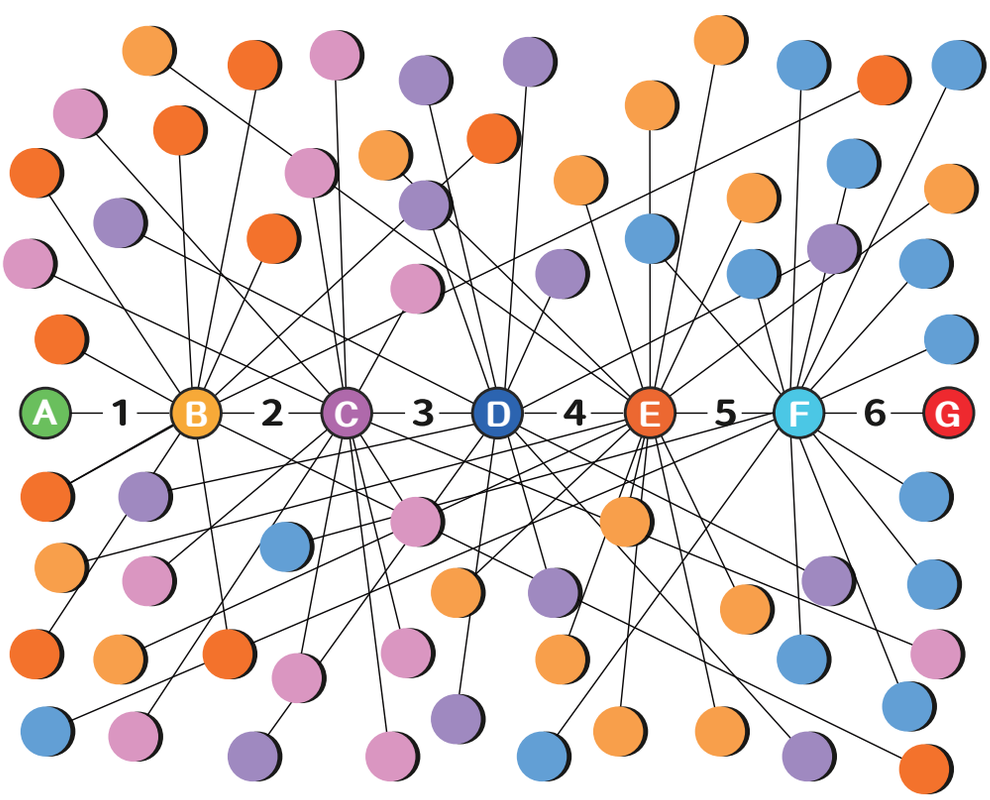
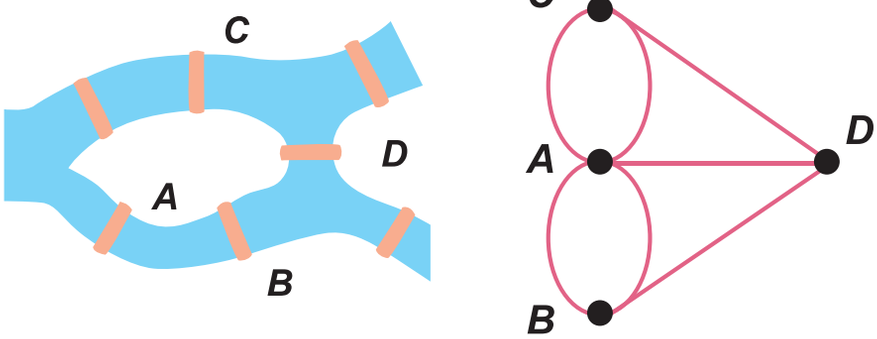


圖1-4-6 運用圖的點與邊的關係，呈現人（節點）與人（節點）之間的連結（邊）關係。如互不相識的A和G兩者之間的連結，存在有5位好友（節點B、C、D、E、F）、6個空間（邊）的距離，可呈現出六度分隔的概念。

補給站

圖論

圖論起源於俄羅斯柯尼斯堡：河流經過的市中心有兩座島，島與河岸總共有七座橋相連接。而著名的「七橋問題」問的就是：如何在所有的橋都只能經過一次的規則下，將七座橋全都走過一遍呢？



著名的瑞士數學家尤拉（Euler，1707～1783）找到了要在不重複走訪橋（邊）的前提下，將所有橋都走過一遍的兩項條件：

1.每個節點都有偶數個邊。（起點和終點必相同）

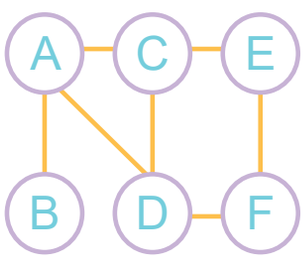
2.只存在兩個節點是有奇數個邊，其餘節點皆是偶數個邊。（起點和終點必不同）

只要滿足上述的任一項條件，就可以在不重複走訪每一條邊的前提下，造訪所有的邊。而七橋問題在這兩項條件上都不符合，因此「在不重複走訪每一條邊的前提下，造訪所有的橋」的可能性並不存在。

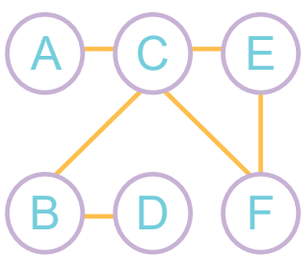
科技即戰力

試問下列圖形，哪些能以一筆畫（每一邊僅能走訪一次）的方式，走訪所有節點？

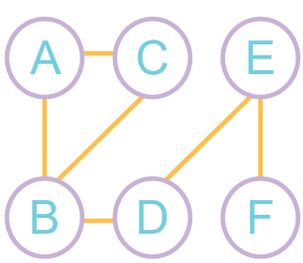
(A)



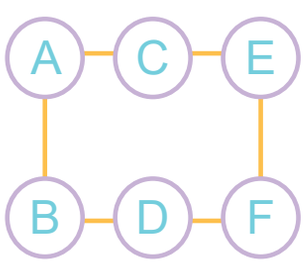
(B)



(C)



(D)



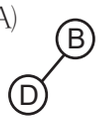
課堂檢測

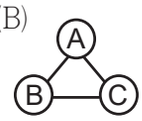
（　　）1. 請問瀏覽器中「上一頁」、「下一頁」的功能是來自下列何種資料結構的實作？
(A)樹 (B)堆疊 (C)佇列 (D)陣列。

（　　）2. 若有一個堆疊的資料結構，進行以下動作：push A、pop、push B、push C、pop、push D、pop，則最後堆疊內部結果為何？
(A)A (B)B (C)D (D)C。

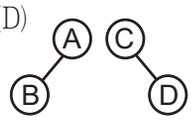
（　　）3. 請問印表機中的列印工作排程是使用下列哪一項資料結構？
(A)樹 (B)堆疊 (C)佇列 (D)陣列。

（　　）4. 有一佇列經過以下指令：加入 A、加入 B、刪除、加入 C、刪除、加入 D、刪除，則最後佇列的內部結果為何？
(A)A (B)B (C)C (D)D。

（　　）5. 下列何種資料結構是樹？
(A) 

(B) 

(C)

(D) 

。

（　　）6. 一個高度為5的二元樹最多會有幾個節點？
(A)16 (B)30 (C)31 (D)32。

（　　）7. 一個高度為5的二元樹最少會有幾個節點？
(A)3 (B)4 (C)5 (D)6。

（　　）8. 某比賽若採單淘汰賽制，取前3名，今有16支隊伍參賽，請問主辦單位需要規劃幾場賽事（包含季軍賽）？
(A)31 (B)16 (C)8 (D)32。

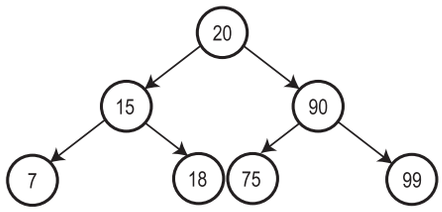
（　　）9. 若以霍夫曼編碼將下列字串ACMBABA進行編碼，請問至少需要幾個位元？
(A)11 (B)12 (C)13 (D)14。

（　　）10. 有一二元樹的前序走訪為FAEKCDHGB，中序走訪為EACKFHDBG，則其後序走訪為何？
(A)EAKCDBGHF (B)ECKAHBGDF
(C)FAKCEDGBH (D)FDGBHAKCE。

（　　）11. 將資料2、1、4、5、3依序插入空的二元搜尋樹，再以前序走訪，其結果為？（將資料較大者放入右子樹，較小者放入左子樹）
(A)21435 (B)12345 (C)21453 (D)54321。

（　　）12. 在二元樹的拜訪順序中，先探訪左子節點，再探訪右子節點，最後探訪父節點的方法是哪一種？
(A)循序法 (B)前序法 (C)中序法 (D)後序法。

( ) 13. 下圖是二元搜尋樹的資料結構，請問需要經過幾次的比較才能搜尋到數值18？

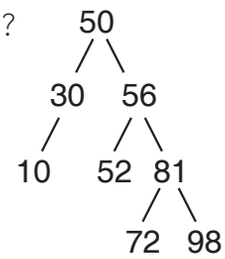


(A) 3 (B) 4 (C) 5 (D) 6。

( ) 14. 請問下列何種資料結構可以呈現作業系統的檔案架構？
(A) 堆疊 (B) 樹 (C) 佇列 (D) 圖。

( ) 15. 下列何者為霍夫曼編碼的實際應用？
(A) 壓縮資料 (B) 撰寫高階語言程式
(C) 加密資料 (D) 增加傳輸頻寬。

( ) 16. 對右列二元樹進行後序走訪時，走訪的次序為何？



(A) 10、30、52、72、98、81、56、50
(B) 50、30、56、10、52、81、72、98
(C) 50、30、10、56、52、81、72、98
(D) 10、30、50、52、56、72、81、98。

( ) 17. 下列何者不是圖的構成要素？
(A) 頂點 (B) 邊 (C) 節點 (D) 長度。

( ) 18. 若在二元搜尋樹中的每個節點存放正整數，則對此二元搜尋樹進行中序走訪，會得到下列何種數列結果？
(A) 數列由大到小排序
(B) 數列呈「前後端點較大，中間最小」排序
(C) 數列由小到大排序
(D) 數列呈「前後端點較小，中間最大」排序。

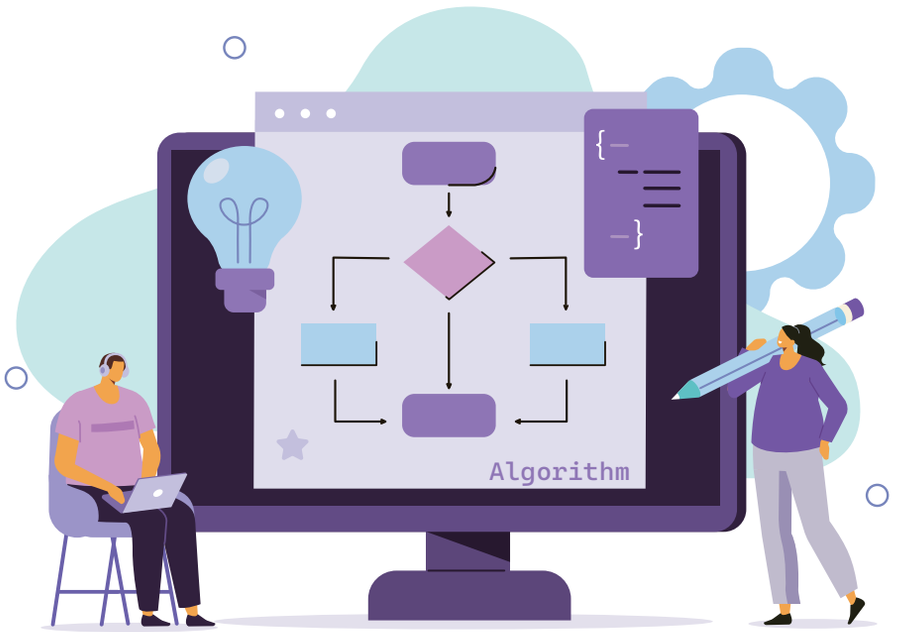
( ) 19. 請問下列日常生活中的應用，何者是先進先出（FIFO）的概念？
(A) 排隊的隊伍
(B) 瀏覽器的上一頁與下一頁的切換
(C) 電梯的進出
(D) 抽屜裡由上而下擺放的衣服。

( ) 20. 請問符合下列哪一項要素即可用一筆畫方式完成節點走訪？
(A) 每個節點都有奇數個邊
(B) 共有偶數個節點
(C) 每個節點都有偶數個邊
(D) 存在任三個節點是有奇數個邊。

chapter 02

重要演算法的概念與應用

我們常會針對特定的問題擬定策略後，規劃在有限的步驟內解決問題，例如：要計算 1+2+3+...+100，我們可以依序一個一個的將數字累加，或是使用其他更快的求解方式。而這些策略在資訊科技領域中，我們將其稱之為**演算法**（Algorithm），透過規劃出來的步驟，運用程式設計將其實現。



2-1 演算法簡介

演算法的定義為「解決特定問題的一連串有限指令或步驟」。在日常生活當中，我們可以看到很多「解決特定問題」的案例，如洗衣機的行程安排，先將衣物放入洗衣機並蓋上蓋板後，進行洗衣行程設定，按下啟動按鈕（圖2-1-1），洗衣機就會進入洗衣模式，執行多次反覆加入清水、洗淨、排水的過程，最後再進入脫水、烘乾等模式。

圖2-1-1 洗衣機的控制面板與啟動按鈕

上述洗衣機的例子就是一連串解決特定問題的有限步驟。但演算法並不是程式語言的一種，只是把解決問題的流程分析並記錄下來而已。

在程式實作前，系統分析師會針對問題提出需求分析，設計解決問題的演算法，再透過程式設計人員將演算法實作為程式碼，最後進行測試與修正。因此演算法可以決定程式設計的架構，軟體開發人員也能在開發過程中，透過演算法（運用文字式演算法、流程圖、虛擬碼等）彼此溝通，進而將軟體的各項功能進行**模組化**，分配給軟體開發團隊的成員。例如：通訊軟體（圖2-1-2）裡就有許多功能（聊天模組、主頁模組、貼文串模組、錢包模組等），將這些模組分別開發並組合後，就可以形成通訊軟體的APP（Application，應用程式）。

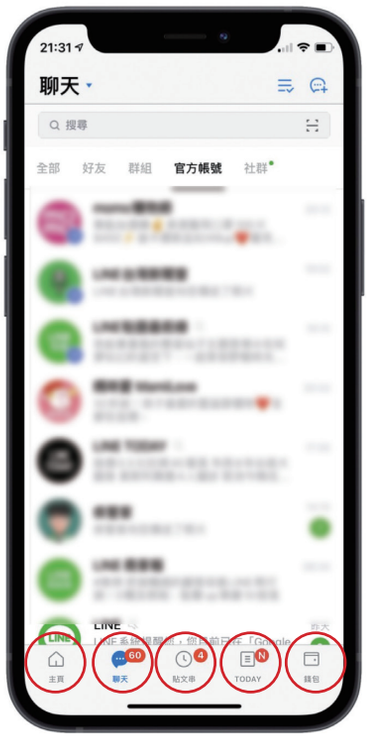


圖2-1-2 將通訊軟體APP中所需的各種功能模組化，再分別進行開發。

模組化：將大的系統架構劃分為功能各自獨立的小模組。
※ 詳見第 92 頁

演算法應具備下列五大特性：

1.輸入（Input）

0個或1個以上的輸入。有些演算法我們會先行設定預設值或初始值。

2.輸出（Output）

至少1個以上的輸出。

3.明確性（Definiteness）

每個步驟都必須明確，不可模稜兩可。例如：天氣「熱」就開冷氣，但「熱」的定義不夠明確，所以可以訂定「室內溫度大於或等於28度」時，即開啟冷氣機。

4.有效性（Effectiveness）

每個步驟都必須具體可行，也可用人工的方式算出相同的結果。

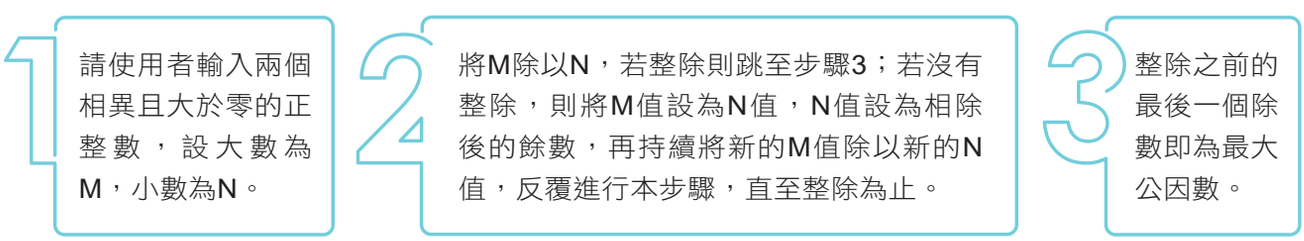
5.有限性（Finiteness）

必須在有限的步驟或指令內完成，不可以像是「無窮迴圈」一樣、無止盡地執行下去。

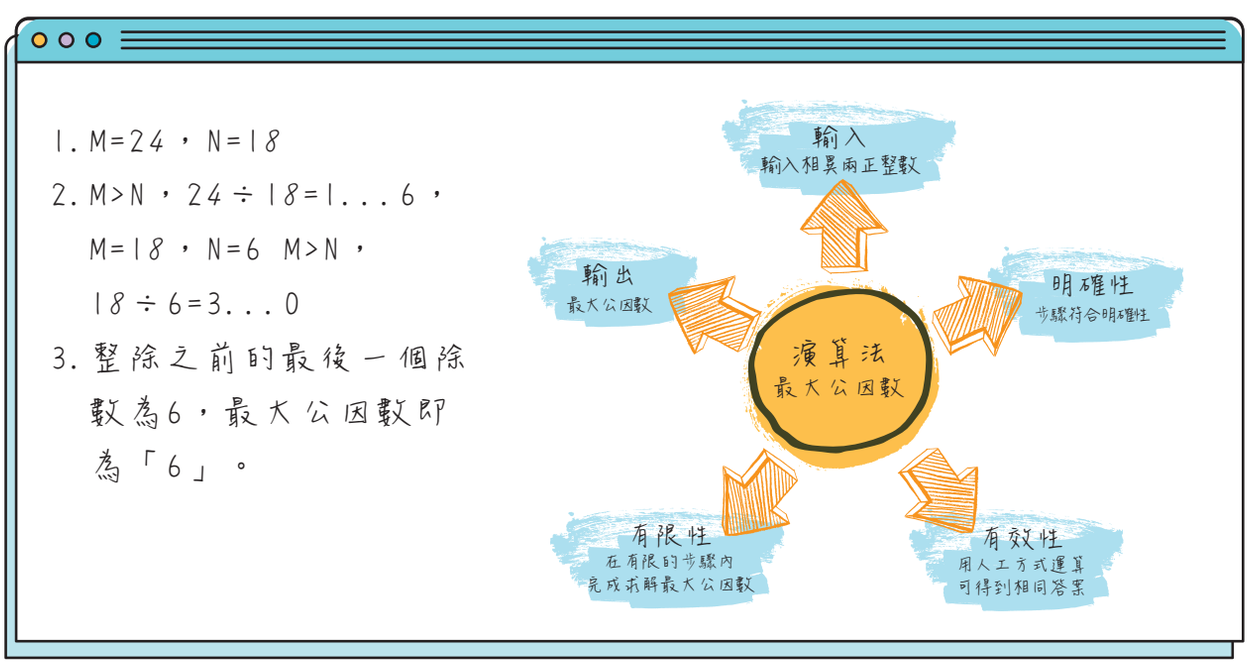
演算法可以透過**文字敘述**、**流程圖**（Flowchart）及**虛擬碼**（Pseudocode）表示。

其中，文字敘述是最容易撰寫的，但人類的語言往往有冗詞贅字，不同的敘述在不同的對象閱讀下，會因認知上的差異而產生不同的詮釋，因此我們應盡量避免選用文字敘述表示演算法。例如：要計算兩正整數的最大公因數（Greatest Common Divisor，GCD），我們可以用文字演算法，將解題步驟描述如下：

以上三個步驟運用文字說明演算法的流程，優點是運用口語化的描述方式呈現，多數人都可以明白計算過程，但缺點在於步驟2的「輾轉相除法」並未詳細說明，違反了演算法所需的「明確性」，因此我們將演算法流程修改如下：

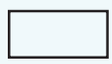
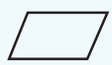


最後我們以人工運算並代入實際的數字，驗證演算法的有效性：



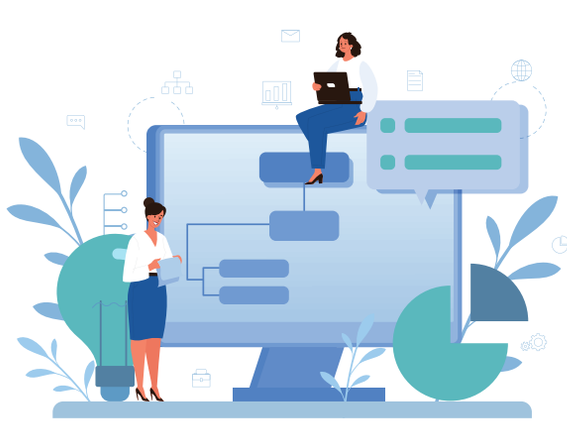
科技即戰力

Q1.請同學將輾轉相除法求最大公因數的例子運用流程圖的規則，繪製演算法。

流程圖符號	符號功能	符號說明
	流程線（箭頭）	表示流程與方向
	開始／結束	流程的開啟點或結束點
	程序（處理）	執行工作
	決策判斷（選擇）	依據判斷條件來選擇下一個階段的流程路線
	輸入／輸出	資料的輸入與輸出

Q2.同上題，運用虛擬碼的方式呈現演算法。

虛擬碼是一種介於程式語言和人類自然語言之間的表達方式，以程式語言為主要語法架構，自然語言僅做簡短敘述，保留自然語言高可讀性的優點，並可讓程式設計人員更有效率地將虛擬碼改為程式碼。



2-2 常見的解題策略與概念應用

2-2-1 暴力法（Brute Force）的解題策略

暴力法又稱**列舉法**、**窮舉法**，其運用直覺且簡單的解題策略和概念來解決問題。但若相同問題有其他解題的策略時，暴力法通常會是效率較低的解題方式。

暴力法是根據欲解決的問題，先列出所有可能的答案，並依據問題給定的條件，逐一檢驗每項答案是否能滿足該條件。如循序搜尋的演算法等。

科技即戰力

一顆蘋果32元、一顆西瓜150元、一顆芒果55元。若小芸身上有200元，請問有幾種買法呢？請用暴力法列出。（任一水果購買數量≥0，且剩餘的零錢不能再買任何一種為止。）

實作解析2-2-1

將每一種購買組合逐一列出（暴力法）如下表（應有蘋果小於等於6，西瓜小於等於1，芒果小於等於3的可能性）：

組合	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
蘋果	6	5	4	4	3	3	2	2	2	1	1	1	1	0	0	0	0
西瓜	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
芒果	0	0	0	1	0	1	2	0	1	0	1	2	3	0	1	2	3
餘額（元）	8	40	72	17	104	49	26	136	81	18	113	58	3	50	145	90	35

答案：組合A、D、G、J、M符合題目給定條件。

2-2-2 貪婪法（Greedy Method）的解題策略

貪婪演算法（Greedy Algorithm）又稱**貪心法**，是在每一步驟的選擇中，都採用對於當下最有利或最佳的選擇，以期能將最終結果導向到最有利或最佳解的解題策略。常見的問題如求最短路徑問題、最佳成本問題、背包問題、求霍夫曼編碼問題

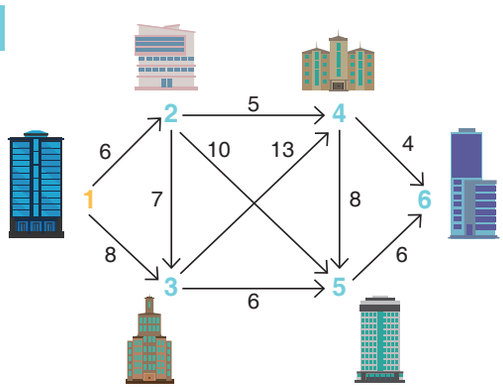
（Chapter 01曾經介紹過，其希望用最少的位元記錄頻率較高的字元）等，皆可嘗試透過貪婪法的解題策略求取最佳解。但貪婪法在每一次的選擇過程中僅專注於當下的最佳選擇，在逐步選擇的過程裡，並沒有考慮到各選項對於後續結果的影響，因此貪婪法在最終並不一定能獲得最佳解。

補給站

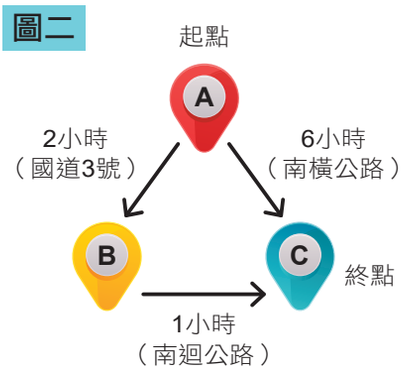
延伸活動——求某一點至各點之最短路徑（Shortest Path）長度

在第1-4小節提到的「圖」裡，我們曾在每一個邊上加上權重，這些權重可能是成本或是距離。而在多個節點與邊形成的權重網路中，我們將以下列的範例搭配**戴克斯特拉演算法**（Dijkstra's Algorithm），求得某一點至各點之最短路徑長度。

圖一



圖二



我們將上圖一的每個編號節點視為城市所在地，邊長的權重為兩城市間的距離，並試著從戴克斯特拉演算法的角度來思考，求出①號城市節點到各節點的最短路徑成本。（詳見下頁各步驟說明）

戴克斯特拉演算法

戴克斯特拉演算法可應用於帶有權重的「圖」，為其尋找某一點至各點的最短路徑。其算法由起始節點開始，依次計算並更新每個節點到起始節點的最短路徑，直至找到最終節點為止。如上圖二，計算A點至B點的最短距離之步驟如下：

步驟 \ 地點	A：臺南市	B：屏東縣	C：臺東市
STEP 1： 計算和A點相鄰的B、C點的時間	0	2	6
STEP 2： 計算和B點相鄰的C點的時間	0	2	2 + 1 < 6 自A點經由B點抵達C點的時間較A點直達C點短，故以2 + 1 = 3（小時）取代原來的6小時路程。

換你試試看（請參考前頁圖一）

STEP 1

A={2, 3, 4, 5, 6}，B={1}

從A集合中挑選起始節點1加入B集合，節點1到節點2的距離為6，從節點1到節點3的距離為8，節點4、5、6因未與節點1直接相連，先標示距離無限大。

節點	1	2	3	4	5	6
距離	0	6	8	∞	∞	∞

STEP 2

A={3, 4, 5, 6}，B={1, 2}

節點2目前與節點1距離最短，因此選擇節點2與相連的節點3、4、5重新計算最短距離，若最短距離比原先表格內小，則以較小路徑取代。

路徑1-2-3透過節點2為中繼點，則距離為6+7=13，比原先路徑8大，因此不更新表格內的值；路徑1-2-4的距離6+5＜∞（目前），因此更新表格內部的值；路徑1-2-5的距離6+10＜∞（目前），因此更新表格內部的值。

節點	1	2	3	4	5	6
距離	0	6	8	6+5=11	6+10=16	∞

STEP 3

A={4, 5, 6}，B={1, 2, 3}

節點3目前與節點1距離最短，因此選擇與節點3相連的節點4、5，重新計算最短距離，若最短距離比原先表格內小，則以較小路徑取代。

路徑1-3-4的距離8+13＞11（目前），故不更改；路徑1-3-5的距離8+6＜16（目前），因此更新表格內部的值。

節點	1	2	3	4	5	6
距離	0	6	8	11	8+6=14	∞

STEP 4

A={5,6}，B={1, 2, 3, 4}

節點4目前與節點1距離最短，因此選擇與節點4相連的節點5、6，重新計算最短距離，若最短距離比原先表格內小，則以較小路徑取代。

路徑1-4-5的距離11+8＞14（目前），故不更改；路徑1-4-6的距離11+4＜∞（目前），因此更新表格內部的值。

節點	1	2	3	4	5	6
距離	0	6	8	11	14	15

STEP 5 **A={6}，B={1, 2, 3, 4, 5}**

節點5目前與節點1距離最短，因此選擇與節點5相連的節點6，重新計算最短距離，若最短距離比原先表格內小，則以較小路徑取代。

路徑1-5-6的距離14+6>15（目前），故不更改。

節點6為終端節點，故直接輸出至集合B，A = { }，B = {1, 2, 3, 4, 5, 6}，並得到從1號節點到各節點的最短距離如下：

節點	1	2	3	4	5	6
距離	0	6	8	11	14	15

2-2-3 分而治之（Divide and Conquer）的解題策略與合併排序法（Merge Sort）

分而治之法又稱為**分治法**或**各個擊破法**，其概念是先將大而複雜的問題逐步**分解**（Divide）成兩個或更多的相同或相似的子問題，再將子問題不斷地分成更小的子問題，直到子問題化簡到可以用簡單的方式逐一**解決**（Conquer）並求得結果，最後再將子問題的結果合併以得到最終目標答案。切割後的問題規模愈小，愈容易直接求解，解題所需的計算時間也愈少。如合併排序、二分搜尋、河內塔等演算法。

分治法主要有三個步驟（圖2-2-1、圖2-2-2）：

圖2-2-1 分治法三步驟

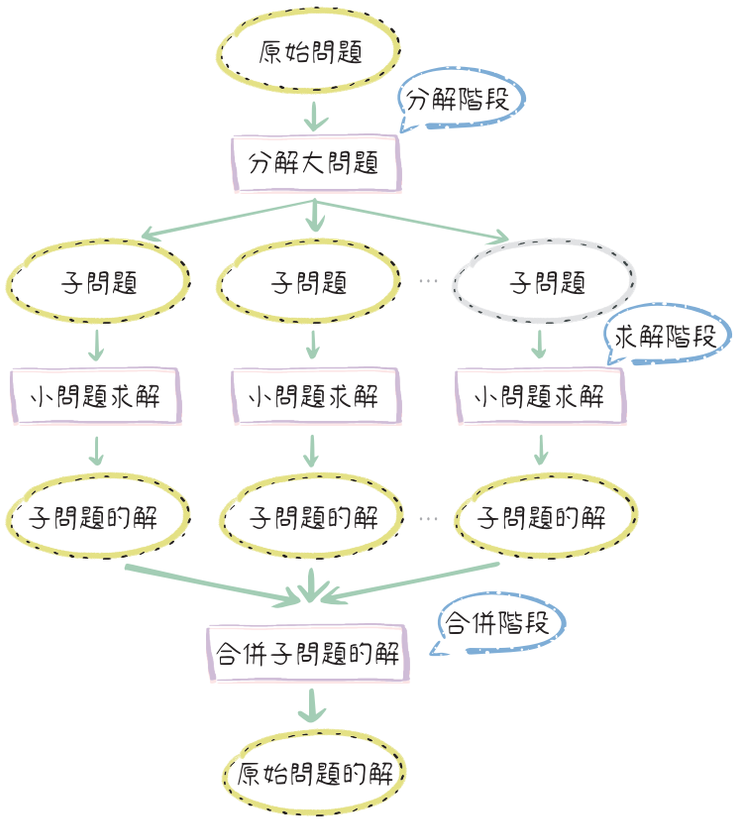


圖2-2-2 分治法的求解過程

合併排序法屬於上述分而治之演算法的一種應用，其演算法流程為先將原始陣列中的資料，逐步進行拆解，待分解到僅剩下一個元素時，再開始進行合併且排序的過程（圖2-2-3）。

圖2-2-3 合併排序法

科技即戰力

合併排序 1

請將數列{6，1，3，9，7，5，4，2}以合併排序演算法進行由小到大的排序，並將過程記錄下來。

實作解析2-2-2

【STEP 1. 分解】

1.將存放在陣列中的數列資料，先分解為兩筆陣列資料：{6，1，3，9}與{7，5，4，2}。

2.將{6，1，3，9}與{7，5，4，2}分別持續進行陣列分解，分別產生四個更小的陣列：{6，1}，{3，9}，{7，5}，{4，2}。

3.再將陣列{6，1}，{3，9}，{7，5}，{4，2}分別拆解至僅剩一個元素的陣列：{6}，{1}，{3}，{9}，{7}，{5}，{4}，{2}。

【STEP 2. 解決且合併】

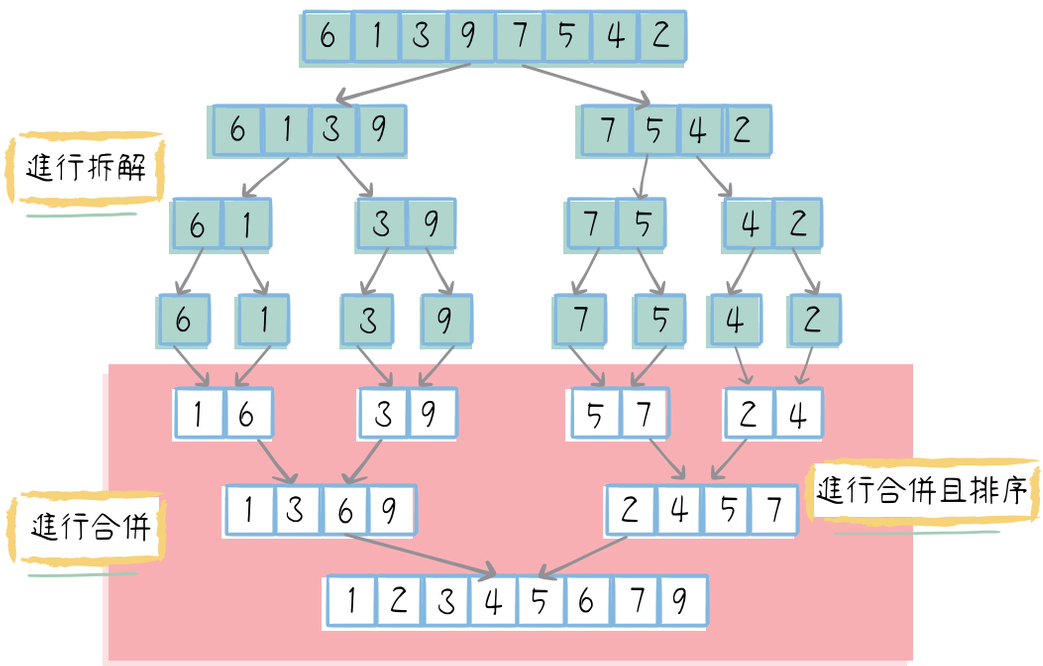
1.將僅存放一個元素的小陣列開始進行合併，且同時進行排序：{6}，{1}合併且排序為{1，6}；{3}，{9}合併且排序為{3，9}；{7}，{5}合併且排序為{5，7}；{4}，{2}合併且排序為{2，4}。

2.將上述合併後的陣列持續合併且排序：{1，6}，{3，9}合併且排序為{1，3，6，9}；{5，7}，{4，2}合併且排序為{2，4，5，7}。

3.最終將{1，3，6，9}與{2，4，5，7}合併且排序為{1，2，3，4，5，6，7，9}。

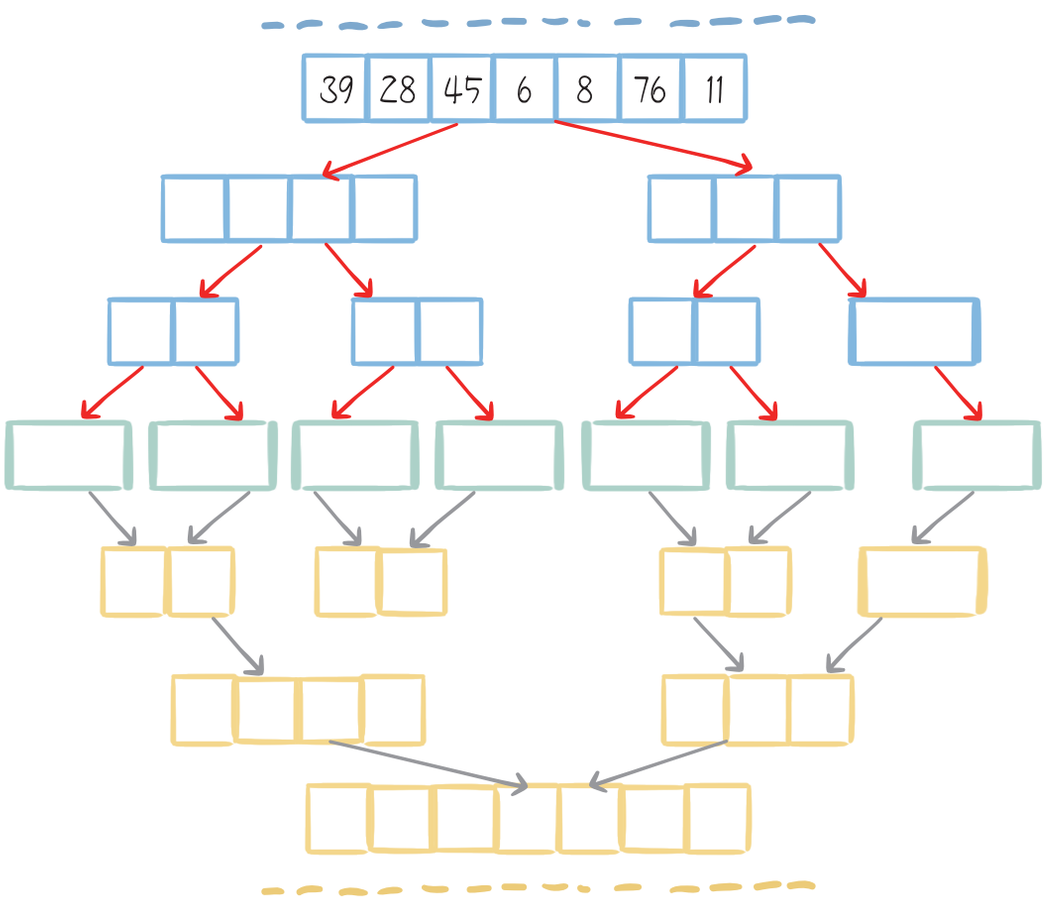
4.反覆進行STEP 2，直到所有陣列最終合併為一個大陣列，即完成所求。

（流程如下頁圖所示）



合併排序 2

請將下列的陣列資料以合併排序的方式逐步填入空格內：



2-3 遞迴結構的概念與應用

2-3-1 遞迴結構的概念

在生活中，可以看到許多關於**遞迴**（Recursion）的例子，如羅馬花椰菜（圖2-3-1）、蕨類的葉片（圖2-3-2）等，皆可以在其本身的局部結構中，發現與本身相似的結構，我們將其稱之為碎形。而遞迴結構的解題策略，是將一個大問題拆解成數個小問題，而這些小問題在每次拆解後能變得更簡單，且在問題不斷拆解後，最終將會由一個明確的終止條件對其進行收斂，並求得其結果。

圖2-3-1 羅馬花椰菜

圖2-3-2 蕨類葉片

當一個函式在解題過程中，藉由「不斷地自己呼叫自己」來進行解題，則表示這個函式具有遞迴的特性，如f(n) = f(n - 1) + 2，在f(n) 函式中透過呼叫 f(n-1) 函式本身，就是一種遞迴的解題過程。而為了避免函式無止盡地**自我呼叫**（Self-calling），必須給予函式一個明確的終止條件。因此，我們可以歸納出設計遞迴結構的兩項要素：

01 找出函式的一般項（函式通式）

透過一般項函式，使得函式能在解題過程中自己呼叫自己。

當中的通式可藉由觀察問題的特性後，尋找其中的規律性，例如：常見的有

累加 `f(n) = f(n-1) + n`；

累乘 `f(n) = f(n-1) × n` 等一般項函式。

02 結束呼叫的終止條件

設定最終收斂函式的結束條件。若沒有結束條件停止函式的自我呼叫，則此函式將永無止盡地執行，違背了先前提到的演算法「有限性」的特性。

將遞迴結構以虛擬碼呈現如下：

```
函式F(){
  if(條件式)                                    終止條件
              return 收斂結果;                  當終止條件成立，回傳結果
  else
              return 遞迴函式F();               當終止條件不成立，持續呼叫函式
}
```

```
函式F( ){
 if(條件式)
          return 收斂結果;
 else
          return 函式F( );
}
```

```
函式F( ){
 if(條件式)
          return 收斂結果;
 else
          return 函式F( );
}
```

```
函式F( ){
 if(條件式)
          return 收斂結果;
 else
          return 函式F( );
}
```

又好比數學中的階乘，亦為遞迴結構的概念之一：

科技即戰力

若 n 階乘的定義如下：

$n! = n \times (n-1)!$

$0! = 1$

試求5!=?

實作解析2-3-1

$$
\begin{aligned}
5! &= 5 \times 4! \\
&= 5 \times 4 \times 3! \\
&= 5 \times 4 \times 3 \times 2! \\
&= 5 \times 4 \times 3 \times 2 \times 1! \\
&= 5 \times 4 \times 3 \times 2 \times 1 \times 0! \\
&= 5 \times 4 \times 3 \times 2 \times 1 \times 1
\end{aligned}
$$

因此我們可以將階乘函式整理如下：

當 n > 0 時, f(n) = n × (n-1)!	階乘函式一般項的通式
當 n = 0 時, f(n) = 1	階乘函式的終止條件

2-3-2 遞迴的應用

費氏數列

費氏數列是13世紀的數學家費波那契（Leonardo Fibonacci，1175～1250）在一組數列中所發現的規律性，如「1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233……」，這群數列稱之為費氏數列。若將費氏數列中的數字逐項由小到大、依序作為圓半徑，並畫四分之一弧，且將其一一連接後，會形成一組螺線（圖2-3-3）。

自然界當中也可以觀察到許多生物具有費氏數列的特徵，如螺旋狀的貝殼、植物的花形與生長趨勢等（圖2-3-4）。

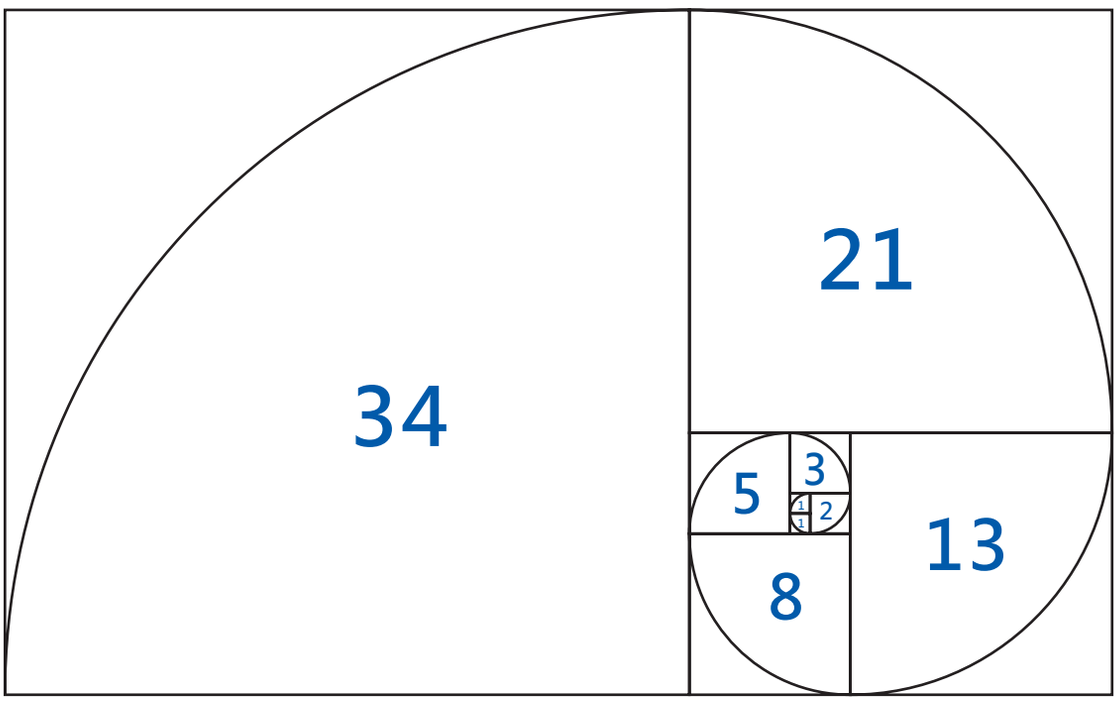


圖2-3-3 費氏數列所形成的螺線

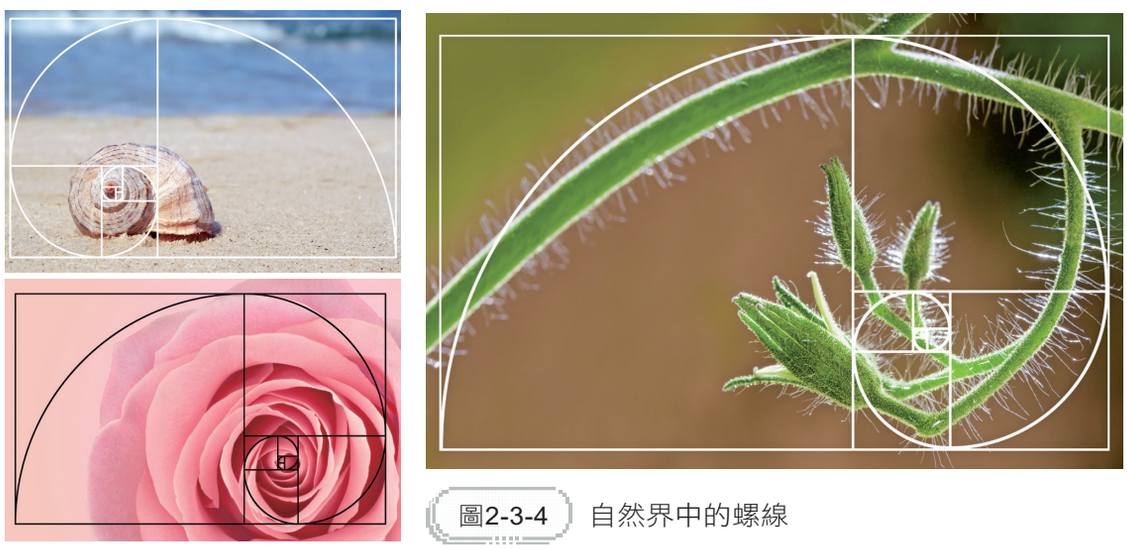


圖2-3-4 自然界中的螺線

人類也將費氏數列應用在美學上的黃金比率和建築結構，如希臘的帕德嫩神殿的屋頂與柱體（圖2-3-5）。

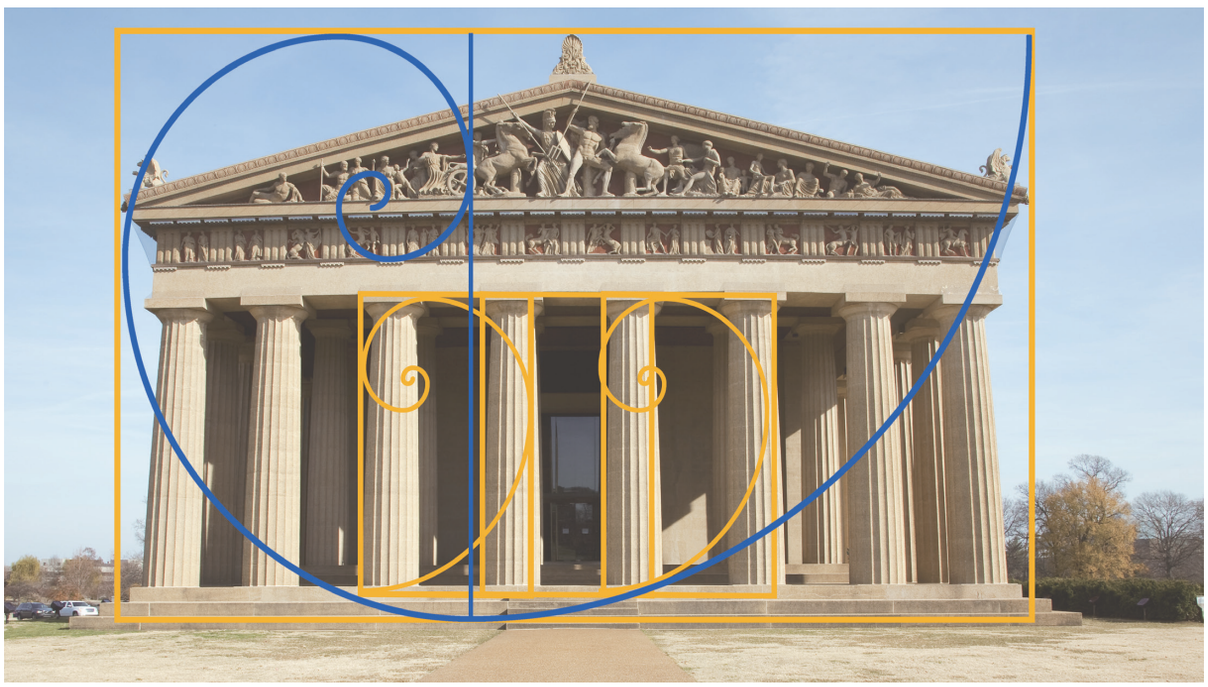


圖2-3-5 帕德嫩神殿的屋頂與柱體

費氏數列

定義 fib(0) = 0, fib(1) = 1,

fib(n) = fib(n-1) + fib(n - 2)

fib(2) = fib(0) + fib(1) = 0 + 1 = 1

fib(3) = fib(2) + fib(1) = 1 + 1 = 2

fib(4) = fib(3) + fib(2) = 2 + 1 = 3

另種費氏數列的首兩項定義為 fib(1) = 1 , fib(2) = 1

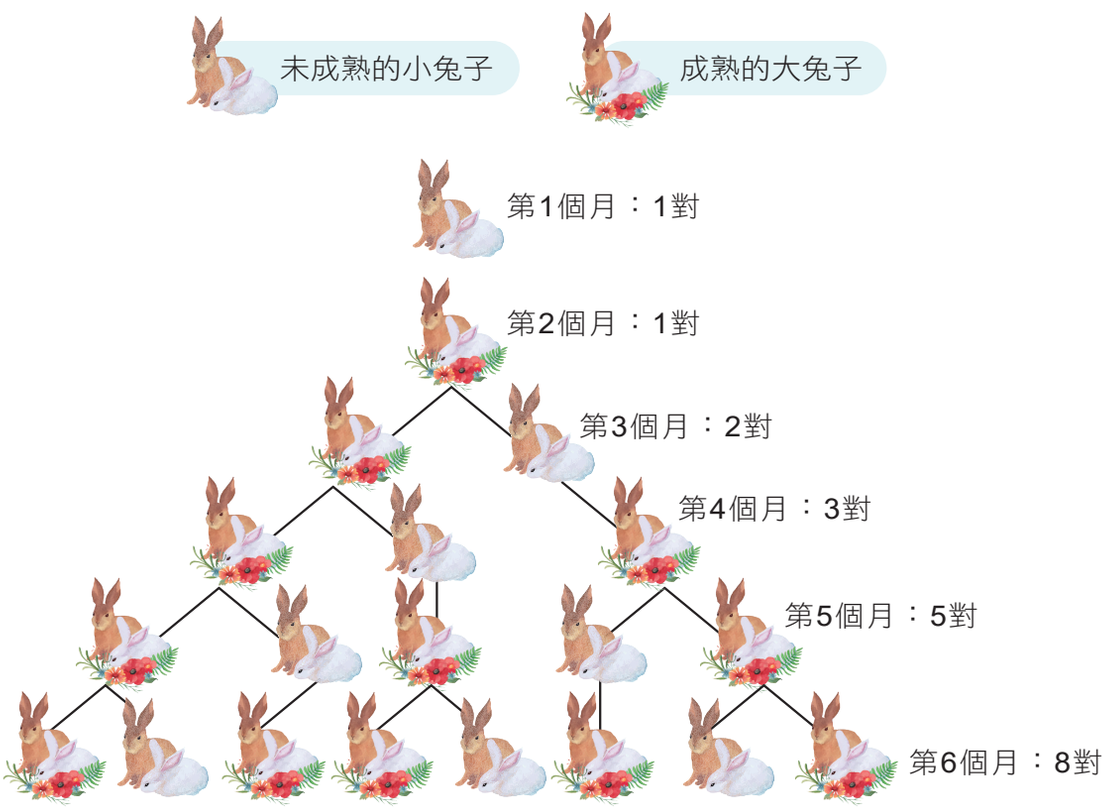
總結費氏數列的遞迴函式如下：

$$fib(n)=\begin{cases} 0, \text{當} n = 0\text{時} & \cdots\cdots \text{終止條件} \\ 1, \text{當} n = 1\text{時} & \cdots\cdots \text{終止條件} \\ fib(n-1) + fib(n-2), \text{當} n \geq 2\text{時} & \cdots\cdots \text{函式一般項通式} \end{cases}$$

科技即戰力

當有一對兔子在出生後滿兩個月時，便可繁衍出另一對兔子，之後的每個月可以繼續繁衍出新一對的兔子，而繁衍出來的兔子也遵循一樣的規則。
試問，若現在這一對兔子在籠子裡，第12個月之後，籠中共有幾對兔子呢？

實作解析2-3-2



運用費氏數列：

```
fib(1) = 1, fib(2) = 1,
fib(3) = fib(2) + fib(1) = 1 + 1 = 2
fib(4) = fib(3) + fib(2) = 2 + 1 = 3
fib(5) = fib(4) + fib(3) = 3 + 2 = 5
fib(6) = fib(5) + fib(4) = 5 + 3 = 8
fib(7) = fib(6) + fib(5) = 8 + 5 = 13
fib(8) = fib(7) + fib(6) = 13 + 8 = 21
fib(9) = fib(8) + fib(7) = 21 + 13 = 34
fib(10) = fib(9) + fib(8) = 34 + 21 = 55
fib(11) = fib(10) + fib(9) = 55 + 34 = 89
fib(12) = fib(11) + fib(10) = 89 + 55 = 144
```

→即為所求

補給站

以遞迴求兩數最大公因數

最大公因數的問題除了使用迴圈（重複執行相同敘述的結構）逐一查找相同的因數外，也可以利用輾轉相除法求解。輾轉相除法是運用反覆的相除，並逐漸收斂除數與被除數的大小，求得最大公因數，因此可以運用遞迴的方式實作輾轉相除法。

輾轉相除法可以歸納出兩個重點：

1.在整除之前的最後一個除數，即為最大公因數。（終止條件）

2.在尚未整除前，原來的除數為下一次函式運算的被除數，而原函式的餘數轉為下一次函式運算的除數。（遞迴函式）

我們定義輾轉相除法（以遞迴方式）的演算法為：

```
                ┌ return a, if b = 0              當b = 0時，傳回a值
GCD(a, b) = {
                └ return GCD(b, a%b), if b > 0    當b > 0時，傳回
                                                  GCD (b, a除以b的餘數)
```

例：

```
GCD(55, 22) = GCD(22, 11) = GCD(11, 0)  return 11
```

2	55	22	2
	44	22	
	11 ↑ GCD	0	

補給站

河內塔遊戲

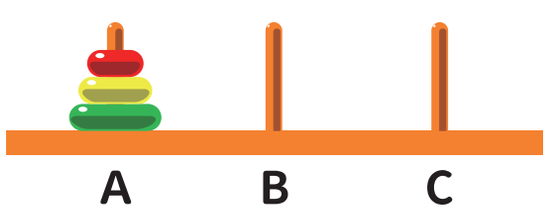
有3根木樁，分別為A木樁、B木樁、C木樁，遊戲一開始於A木樁放置n個盤子，移動過程中可透過B木樁作為暫時存放盤子的地方，最終目標是要將n個盤子全數移動到C木樁。

規則如下：

1.一次只能移動一個盤子。

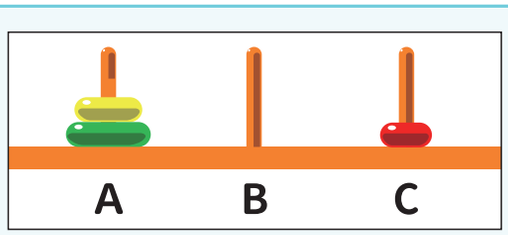
2.小的盤子必須疊在大盤子之上。

Q1 假如要依照規則、移動3個盤子，最少需要幾次步驟呢？

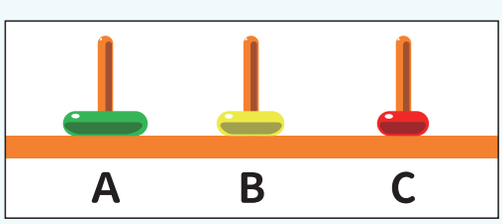


解題觀點：先將問題拆解，把最上層兩個盤子（紅、黃）視為一組移至B柱後，即可將綠色盤子移動到C柱。而後再將最上層兩個盤子（紅、黃）移至C柱。

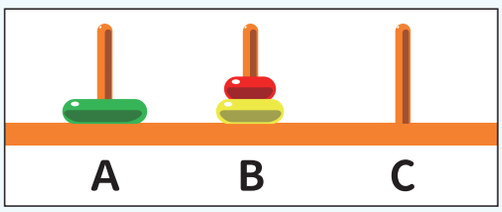
STEP 01 紅色盤子由A柱移動到C柱



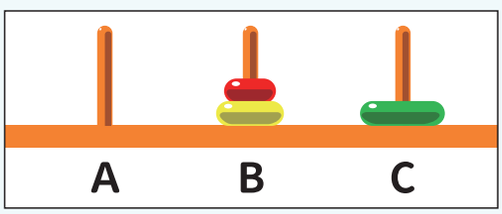
STEP 02 黃色盤子由A柱移動到B柱



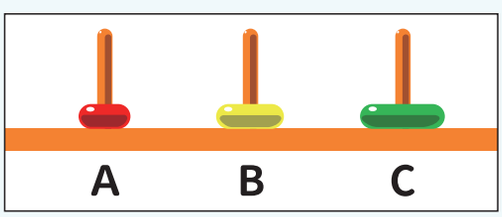
STEP 03 紅色盤子由C柱移動到B柱



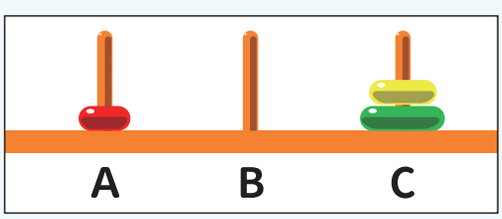
STEP 04 綠色盤子由A柱移動到C柱



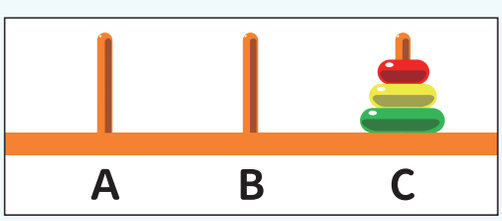
STEP 05 紅色盤子由B柱移動到A柱



STEP 06 黃色盤子由B柱移動到C柱



STEP 07 紅色盤子由A柱移動到C柱



A1 3個盤子，共需7次移動。

Q2 若有n個盤子，則至少需移動多少次呢？是否能找出共通的演算法規則呢？（盤子從1開始，由上而下編號至n）

問題解析（一）：

當n = 1時，將1號盤子從A柱移到C柱，1個步驟即可完成。

總共須移動2^1-1 = 1次。

當n = 2時，移動次序如下：

(1)移動1號盤子從A柱到B柱。

(2)移動2號盤子從A柱到C柱。

(3)移動1號盤子從B柱到C柱。

總共須移動2^2-1 = 3次。

而當n = 3時，總共須移動2^3-1 = 7次；當n = 4時，則須移動2^4-1 = 15次。

因此，我們可以歸納出一套規則：當有n個盤子時，共需2^n-1次的搬動。

也就是說，若有64個盤子需移動，則估算2^{64}-1次，大約需要1.84×10^{19}次的移動次數。如果每秒能搬動1次，大約仍需要5850億年！如果一部電腦每秒能運算一百萬次的移動，還是需要58.5萬年才能算完！

問題解析（二）：

3根木樁分別為A、B、C，盤子數量為n，設演算法函數為Hanoi(n，A，B，C)。當n為1時，表示只要再移動一次，也就是只要把盤子由A移到C就完成工作。可表示為：

```
當n = 1時，Hanoi ( 1, A, B, C )
```

當n不等於1時，進行3個動作：

(1)執行**Hanoi(n-1, A, C, B)**，將n-1個盤子由A柱透過C柱移動到B柱。

(2)執行將第n個盤子由A柱移動到C柱。

(3)執行**Hanoi(n-1, B, A, C)**，將暫存放在B柱的n-1個盤子透過A柱移動到C柱。

其中，每次移動後的n-1個盤子又會衍生新的子問題，再反覆藉由上述步驟執行完成。

由此可知，河內塔遊戲也可以應用遞迴的概念，求得最少次數的移動過程。

2-4 演算法效能分析

2-4-1 時間複雜度（Time Complexity）

前面提到演算法的定義是「解決特定問題的有限指令或步驟」，因此不侷限於某一種方法。但在眾多方法中，如何選擇最佳的方式或是評斷一個演算法的效能呢？像這種時候，我們可以透過分析**時間複雜度**來了解不同演算法的執行效能。

時間複雜度是用來衡量演算法執行效能的指標之一，我們通常用 Big O（英文大寫O）表示演算法的時間複雜度。

Big O 是用來分析演算法效率的數學符號，其代表該演算法時間的上限，表示在最差的狀況下，演算法的執行時間不會超過 Big O。

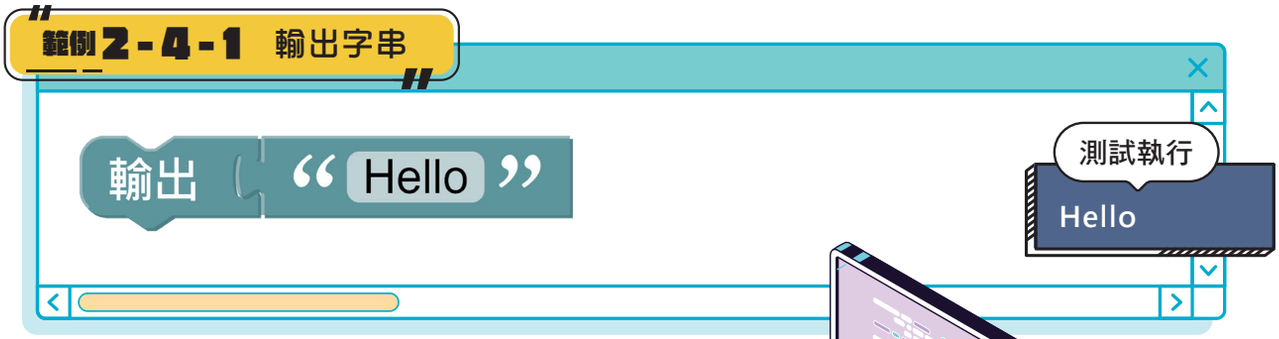
> 關於「Big O」，本節後續將會有更詳盡的說明。

演算法的執行次數

在本章節一開始介紹了撰寫程式的前置作業——演算法，接下來我們將運用 Blockly Demo線上開發平台(https://blockly-demo.appspot.com/)，探討程式執行次數與演算法的效能。

演算法的敘述包含一般的處理、儲存、輸入、輸出等，而每塊程式積木都是執行的步驟。讓我們運用下列三個例子，練習計算這些積木的執行次數。

範例 2-4-1 輸出字串

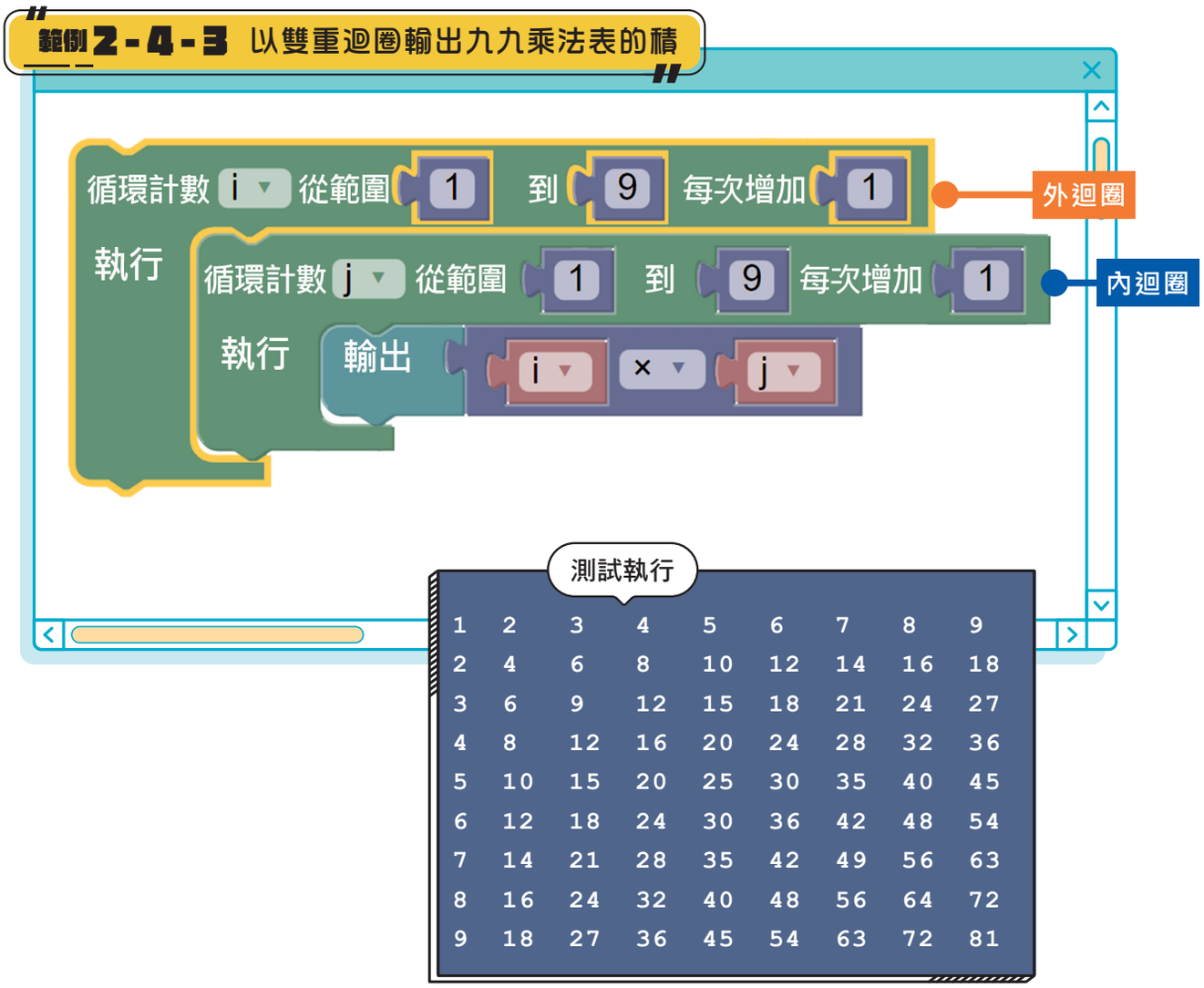


在螢幕上輸出並顯示字串 "Hello" 的這塊程式積木，記為執行 1 次。

範例2-4-2 計算2×1、2×2、2×3、……2×9的積，並依序輸出結果

在迴圈結構當中，有個負責輸出2×i的程式積木組合，其中的i由1到9依序帶入2×i，因此迴圈內負責輸出的程式積木組合共被執行9次。

範例2-4-3 以雙重迴圈輸出九九乘法表的積



雙重迴圈的外迴圈每進入一次i值的循環，則內迴圈負責輸出的程式積木組合i ×j會被執行9次（j由1到9）。因此有9次外迴圈的循環（i由1到9），則內迴圈負責輸出的程式積木組合i×j，共會被執行9×9＝81次。

執行時間常會由於當下使用的軟體或硬體的差異，而使相同的程式演算法得到不同時間結果。因此當我們想了解一個演算法的效能有多快，通常是以步驟次數來評估較為客觀。

舉例來說，在2022年時，美國超級電腦前沿（Frontier）是全球運算速度最快的超級電腦，每秒可以運算超過100京次（10^{18}）浮點運算（exaFLOPS），而台灣杉三號（Taiwania 3）是我國國家高速網路運算中心的超級電腦，整體效能可達每秒2.7千兆次（10^{15}）浮點運算（PFLOPS），兩者相差了約370倍之多。

而在不同的系統軟體執行，也會造成時間快慢的影響。相同的演算法指令在相異的硬體與軟體下，會產生巨大的差異性。若以時間秒數計算演算法的效能，會受外部因素的影響較大，因此用時間秒數來評估效能並不是客觀的方法。

執行次數與Big O值

執行的「步驟次數」通常會以演算法中，一段最主要的敘述決定其次數。

方才例子中，我們分別計算了兩個重覆結構（單一迴圈、雙重迴圈）的執行次數，但倘若一個演算法中包含多個迴圈，比如同時有單一迴圈、雙重迴圈等，則其執行次數應如何計算，並以Big O來表示呢？下表以常見的執行次數函式為例，並標示其時間複雜度與類型。

表 2-2 時間複雜度與類型

執行次數函式舉例	時間複雜度	類型
12	O(1)	常數時間
5log2n+20	O(log n)	對數時間
2n+3	O(n)	線性時間
2n+3nlog2n+19	O(n log n)	線性對數時間
$3n^2+2n+1$	$O(n^2)$	平方時間
$6n^3+2n^2+3n+4$	$O(n^3)$	立方時間
2^n+4	$O(2^n)$	指數時間
n!+2	O(n!)	階乘時間

時間複雜度比較：

$$O(1) < O(n \log n) < O(n) < O(n \log n) < O(n^2) < O(n^3) < O(2^n) < O(n!)$$

以表2-2中「平方時間」的執行次數$3n^2 + 2n + 1$為例，我們不會將其Big O記為$O(3n^2 + 2n + 1)$，而是會取其最高次方並忽略其係數以表示Big O，因此這個範例我們會記為$O(n^2)$。

在演算法中，時間複雜度是種表達其執行次數的概念，並以執行次數結合Big O描述該演算法的複雜度。運算過程中，當n值趨近無限大時，兩個演算法若以時間複雜度進行比較，會先以降冪排列後取其最高的項次作為代表，首項係數與其他項次因影響程度有限，故忽略不計。

其實我們也可以將演算法的時間複雜度Big O的表示法劃分為多個常見類型，將其簡化後能快速歸類出各種不同演算法的效能。時間複雜度的分類大致分為常數時間、對數時間、線性時間、線性對數時間、平方時間、立方時間、指數時間、階乘時間等。

常見的時間複雜度比較

在擬定一個解決問題的演算法時，其時間複雜度常以Big O評估並比較不同演算法執行的效能。例如：「排序」、「搜尋」等一些常見的演算法，其時間複雜度若是越能趨近於O(1)、O(log n)、O(n)，就越接近理想的情況。

根據以下的函數成長圖（圖2-4-1）所示，當演算法的時間複雜度為$O(n^2)$時，只要輸入的n數量一增加，演算法需要的時間就會大幅度地上升。

試想一下，當n = 10,000時，$O(n^2)$所需要的時間就會比O(n)多出一萬倍。尤其在現今的資訊科技領域，巨量資料（Big Data）的應用愈來愈受到關注，其資料的數量和種類遠超乎我們的想像，規劃一個良好的演算法將可大幅提升時間上的效能。

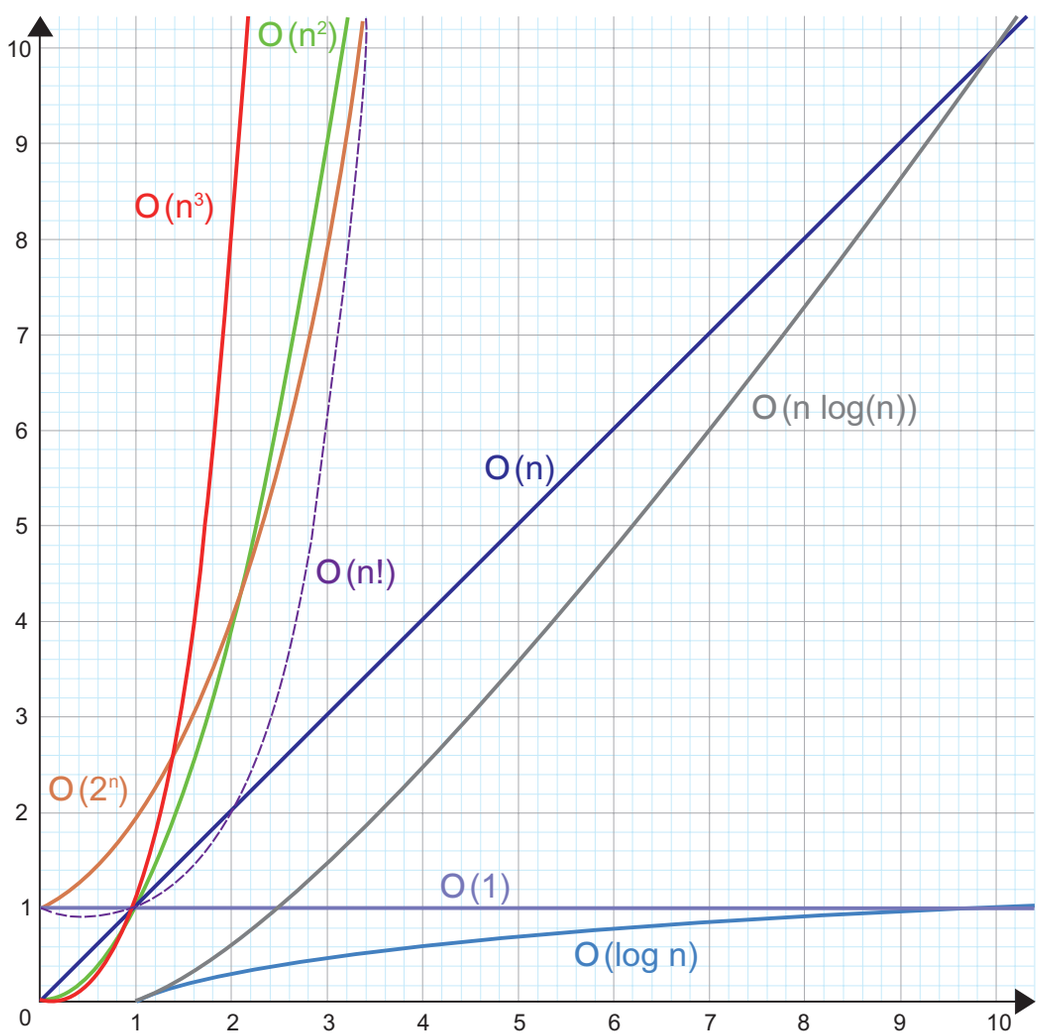


圖2-4-1 函數成長圖

課堂檢測

( ) 1. 下列何者非演算法的特性？
(A)明確性 (B)有效性
(C)保密性 (D)有限性。

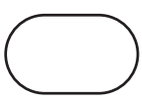
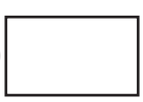
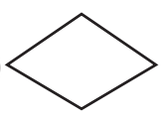
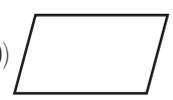
( ) 2.「每個步驟都必須明確，不可模稜兩可。」是屬於演算法的哪一個特性？
(A)明確性 (B)有效性
(C)條理性 (D)有限性。

( ) 3. 演算法至少要有幾個輸出？幾個輸入？
(A)0, 1 (B)1, 0 (C)0, 0 (D)1, 1。

( ) 4. 下列關於演算法的特性，何者有誤？
(A)每個步驟都必須明確，不可模稜兩可
(B)無法以人工的方式算出結果
(C)在有限的步驟或指令內完成
(D)至少要有1個以上的輸出。

( ) 5. 下列哪一種方法並非演算法的表示方式？
(A)流程圖 (B)虛擬碼 (C)文字敘述 (D)機器碼。

( ) 6. 下列何者為流程圖演算法中，決策判斷（選擇）的符號？

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

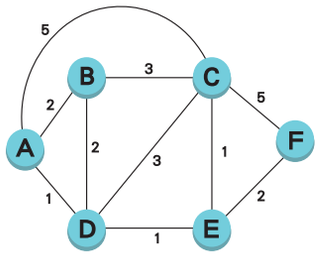
( ) 7. 演算法在執行過程中，透過函式自己呼叫自己，這種方法稱之為？
(A)遞迴 (B)堆疊 (C)佇列 (D)樹。

( ) 8. 若有一費氏數列，且定義fib(1)=0，fib(2)=1，試問fib(10)=？
(A)34 (B)55 (C)89 (D)56。

( ) 9. 河內塔問題中若有10個盤子，請問最少需要幾個步驟才能完成呢？
(A)1024 (B)1022 (C)1023 (D)1025。

( ) 10. 循序搜尋法是從第一筆資料開始，逐一將每筆資料取出並與目標值進行比對，直到在資料中找到目標值或所有資料都比對完才結束搜尋，試問此屬於哪一種類型的演算法？
(A)各個擊破法 (B)遞迴法 (C)貪婪法 (D)暴力法。

( ) 11. 下列哪一項演算法是指在某些條件的限制下，逐步選擇數種相異的選項組合，並期望此結果是在當下情況得到最佳解的方法，如求最短路徑、最佳成本、背包問題等？
(A)各個擊破法 (B)遞迴法 (C)貪婪法 (D)暴力法。



( ) 12. 考慮右圖標示有連結成本（Link Cost）之網路，請使用戴克斯特拉最短路徑演算法（Shortest Path Algorithm）來計算節點F至節點A之最短路徑長度（成本）？
(A)3 (B)4 (C)5 (D)6。

( ) 13. 承上題，使用戴克斯特拉最短路徑演算法來計算節點F至節點B之最短路徑長度（成本）？
(A)3 (B)4 (C)5 (D)6。

( ) 14. 承第12題，使用戴克斯特拉最短路徑演算法來計算節點F至節點C之最短路徑長度（成本）？
(A)3 (B)4 (C)5 (D)6。

( ) 15. 將大而複雜的問題逐步分解成兩個或更多的相同或相似的子問題，請問此屬於哪一種類型的演算法？
(A)各個擊破法 (B)遞迴法 (C)貪婪法 (D)暴力法。

( ) 16. 下列演算法的時間複雜度Big O為何？

```
for i from 1 to n:
    for j from 1 to n:
        get_meal( )
```

(A)O(1) (B)O(n)
(C)O(n log n) (D)(n^2)。

( ) 17. 下列演算法的時間複雜度Big O為何？

```
for i from 1 to n:
    get_meal( )
```

(A)O(1) (B)O(n) (C)O(n log n) (D)(n^2)。

( ) 18. 下列Big O的大小排列，何者正確？
(A) $O(n) < O(n!) < O(2^n)$
(B) $O(\log n) < O(n) < O(2^n) < O(n!)$
(C) $O(n \log n) < O(2^n) < O(n^{10})$
(D) $O(1) < O(n) < O(\log n) < O(n^2)$。

( ) 19. 以Big O代表某一個演算法的效率，則下列何者的時間複雜度Big O的方法效率最佳？
(A)$O(3n + n^2)$ (B)$O(n^2)$ (C)O(n log n) (D)$O(3n^2)$。

( ) 20. 若要將分而治之法（各個擊破法）的解題三步驟：「a.分解、b.解決、c.合併」按照順序來排列的話，下列何者正確？
(A)abc (B)bca (C)acb (D)cab。

主題二

程式設計

chapter 03
程式設計概念與實作　070

chapter 04
串列資料結構的程式設計
實作　122

chapter 05
重要演算法的程式設計
實作　142

int
CSS
Js
cin

chapter

03 程式設計概念與實作

在 Chapter 01 我們提到資料結構是程式設計的基礎之一，選擇適當的資料結構將會使得程式在執行上更有效能。而透過程式語言的實作，也可以將解決特定問題的演算法予以實現。

3-1 程式設計的概念

3-1-1 程式設計是什麼

早期的電腦又稱「計算機」，顧名思義是用來替我們解決數學運算方面的問題。而現今資訊科技的發展日新月異，各種新興科技產品不斷地推陳出新，例如：近年來智慧型手機蓬勃的發展，大大地改變了我們的生活。

然而，如果單單只有硬體的發展，無法促成新世代的科技創新，關鍵在於必須搭配各種軟體的發展。而軟體的開發必須高度仰賴人類的智慧，如世界知名的科技公司：微軟（Microsoft）、臉書（Meta）、谷歌（Google）等，無論是在作業系統、應用軟體或雲端應用等，都必須透過程式設計師團隊的分工合作，才有辦法完成。

因此無論是工業、交通、醫療、太空、軍事、商業、教育等等，各個領域都不斷地透過程式語言，開發相關的應用程式，進而也使得科技大幅改變了我們的生活樣貌。

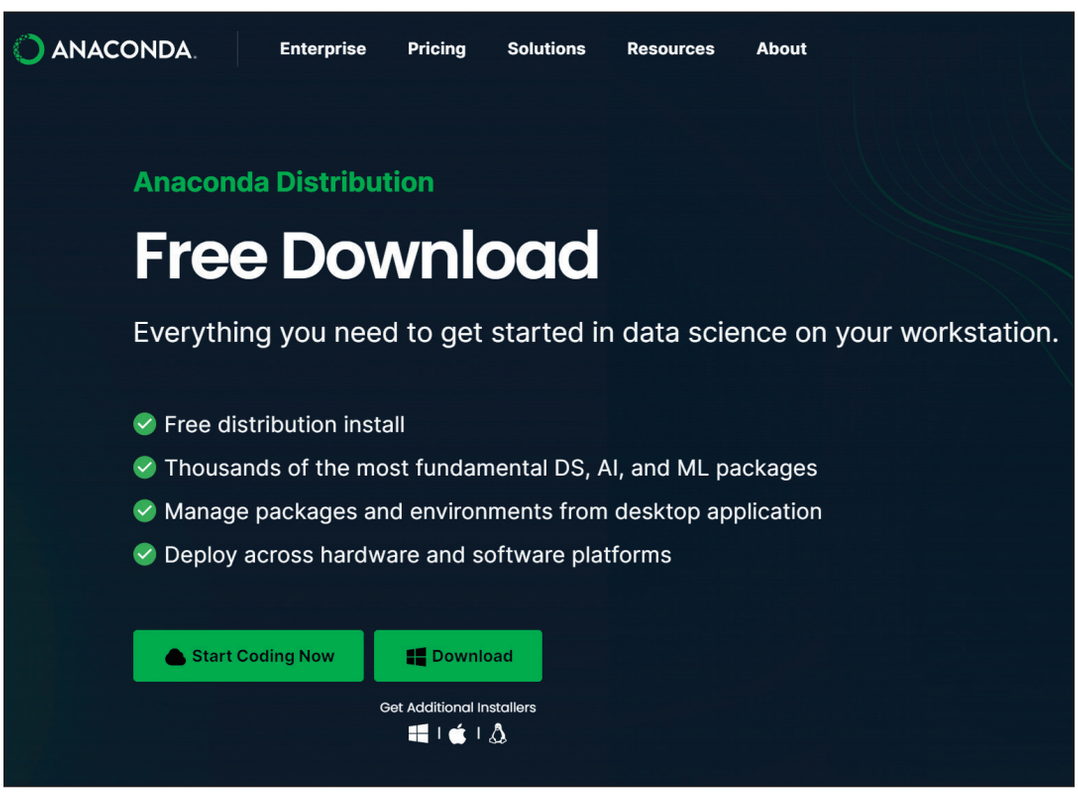
3-2 程式設計工具

3-2-1 Python與其開發環境

本書以Python作為程式語言實作的語法和範例，Python是一種高階、直譯式的程式語言，具有簡潔明瞭、易於學習的特點。其語法簡潔，並且擁有豐富的函式庫和第三方開發的函式庫，可以用於各種應用領域，如網路程式設計、巨量資料分析、人工智慧、科學計算等。

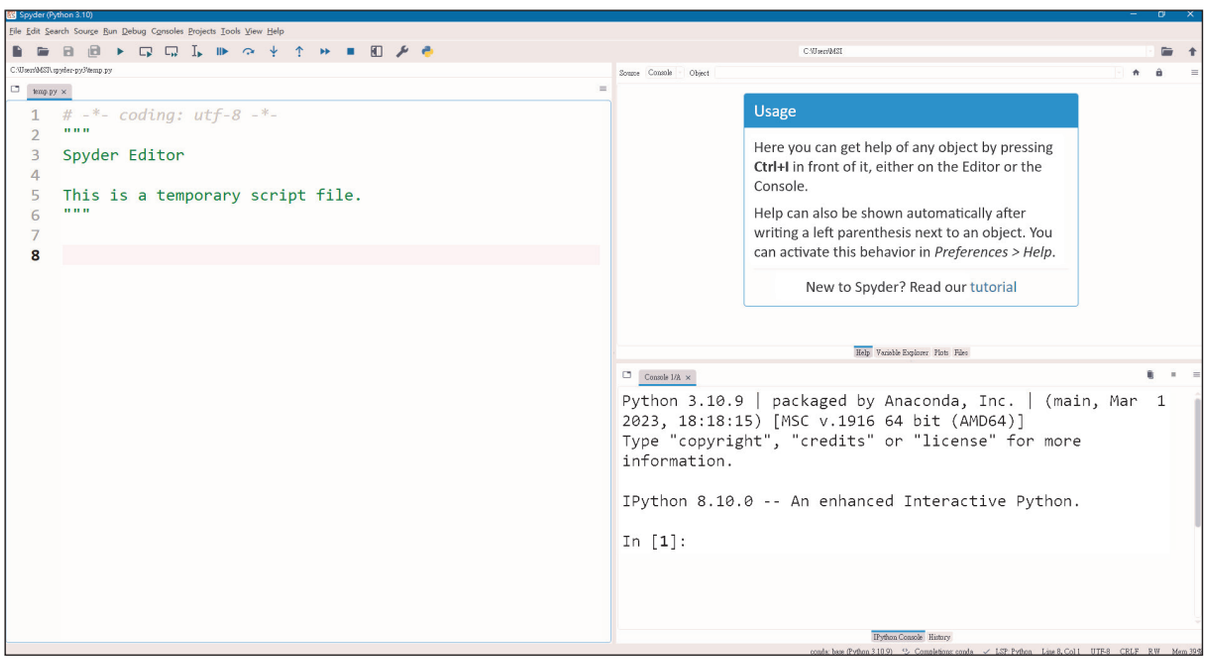
開發環境介紹

STEP 01 請於電腦開啟瀏覽器，鍵入網址：https://www.anaconda.com/download，下載 Python 開發軟體，並執行安裝程式。



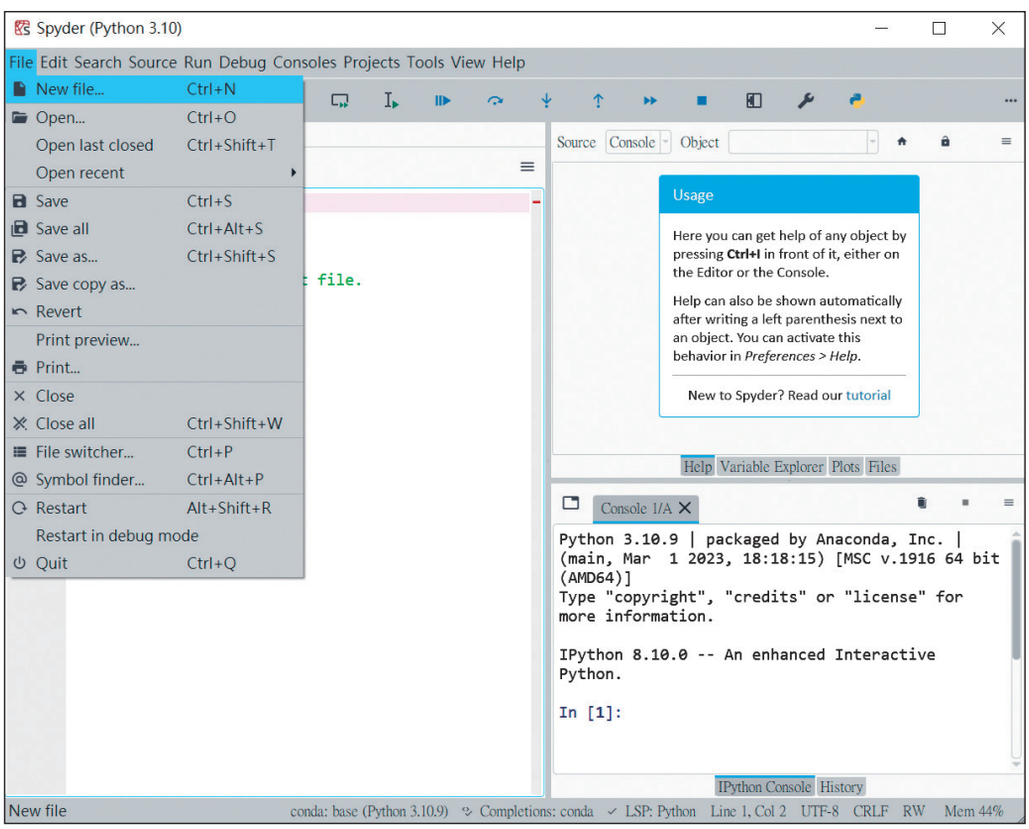
STEP 02

安裝後，請開啟Spyder程式，作為Python的整合開發環境（IDE）。



STEP 03

左側編輯視窗顯示的**temp.py**即為預設的Python編輯檔案。也可以從上方工具列**File→New file...**新增Python檔案。



3-2-2 實作第一個Python程式——Hello, Python!

請於Spyder程式中，輸入下方第8行程式碼：

```
1  # -*- coding: utf-8 -*-
2  """
3  Spyder Editor
4
5  This is a temporary script file.
6  """
7
8  print("Hello, Python!")
```

編輯後選擇File→Save as，並輸入自訂檔名後，再點選**存檔**。

於功能表列選擇Run→Run或按F5鍵，執行結果如下：

```
Python 3.10.9 | packaged by Anaconda, Inc. | (main, Mar  1 2023, 18:18:15) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 8.10.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/.../.spyder-py3/temp.py', wdir='C:/Users/.../.spyder-py3')
Hello, Python!
```

程式說明

1. <第1行>　*# -*- coding: utf-8 -*-*

#是程式的單行**註解**符號，程式的編譯器在執行時會自動忽略後面的文字，因此註解通常是給編輯人員看的。而在這一個程式中的第一行**coding: utf-8**指的是此程式宣告所採用的編碼規則為utf-8編碼系統，通常會寫在程式的前兩行，但在Python3以上的版本預設就是採用utf-8編碼系統，故亦可省略。當中的-*-僅作為醒目標示，無任何意義。

例如：

```
#這是我的第一個程式
```

善用註解是程式設計師的良好習慣之一，也可以幫助往後程式的除錯或是專案團隊的合作。雖然註解對程式本身的執行效率等毫無影響，但可以幫助程式設計人員了解程式的運作。

2. <第2行～第6行> """

三個引號所涵蓋的範圍為**多行註解**。

例如：

```
"""
...此為多行註解...
可以說明程式的功能或用途
"""
```

3. <第3行> `print("Hello, Python!")`

print為輸出指令，將所要輸出的內容放於( )內，而此處輸出字串則可以使用雙引號（" "）或單引號（' '）括住想輸出的文字。

程式的編譯與直譯

當程式撰寫完後，無論程式語言的類型是使用**編譯器**（Compiler）或**直譯器**（Interpreter），第一個動作就是將程式碼**轉譯**為CPU（中央處理單元）能直接讀取的**機器語言**（**機器碼**），再進行後續的執行步驟，而編譯器程式和直譯器程式就是負責這項工作的翻譯程式。

1.編譯器程式

編譯器程式是負責在執行前將所有的程式碼，一次轉換為機器語言，並形成**執行檔**（.exe），供使用者執行程式。如C、C++、Objective-C。

編譯器語言將程式碼轉換為機器碼的過程如下（圖3-2-1）：

⑴程式原始碼透過編譯器程式轉譯為機器碼（目的程式）。

⑵再將機器碼透過連結程式（Linker）與該程式語言的函式庫連結，以提供目的程式所需的內建函式，並產生執行檔。

⑶當每次執行程式時，只要將上一步驟所產生的執行檔藉由作業系統的載入程式（Loader）放入該機器的記憶體（RAM）中，等待CPU的執行，不需要再次進行編譯過程。因此編譯器僅需一次性將程式碼轉換為執行檔即可，但若程式碼有修改則必須重新編譯。

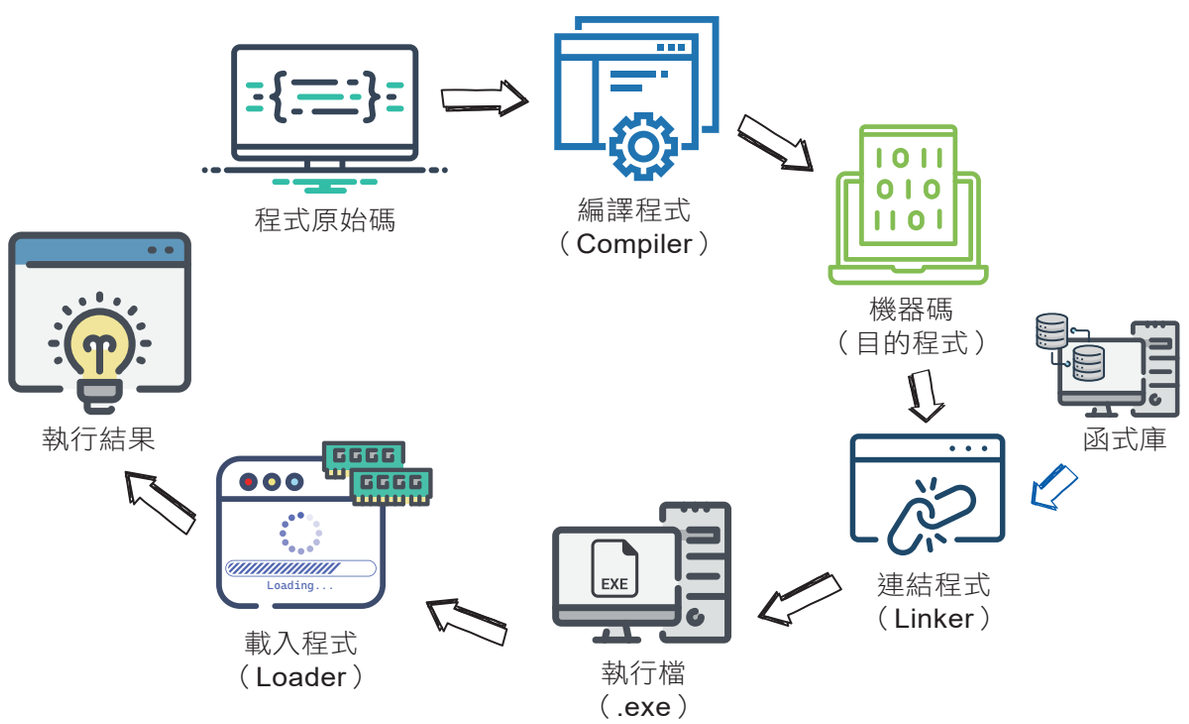


圖3-2-1 編譯程式運行過程

2.直譯器程式

直譯程式是將程式原始碼逐行翻譯為機器碼且同時直接執行後，再繼續翻譯下一段程式原始碼（圖3-2-2）。使用直譯器翻譯程式相較於編譯器程式的缺點在於，直譯器程式在每次執行前都必須進行直譯，相較之下，編譯程式只要編譯一次即可產生執行檔，因此直譯程式在執行時的速度通常較慢。但由於其逐行翻譯並執行的特性，若在轉譯後立即發現錯誤，則會停止執行，並顯示錯誤訊息，因此讓程式在開發階段的測試和**除錯**（Debug）容易許多；也因其具有動態靈活的特性，常用在網頁動態語言上，如JavaScript、Python、Ruby等。

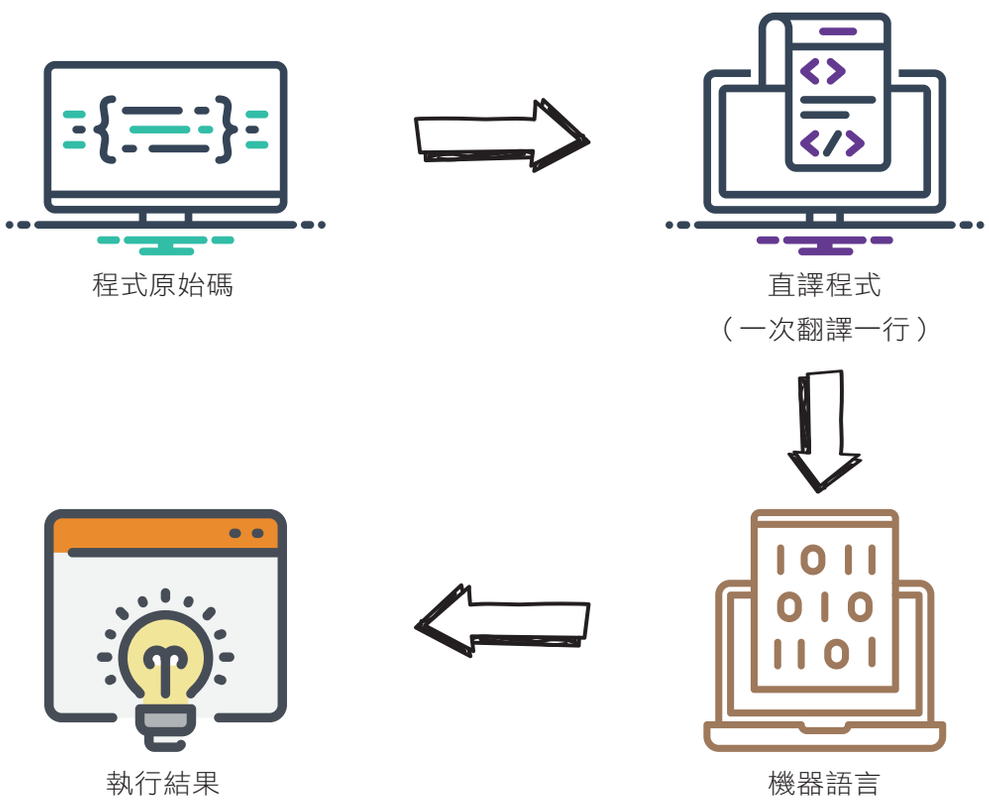


圖3-2-2 直譯程式運行過程

機器語言（Machine Language）

由一系列的0和1所組合而成的二進位指令。只要依據指定的語法就可以將0和1組合成我們想要執行的指令，讓電腦能夠直接執行，不需要再經過任何的轉譯。機器語言具有下列特點：

1.由於指令是以0和1組合而成，不需再經由轉譯即可執行，因此速度最快。

2.對於人類來說，其指令的撰寫必須參照其指定的代碼組合，除了撰寫困難外，若有錯誤也不容易找出。

3-2-3 除錯

程式直譯與執行時期所出現的錯誤可分為**語法錯誤**（Syntax Error）、**語意錯誤**（Semantic Error）、**執行階段錯誤**（Runtime Error）三種：

語法錯誤

可透過直譯器告訴我們錯誤的位置在哪裡，進而修正錯誤的程式碼。通常是打錯字，或型態使用錯誤導致無法直譯通過。如圖3-2-3，提示錯誤的位置是在第 8 行少打了一個字串的雙引號，導致錯誤。

```
1  # -*- coding: utf-8 -*-
2  """
3  Spyder Editor
4
5  This is a temporary script file.
6  """
7
8  print("Hello, Python!)
```

```
    print("Hello, Python!)
          ^
SyntaxError: unterminated string literal (detected at line 8)
```

圖3-2-3 語法錯誤

語意錯誤

語意錯誤又稱為**邏輯錯誤**（Logical Error），從語法上來說雖是一段正確的程式，但在語句中出現了如「公式寫錯」或「演算法的邏輯錯誤」等情況。這在直譯過程中並不會直接顯示如語法錯誤的訊息，且通常仍可順利執行，但其結果會與預期不符。例如：求兩數a，b的平均值，正確應為(a + b) / 2，卻誤植為 a + b / 2，該程式就不會產生我們所預期的答案。

執行階段錯誤

執行階段錯誤即為在程式執行過程中，發生不可預期的錯誤，如執行過程中，使用者輸入錯誤的值，導致程式異常中止。比方說在執行過程中有除法運算，但使用者輸入的分母卻為零，就會造成執行階段的錯誤。

3-3 變數的概念

在程式設計的過程中，我們實際以記憶體來儲存資料，而在程式碼中則是以**變數**作為代號，對記憶體中的資料進行存取，如數學常見的未知數 x、y，可儲存變動的資料。我們可以也自訂變數名稱，賦予一個有意義、易分辨的名稱，方便我們在撰寫程式時存取該變數，例如：x1、x2、y1、y2、num_1、sum 等。

3-3-1 變數的命名規則

變數名稱可依據設計者習慣自訂，但仍必須遵循以下規則：

01 必須由字母（大小寫視為不同變數）、數字、底線（_）、中文組成。

02 不可以數字作為變數開頭。

03 不可使用 Python 的保留字（又稱關鍵字），如表 3-3-1 中的保留字皆有其特殊功能。

表 3-3-1 Python 的保留字

False	class	from	or
None	continue	global	pass
True	def	if	raise
and	del	import	return
as	elif	in	try

assert	else	is	while
async	except	lambda	with
await	finally	nonlocal	yield
break	for	not	

3-3-2 變數的型態

不同的變數型態會有不同的記憶體配置，因此將其分門別類才能更有效的運作，而 Python 常見的變數型態有**數值**和**字串型態**。以下我們將介紹最基本的**宣告**變數名稱同時給予變數值的語法。

語法：
變數名稱 = 變數值

在程式設計中，單一個 = 等號為**指定運算子**，會將等號右側的值或算式結果，指派給左邊的變數。不是數學中「等於」的意思喔！

數值型態

整數（int）：沒有包含小數點的數值。

```
a = 3
b = 9
```

浮點數（float）：有包含小數點的數值。

```
c = 3.5
d = 6.32
```

布林值（bool）：只有True或False兩種狀態。而True以數字1作為替代的值，False以數字0作為替代的值，因此可以將布林值進行運算式的計算。

```
a1 = True
a2 = a1 + 3    變數a2的計算結果為1 + 3 = 4
```

字串（str）型態

Python的字串型態常以成對的雙引號（"..."）或單引號（'...'）將文字包含在內。

```
str1 = "Hello"
str2 = 'Python'
```

3-3-3 輸出與輸入

print( ) 輸出

print( ) 基本輸出語法：

```
print(參數1, 參數2, ... , sep = '間隔字串', end = '結束字串')
```

可省略，預設為空格

可省略，預設為換行符號

print( ) 格式化輸出語法：

```
print("%d ... %s ... %f"%(參數1, 參數2, 參數3)
```

其中，「%d」代表整數輸出格式，其對應參數1的資料；「%s」代表字串輸出格式，其對應參數2的資料；「%f」代表浮點數輸出格式，其對應參數3的資料。

若想要輸出特殊符號時，如換行、縮排4個空白間隔（同Tab鍵）、倒退鍵（Backspace）等，我們可以藉由跳脫字元的方式輸出，如表3-3-2。

表 3-3-2 Python跳脫字元表

跳脫字元	顯示作用
\'	輸出單引號 '
\"	輸出雙引號 "
\\	輸出反斜線 \
\b	倒退鍵（Backspace）
\n	換行
\t	縮排4個空白間隔（同Tab鍵）

input( ) 輸入

input( ) 基本輸入語法：

```
變數 = input("輸入的提示文字")

str1 = input("請輸入年齡")
```

資料型態轉換

若想將原先的資料型態進行型別轉換，可以使用**轉換型別函式**，如下列函式：

表 3-3-3 轉換型別函式

項目	說明
int	（變數或資料）可轉為整數型態
float	（變數或資料）可轉為浮點數型態
str	（變數或資料）可轉為字串型態

轉換語法：

```
變數 = 轉換型別函式名稱(變數或資料)
```

```
a1 = int(33.2)
a2 = str(a1)
a3 = float(20)
print(a1, a2, a3)
```

測試執行

```
33 33 20.0
```

若輸入時的資料型態為字串，想在輸入時同時將其轉換為數值型態，可以使用以下範例語法：

```
變數2 = 轉換型別(input("輸入的提示文字"))

str1 = int(input("請輸入年齡")
```

範例 3-3-1

Q 請寫一程式，可以讓使用者輸入姓名、年齡，並將其輸出至指定句型。

```
a1 = input('輸入姓名:')
a2 = int(input('輸入年齡:'))
print("%s is %d years old!"%(a1, a2))
```

測試執行

```
輸入姓名:Mike
輸入年齡:17
Mike is 17 years old!
```

3-4 變數的運算

3-4-1 運算式

運算式是由運算子與運算元組成的單行敘述，例如：x = x + y;，其中的 x 和 y 皆為運算元，而 = 和 + 為運算子。

3-4-2 運算子

1.一元運算子：由一個運算元組成，例如：x = -y，負號後面僅接一個運算元 y。

2.二元運算子：由兩個運算元組成，例如：a = b + c，加號搭配兩個運算元 b、c。

3.指定運算子：變數名稱 = 運算式或值或另一變數。例如：

```
x = a + b    將 a+b 的值指定給 x
x = 3        將 3 指定給 x
x = y        將 y 指定給 x
```

4.算術運算子：運算須遵循先乘除後加減，括號要先算的原則，如表3-4-1：

表 3-4-1 算術運算子

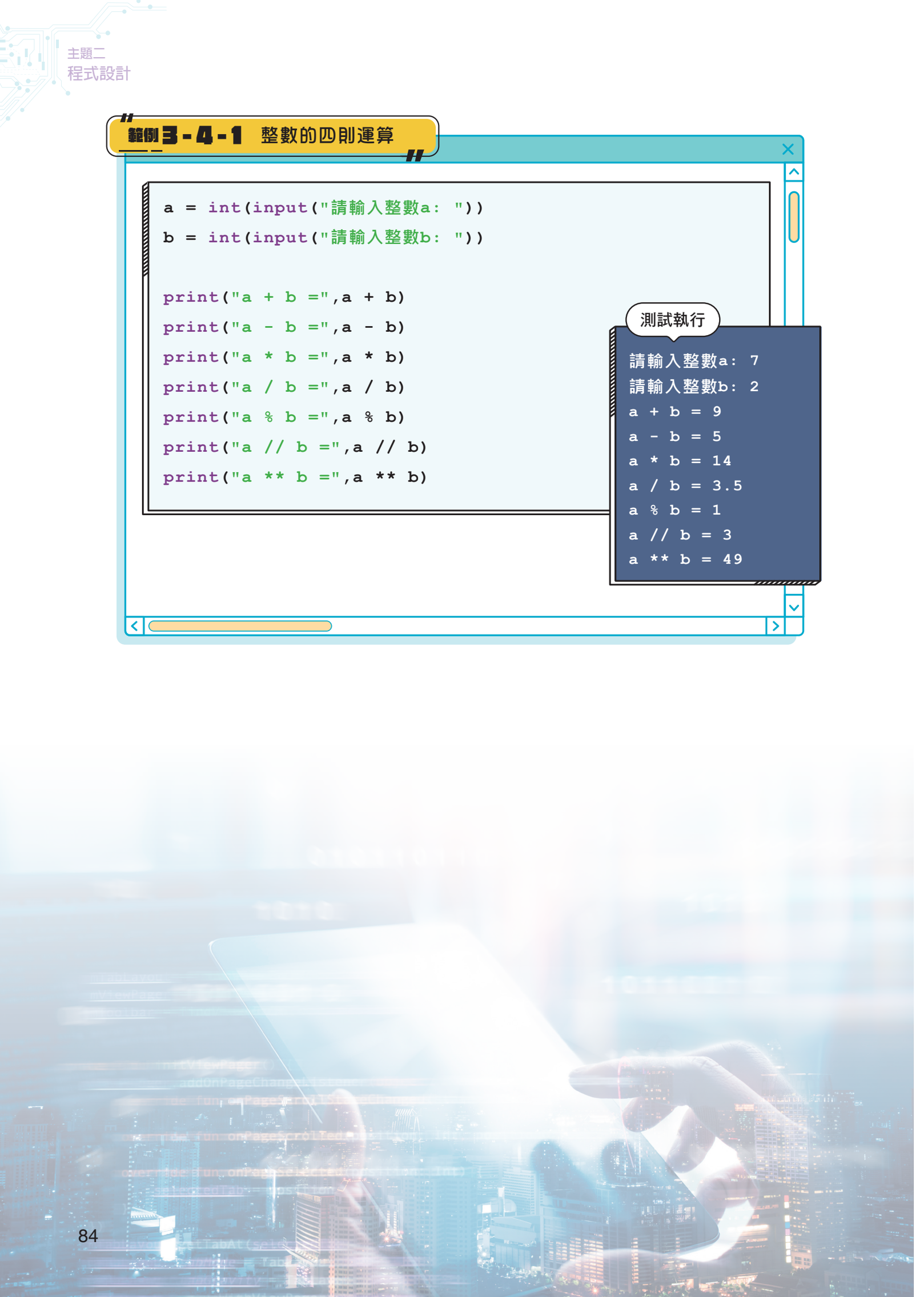
算術運算子	功能	範例	算術運算子	功能	範例
+	加	1+2=3	%	除法（取餘數）	18%4=2
-	減	2-7=-5	//	除法（取整數商）	18//4=4
*	乘	6*3=18	**	a**b a的b次方	3**2=9 3的2次方
/	除法	18/4=4.5			

範例3-4-1 整數的四則運算

```
a = int(input("請輸入整數a: "))
b = int(input("請輸入整數b: "))

print("a + b =",a + b)
print("a - b =",a - b)
print("a * b =",a * b)
print("a / b =",a / b)
print("a % b =",a % b)
print("a // b =",a // b)
print("a ** b =",a ** b)
```

測試執行

```
請輸入整數a: 7
請輸入整數b: 2
a + b = 9
a - b = 5
a * b = 14
a / b = 3.5
a % b = 1
a // b = 3
a ** b = 49
```



補給站

延伸活動 ——交換兩個變數的值

Q1 當A = 1，B = 2，請試著交換兩變數中的值。

解題觀點1：

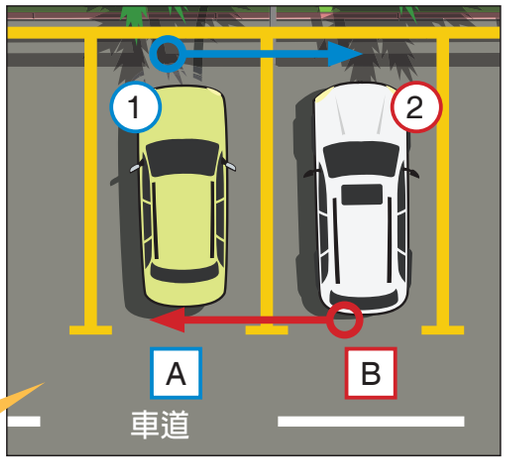
- 嘗試將B的值放入A：
  A = B
- 嘗試將A的值放入B：
  B = A

→則最終結果為：A = 2，B = 2。

我們發現直接將B的值指定給A，則A的值1會被B的值2取代並消失，因此這種方法無法達成兩變數的值交換。

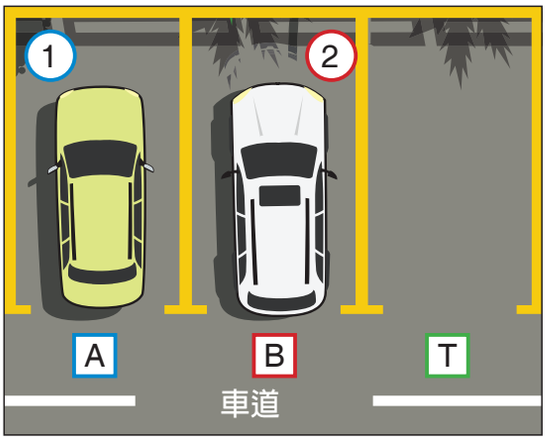
解題觀點2：

我們可以把問題想像成家裡同時有停車位A和停車位B，分別停放了兩部不同顏色的車輛（如右圖）。試問自己能否獨立將兩車位中的車輛進行互換？（同一時間僅能移動一部車子，且車道不能停放）



最一開始，車位A停放1號車，車位B停放2號車，直接互換是不可行的。

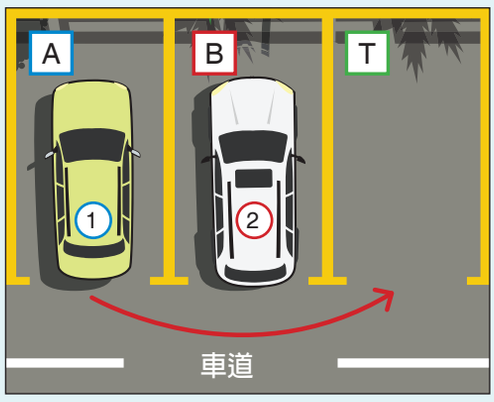
但如果增加一個車位T，如下圖，則可利用T、當作暫時停放的位子：



A = 1　　B = 2

將車位A的1號車停放到車位T

STEP 01

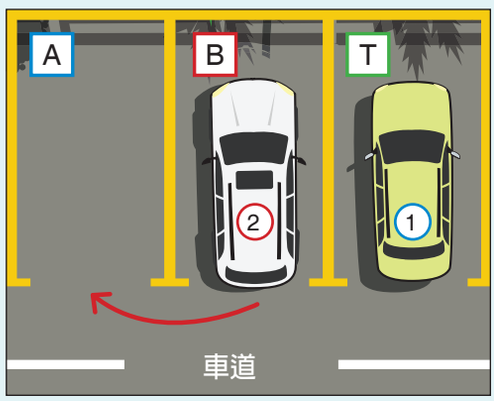


T = A

將車位B的2號車停放到車位A

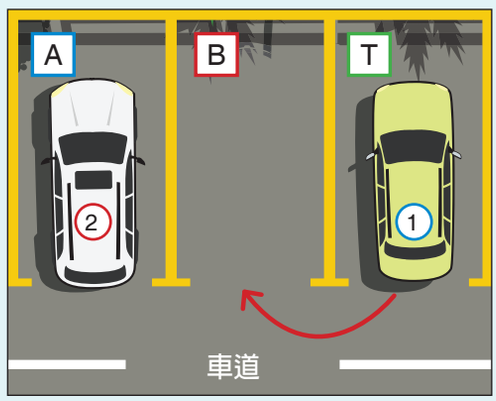
將車位T的1號車停放到車位B

STEP 02



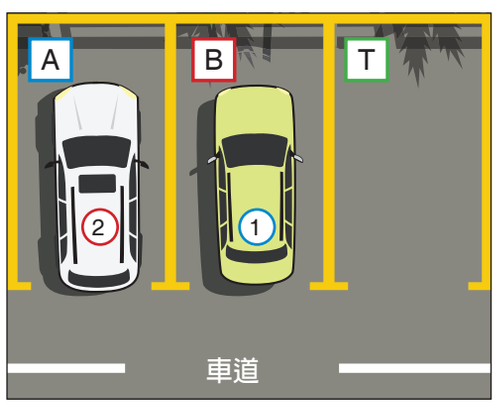
A = B

STEP 03



B = T

完成兩車位中
車輛的互換



範例3-4-2 兩數交換程式

```
a = int(input("請輸入 a:"))
b = int(input("請輸入 b:"))
t = a
a = b
b = t
print("交換後的值 a = ", a, ", b = ", b, sep = "")
```

測試執行

```
請輸入 a:55
請輸入 b:66
交換後的值 a = 66, b = 55
```

範例3-4-3

Q 筆1打12支，賣50元，零賣一支5元。請算出在每人一支筆的情況下，輸入人數所對應的購買總金額。
（不考慮以量制價）

解題過程：

輸入的人數屬於整數型態，因此直接除以12可以先得到「打」的數量（整數）；再次將人數除以12（以%的方式）並取其餘數，此餘數為需要零買的數量；最後分別乘以打的數量和零買的數量，分別得到兩者的費用後，相加即為所求。

```
n = int(input("人數:"))
print("所需費用", (n // 12) * 50 + (n % 12) * 5, "元")
```

測試執行

```
人數:25
所需費用105元
```

範例3-4-4

Q 依座號順序三人一組，請輸入座號後，輸出相對應的組別。

解題過程：

1	2	3	4	5	6	7	8	9	10	11	12	13	...
第一組			第二組			第三組			第四組				

將座號直接除以3取商（組別），可得到下表：

0	0	1	1	1	2	2	2	3	3	3	4	4	...

因此尋找數列的規律性後，若我們將全部座號加2（如下表），則新的座號除以3取商後，即會落在正確的組別上。

3	4	5	6	7	8	9	10	11	12	13	14	15	⋯

座號除以3取商（組別）：

1	1	1	2	2	2	3	3	3	4	4	4	5	⋯

```
a = int(input("座號:"))
print("第",(a + 2) // 3, "組")
```

測試執行

```
座號:16
第 6 組
```

範例3-4-5

Q 請輸入一個三位數，並且將此三位數切割為三個0～9之間的數值後，以空格間隔三數。

```
s = int(input("請輸入三位整數 "))
d1 = s % 10
d2 = s // 10 % 10
d3 = s // 100
print(d3, "\t", d2, "\t", d1, sep ="")
```

測試執行

```
請輸入三位整數 567
5	6	7
```

6.**關係運算子**：比較運算子左右兩運算元（或運算式）的關係，其結果為**布林值**，即True（真）或False（假）。（下表3-4-2運算式中，設a=3, b=2）

表 3-4-2 關係運算子

關係運算子	數學符號	範例	結果
>	>	a>b	True
<	<	a<b	False
== （中間不能有空格）	= （在程式設計中， 單一等號是指定assign的意思）	a==b	False
>= （不能寫成=>）	≥	a>=b	True
<= （不能寫成=<）	≤	a<=b	False
!=	≠	a!=b	True

7.**邏輯運算子**：邏輯運算子包含了and、or、not，亦常用於條件式為一個以上的複合條件式。例如：數學的不等式區間範圍「60≦x≦100」，我們可以運用邏輯運算子表達為 60<=x and x<=100。

表 3-4-3 邏輯運算子

邏輯運算	說明
and	當兩個運算元都非False，則此 and 運算式會回傳值 True。 若任一運算元為False，則結果為 False。
or	若任一運算元有非True，則結果為True。若兩個運算元皆為False，則結果為False。
not	將True轉為False，False轉為True。

表 3-4-4 邏輯運算與真值表

A	B	A and B	A or B	not A
False	False	False	False	True
True	False	False	True	False
False	True	False	True	True
True	True	True	True	False

3-4-3 優先權與結合性

在數學四則運算中，我們學過了先乘除後加減，括號要先算。相同地，在程式語言中也是有其運算規則，且程式語言的運算子更多、更複雜，因此必須為這些運算子訂定其優先權與結合性。

運算式中的運算規則依序如下：

1.先區分運算式中的運算子與運算元。如sum = a + b，「sum」、「a」、「b」為運算元，「=」、「+」為運算子。

2.比較運算子的優先權（參照表 3-4-5）。如sum = a + b中，「=」的優先權為10，「+」的優先權為5，因此先執行「+」運算子，而後再執行「=」運算子。

3.由該優先運算子依照其結合性的規則計算，若相鄰的運算子優先權順序相同，則依照結合性運算。當由左而右時，以最左側的運算子開始計算。當由右而左時，以最右側的運算子開始計算。如a + b - c，「+」、「-」運算子的優先權相同，則依據其結合性由左至右，由最左側的「+」運算子先執行，而後再往右執行「-」運算子，等同於((a + b) - c)。

Python運算子的優先順序如表3-4-5（由高到低）：

表 3-4-5 運算子優先權

優先順序	運算子
1	()
2	+（正）、-（負）
3	**
4	*、/、//、%
5	+（加）、-（減）
6	>、>=、<、<=、!=、==
7	not
8	and
9	or
10	=、+=、-=、*=、/=、//=、%=、**=

範例3-4-6

Q 當 b = 3、c = 1 時，經過 a = b <= 5 and c == 0 敘述後，請問最終變數 a 的值為何？

解題過程：

```
a = b <= 5 and c == 0
    ↓    ↓     ↓
   10    6     8      6
```

優先權

1.先計算 b <= 5，結果為True。（3<=5為True）
2.再計算 c == 0，結果為False。（1==0為False）
3.然後進行 and 運算，True and False，結果為False。
4.最後將運算結果設定給變數a存放，故 a = False。

<= 的優先順序為6
== 的優先順序為6
&& 的優先順序為8
= 的優先順序為10

3-5 結構化程式設計

在軟體開發的過程中，常以**結構化程式設計**作為規劃軟體架構的技術，其使用「**由上而下**」的分析方式取得軟體的細部需求，並將軟體逐步分解成一個個在階層架構中的「模組」（Module），而每個模組擁有獨立功能的程式片段（模組化）。模組中只有單一的進入點和離開點（**一進一出**），且各模組運用三種模式來設計其程式片段：**循序結構**（Sequential）、**選擇結構**（Selection）、**重複結構**（Iteration）。

結構化程式設計主要特性如下：

1. **模組化**：將大的系統架構劃分為功能各自獨立的小模組。而在程式實作時，通常將以副程式（無回傳值）或函數（有回傳值）兩種程式片段完成模組化的設計。模組化能達到功能的獨立，也能增加程式的再利用性；藉由模組化的設計，方便程式設計師進行分工，可以減輕人員的負擔。
2. 程式採用一進一出的規劃，使其容易維護及掌控程式執行的方向。
3. 由上而下的設計方法：結構化程式設計是將大系統由上而下、逐步進行整體架構的設計，並在向下的過程中逐步將程式功能劃分為數個功能獨立的小模組。

結構化程式設計一般包含三種模式：

01	02	03
循序結構	**選擇結構**	**重複結構**
程式一步一步地由上而下依序執行。	依據條件式選擇相對應的敘述執行。	在滿足條件式的情形下，反覆執行迴圈內部敘述。

3-5-1 循序結構

循序結構是結構化程式設計的基礎架構，將程式的第一行敘述做為進入點開始，由上而下地依序執行每行程式敘述，直到最後一行敘述為止，如下圖3-5-1。

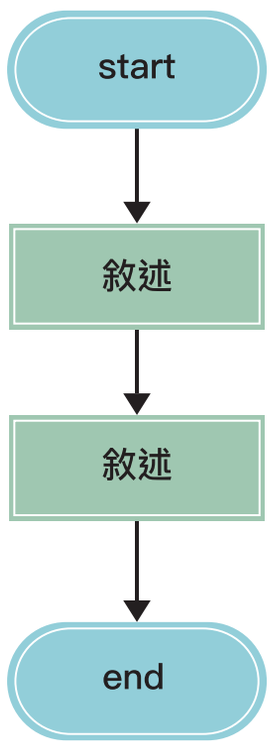


圖3-5-1 循序流程圖

範例3-5-1 計算三角形面積的循序結構與程式碼

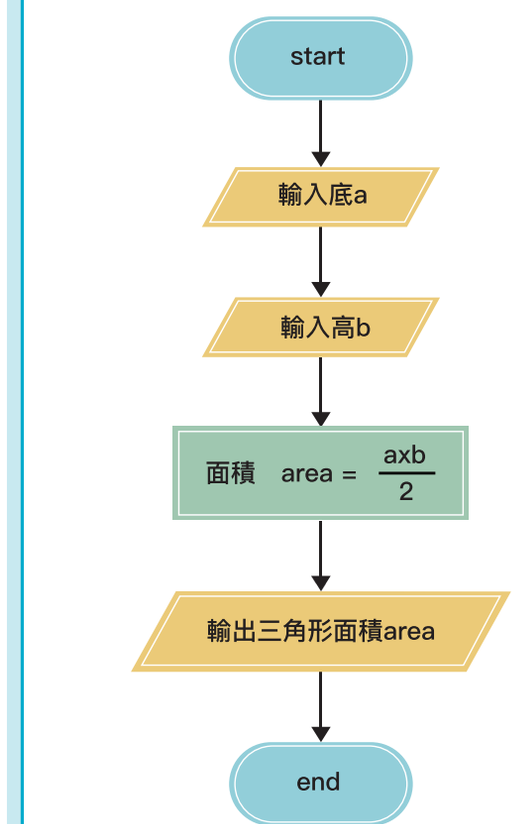


圖3-5-2 範例流程圖

```
a = int(input("輸入a:"))
b = int(input("輸入b:"))
area = (a * b) / 2.0
print(area)
```

測試執行

```
輸入a:3
輸入b:5
7.5
```

3-5-2 選擇結構 if

選擇結構的用途為在**單一條件**、**雙重條件**或**多重條件**下，根據不同條件選擇適切的程式敘述並加以執行。以下示範三種選擇結構的使用方式。

if的用法

單向選擇結構：

若符合條件式，則執行條件式後的敘述；若不符合，則直接離開選擇結構，繼續執行後面敘述或結束程式。適合用在單一條件的選擇，如分數大於且等於60分，則輸出「及格」。

```
if 條件式:
    程式敘述
    ……
    程式敘述
```

如圖3-5-3

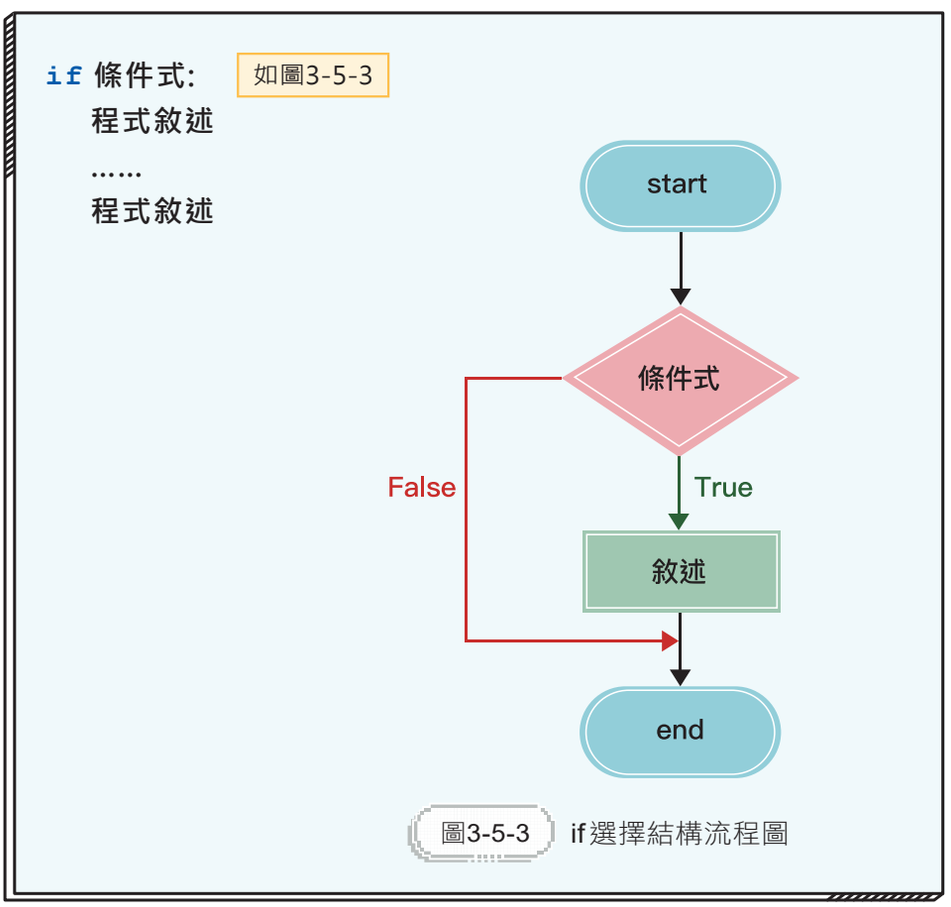


圖3-5-3 if選擇結構流程圖

⑴當if後的條件式成立(True)，才執行後面的敘述，否則(False)將直接跳出，不執行敘述。

⑵Python程式語言有別於多數程式是使用大括號{...}作為執行程式敘述的區塊範圍，Python程式語言運用4個空白（或一個Tab鍵）作為縮排表達所屬之程式敘述的區塊範圍。

範例3-5-2

Q 請輸入期末成績並進行是否及格的判斷。分數為整數0～100間，60分以上為及格。

解題過程：

1.輸入成績後，可以運用單一條件的 if 選擇結構進行判斷。

2.當大於等於60分，輸出及格；當小於60分，輸出不及格。

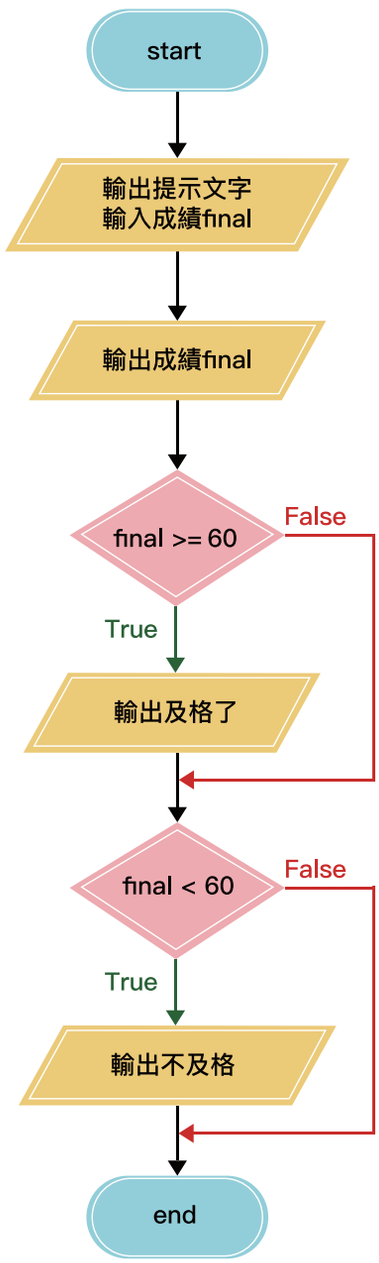


圖3-5-4 範例流程圖

```
final = int(input("請輸入您的成績:"))
print("您的成績為:",final)
if final >= 60:
    print("及格了，繼續保持!!")
if final < 60:
    print("不及格，請加油!!")
```

測試執行 1

```
請輸入您的成績:83
您的成績為: 83
及格了, 繼續保持!!
```

測試執行 2

```
請輸入您的成績:52
您的成績為: 52
不及格, 請加油!!
```

3-5-3 選擇結構if-else

if-else的用法

雙向選擇結構：

若符合條件式（True）則執行條件式後的第一個區塊範圍的敘述，若不符合條件式（False）則執行選擇結構中的另一區塊範圍的敘述（圖3-5-5），適合用在**二分法**的條件，如分數大於且等於60分，則輸出「及格!」，否則輸出「不及格!」。

```
if 條件式:
    敘述1
else
    敘述2
```

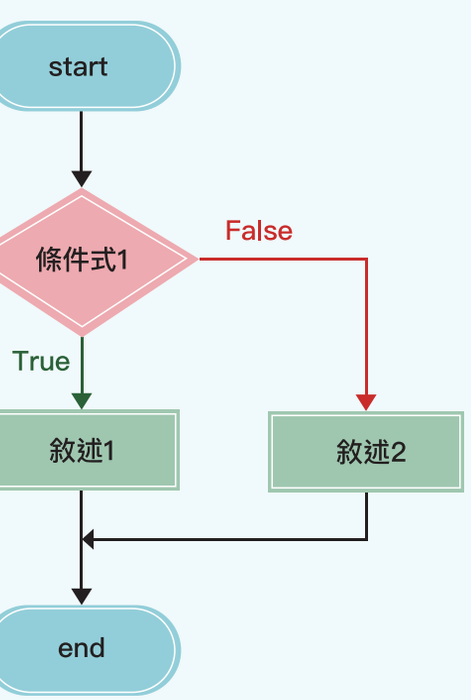


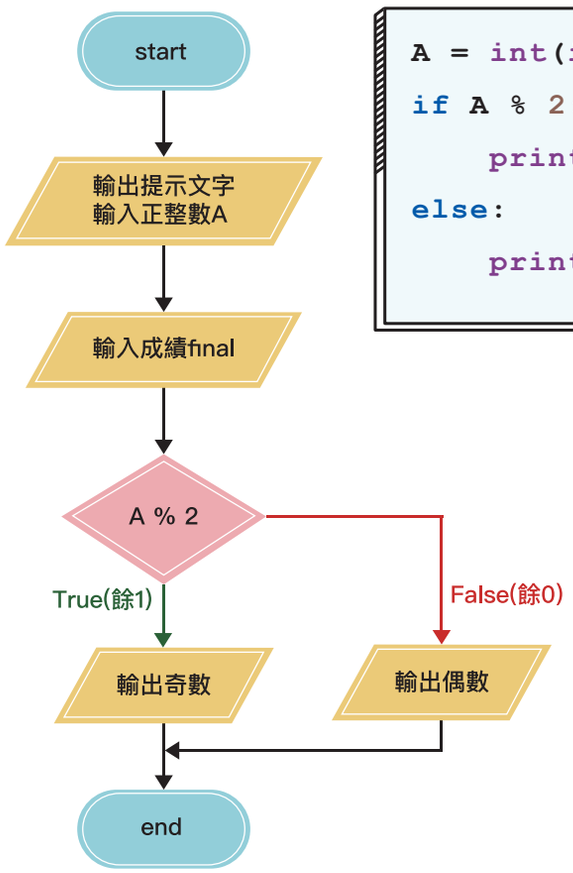
圖3-5-5 if-else選擇結構流程圖

範例3-5-3 輸入正整數並判斷其奇偶數

解題過程：

1.先輸入正整數後，若將其除以2取餘數，會有兩種可能：餘1為布林值的True，則為奇數；整除（餘0）為布林值的False，則為偶數。

2.僅有兩種可能，因此可以選擇使用雙向選擇結構的策略解題。




```
A = int(input("請輸入一個正整數:"))
if A % 2:
    print("這是奇數")
else:
    print("這是偶數")
```

當餘數為0時，等同布林值的False；
當餘數為1時，等同布林值的True。

圖3-5-6 範例流程圖

測試執行 1

```
請輸入一個正整數:55
這是奇數
```

測試執行 2

```
請輸入一個正整數:52
這是偶數
```

3-5-4 選擇結構if-elif

if-elif的用法

多重選擇結構：

為雙重選擇結構的延伸應用，當二分法不敷使用時，多重選擇結構更適合用在多選一的情況下，有多個條件式對應多個敘述區塊，若符合其中一個條件，則進入其所屬敘述區塊執行。if-elif 的選擇必定是前一個條件式沒有成立，才會依序往後面的條件式進行判斷。

if-elif

```
if 條件式1:
    敘述1
    ...
elif 條件式2:
    敘述2
elif 條件式n:
    敘述n
    ...
else:
    敘述
    ...
```

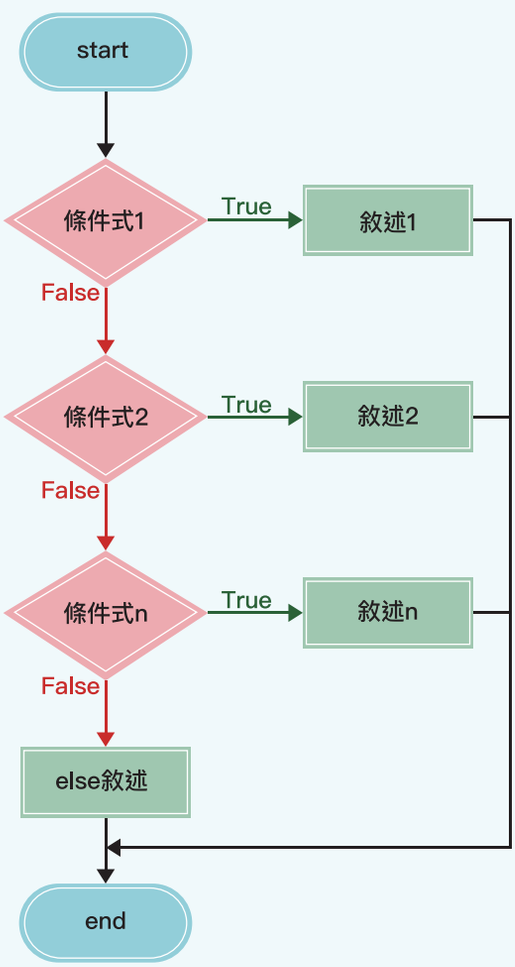


圖3-5-7 if-elif 選擇結構流程圖

if-elif（圖3-5-7）在運用多個條件式的情況下，程式會依序**由上而下**判斷每個條件式是否成立（True），若成立則僅執行該敘述後跳出結構；若完全沒有條件式成立，則會執行最後else的敘述，即「以上皆非」之意。

範例3-5-4 輸入三個正整數，並輸出最大值

解題過程：

1.輸入三數後，最大值可能有三種可能性，因此可以使用多重選擇結構進行判斷。

2.當a為最大值時，則a>b且a>c；當前面條件式不成立時，若b為最大值時，僅需判斷b>c；若前面兩項條件式皆不成立時，則最大值為c。

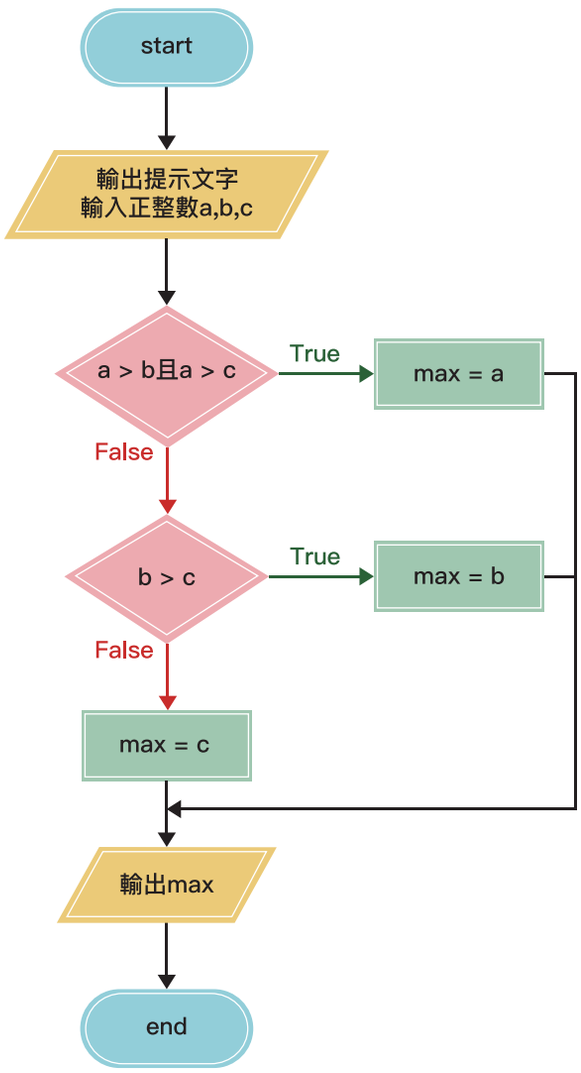


圖3-5-8 範例流程圖

```
a = int(input("請輸入a:"))
b = int(input("請輸入b:"))
c = int(input("請輸入c:"))
if a > b and a > c:
    max = a
elif b > c:
    max = b
else:
    max = c
print("Max =", max)
```

測試執行

```
請輸入a:55
請輸入b:44
請輸入c:98
Max = 98
```

巢狀選擇結構

我們可以在選擇結構當中再加入更多的選擇結構，此種結構稱為**巢狀選擇結構**。巢狀選擇結構適合用在當問題需要有兩個（含）以上的條件式且必須同時成立時，但也可以運用**邏輯運算子** and 同時結合兩個條件式，同樣可以解決此類問題。

範例3-5-5

Q 成績及格與補考條件判斷：輸入成績後，當成績大於等於60分，輸出「及格」；若分數達90分以上，則輸出「太優秀了！」；當成績小於60分，輸出「不及格」；若分數未滿40分，輸出「不能進行補考」。

解題過程：

1. 當分數大於等於60會有兩種選擇：大於等於90要輸出「太優秀了」，而其餘輸出「及格」。
2. 當分數小於60也會有兩種選擇：小於40要輸出「不能補考」，其餘輸出「不及格」。
3. 因此我們可以運用巢狀選擇結構將上述兩道步驟結合，如流程圖。

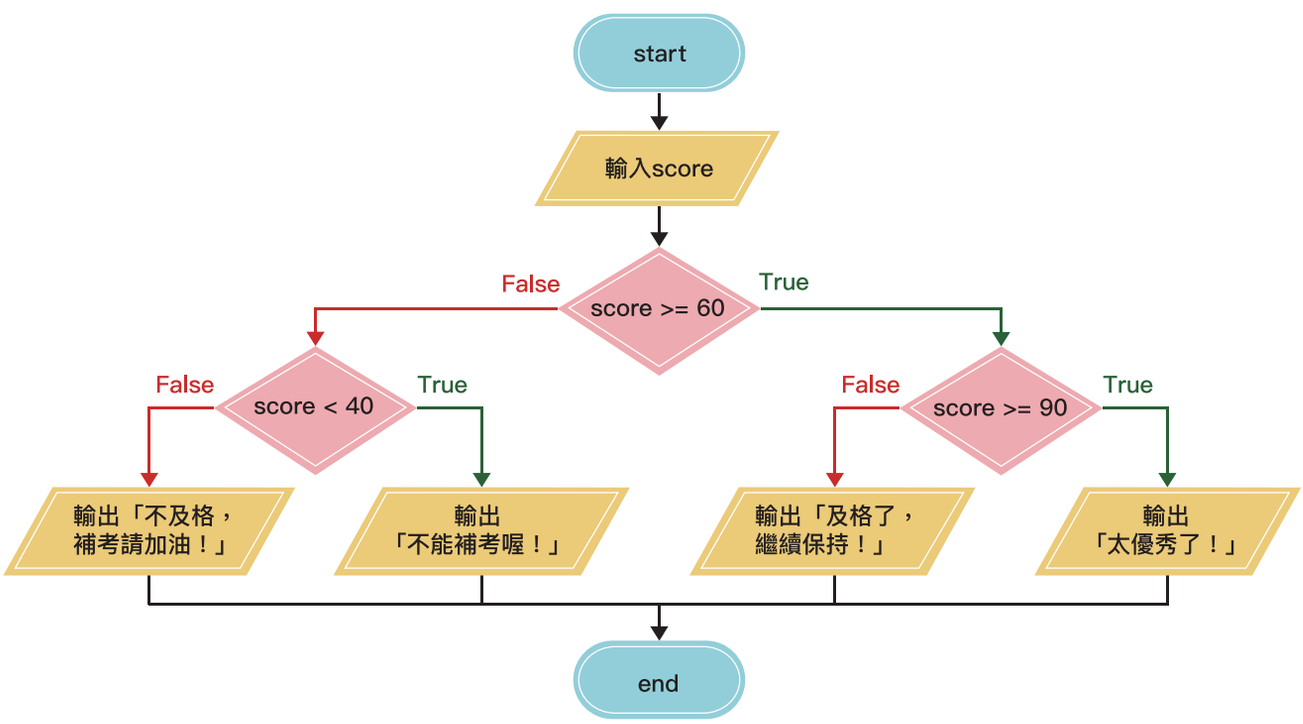


圖3-5-9 範例流程圖

```
score = int(input("score = "))
if score >= 60:
    if score >= 90:
        print("太優秀了！")
    else:
        print("及格了，繼續保持！")
else:
    if score < 40:
        print("不能補考喔！")
    else:
        print("不及格，補考請加油！")
```

測試執行 1

```
score = 93
太優秀了！
```

測試執行 2

```
score = 70
及格了，繼續保持！
```

測試執行 3

```
score = 58
不及格，補考請加油！
```

測試執行 4

```
score = 32
不能補考喔！
```

科技即戰力

請設計一個程式，能在輸入西元年後，依據下列規則判斷並輸出閏年或平年。

1.西元年份除以4不可整除，為平年。
2.西元年份除以4可整除，且除以100不可整除，為閏年。
3.西元年份除以100可整除，且除以400不可整除，為平年。
4.西元年份除以400可整除，為閏年。

實作解析3-5-1

```
y = int(input("請輸入西元年 : "))
if y % 400 ==0:
    print(y, "為閏年")
elif y % 4 ==0 and y % 100 != 0:
    print(y, "為閏年")
else:
    print(y, "為平年")
```

測試執行 1

```
請輸入西元年：2000
2000 為閏年
```

測試執行 2

```
請輸入西元年：2020
2020 為閏年
```

測試執行 3

```
請輸入西元年：2023
2023 為平年
```

3-5-5 重複結構（迴圈）

for迴圈

語法與流程圖

```
for 變數 in range(起始值, 終止值, 更新值):
    敘述區塊
```

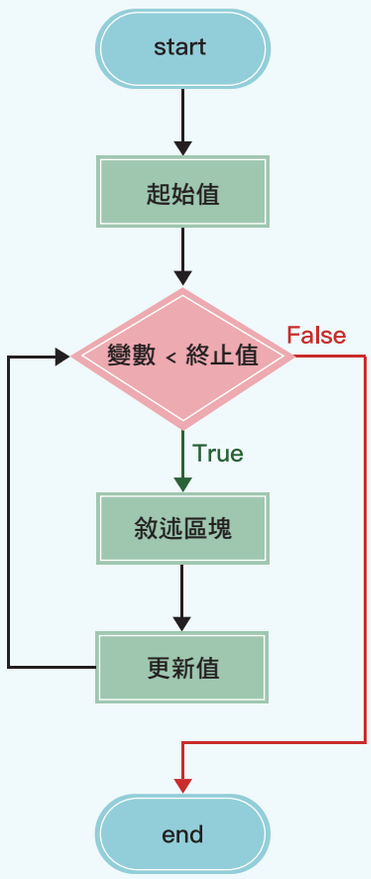


圖3-5-10 for迴圈結構流程圖

1. **起始值**：在進入for迴圈開始時，變數所設定的初始值，可省略，省略後其預設值為0開始。
2. **終止值**：在每次要進入敘述區塊前，變數皆要先判斷是否小於終止值才進入敘述。True才進入敘述迴圈區塊，包含起始值與往後每一次的更新值。
3. **更新值**：當每一次完成敘述區塊後，皆會透過更新值遞增或遞減其值，再透過終止值判斷是否再次進入敘述迴圈。可省略，省略後其預設值為1（圖3-5-10）。

範例3-5-6 計算 1 + 2 + 3 + ... + 10 的總和

解題過程：

1.可以設定儲存總和的變數，其初始值為0。

2.運用迴圈，迴圈變數 i 的起始條件為1，繼續條件為當 i 小於等於10，更新條件每次將 i 加1，並將每次變化的 i 值累加進入sum，即為所求。

```
sum = 0
for i in range(1, 11):
    print("sum + i =", sum, end="")
    sum = sum + i
    print(" +", i, "=", sum)
print("sum =", sum)
```

測試執行

```
sum + i = 0 + 1 = 1
sum + i = 1 + 2 = 3
sum + i = 3 + 3 = 6
sum + i = 6 + 4 = 10
sum + i = 10 + 5 = 15
sum + i = 15 + 6 = 21
sum + i = 21 + 7 = 28
sum + i = 28 + 8 = 36
sum + i = 36 + 9 = 45
sum + i = 45 + 10 = 55
sum = 55
```

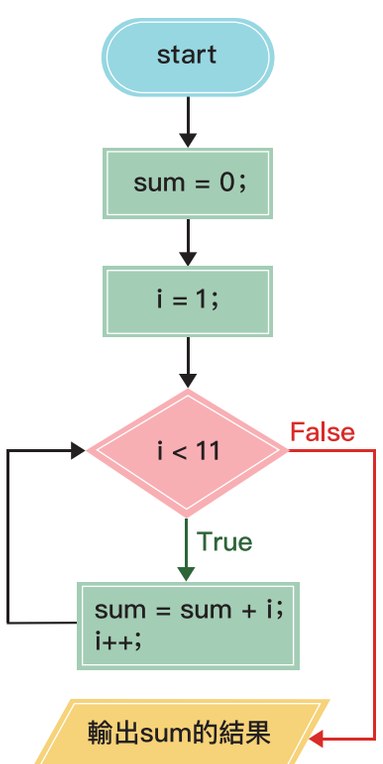


圖3-5-11 範例流程圖

巢狀for迴圈

我們可以在for迴圈結構當中再加入一層（含）以上的for迴圈，稱為**巢狀for迴圈**。先前介紹的for迴圈是控制單一變數，而雙重for迴圈的使用時機在於同時需要控制兩個變數時，可以運用外部與內部迴圈相互搭配以完成需求。

例如：想列出九九乘法表，可運用外迴圈控制同一列的被乘數，內迴圈控制同一列的乘數，當被乘數從1開始，觀察每一列的乘數的變化，當乘數從1遞增到9之後，被乘數才會加1。而被乘數加1之後，乘數又重新從1開始，直到外迴圈結束為止，這就是巢狀迴圈的應用。

內迴圈 / 外迴圈	乘數＝1	乘數＝2	乘數＝3	…	乘數＝8	乘數＝9
被乘數＝1	1×1＝1	1×2＝2	1×3＝3	…	1×8＝8	1×9＝9
被乘數＝2	2×1＝2	2×2＝4	2×3＝6	…	2×8＝16	2×9＝18
被乘數＝3	3×1＝3	3×2＝6	3×3＝9	…	3×8＝24	3×9＝27
⋮	⋮	⋮	⋮		⋮	⋮
被乘數＝8	8×1＝8	8×2＝16	8×3＝24	…	8×8＝64	8×9＝72
被乘數＝9	9×1＝9	9×2＝18	9×3＝27	…	9×8＝72	9×9＝81

範例3-5-7 九九乘法表

```
for i in range(1, 10):          外迴圈控制被乘數
    for j in range(1, 10):      內迴圈控制乘數
        print(i, "*", j, "=", i*j, " \t ", end="")
    print()
```

測試執行（九九乘法表）

```
1 * 1 = 1  1 * 2 = 2  1 * 3 = 3  1 * 4 = 4  1 * 5 = 5  1 * 6 = 6  1 * 7 = 7  1 * 8 = 8  1 * 9 = 9
2 * 1 = 2  2 * 2 = 4  2 * 3 = 6  2 * 4 = 8  2 * 5 = 10 2 * 6 = 12 2 * 7 = 14 2 * 8 = 16 2 * 9 = 18
3 * 1 = 3  3 * 2 = 6  3 * 3 = 9  3 * 4 = 12 3 * 5 = 15 3 * 6 = 18 3 * 7 = 21 3 * 8 = 24 3 * 9 = 27
4 * 1 = 4  4 * 2 = 8  4 * 3 = 12 4 * 4 = 16 4 * 5 = 20 4 * 6 = 24 4 * 7 = 28 4 * 8 = 32 4 * 9 = 36
5 * 1 = 5  5 * 2 = 10 5 * 3 = 15 5 * 4 = 20 5 * 5 = 25 5 * 6 = 30 5 * 7 = 35 5 * 8 = 40 5 * 9 = 45
6 * 1 = 6  6 * 2 = 12 6 * 3 = 18 6 * 4 = 24 6 * 5 = 30 6 * 6 = 36 6 * 7 = 42 6 * 8 = 48 6 * 9 = 54
7 * 1 = 7  7 * 2 = 14 7 * 3 = 21 7 * 4 = 28 7 * 5 = 35 7 * 6 = 42 7 * 7 = 49 7 * 8 = 56 7 * 9 = 63
8 * 1 = 8  8 * 2 = 16 8 * 3 = 24 8 * 4 = 32 8 * 5 = 40 8 * 6 = 48 8 * 7 = 56 8 * 8 = 64 8 * 9 = 72
9 * 1 = 9  9 * 2 = 18 9 * 3 = 27 9 * 4 = 36 9 * 5 = 45 9 * 6 = 54 9 * 7 = 63 9 * 8 = 72 9 * 9 = 81
```

3-5-6 while迴圈

while loop

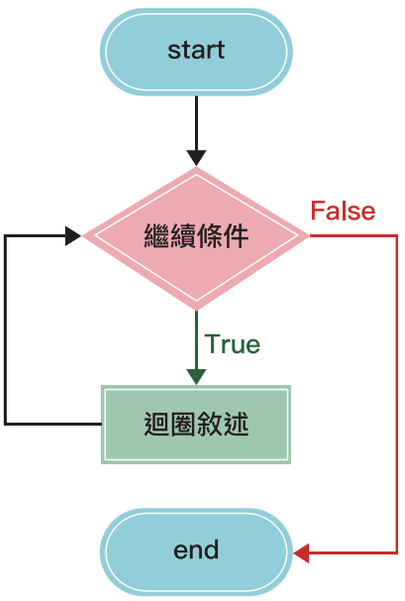


圖3-5-12 while迴圈結構流程圖

while 迴圈語法

```
while 繼續條件:
    敘述
    敘述
    ...
```

改寫 for 迴圈

```
起始值
while 繼續條件:
    敘述
    敘述
    ...
    更新值
```

當 while 後的繼續條件成立（True）時，則會進行迴圈內的敘述；若不成立時（False），則不進入迴圈。因此在執行迴圈敘述前，會先進行繼續條件判斷，當繼續條件一開始就不成立（False）的時候，迴圈執行次數就是 0。而在迴圈內部也可以插入 break 指令作為中斷迴圈的執行，或是插入 continue 指令以直接跳往下一次迴圈。

範例3-5-8 數字反轉

解題過程：

若輸入的整數數字為「327」，我們想將之轉換為「723」，可以運用取餘數與除法的運算子進行下列步驟。

1.先以%運算子除以10取其餘數（327%10獲得個位數7）並直接輸出做為最前面的位數。

2.再將原先的數字以除法運算子除以10（327/10獲得尚未輸出的32），此步驟去除已輸出的個位數部分。

3.當步驟2的商大於0時（代表仍有數字尚未輸出），再重覆步驟1、2，即可完成數字的反轉。

```
i = int(input("請輸入數字： "))
print("反轉後的數字為： ", end="")
while(i > 0):
    print(i % 10, end="")
    i = i // 10
```

測試執行

```
請輸入數字：7892
反轉後的數字為：2987
```

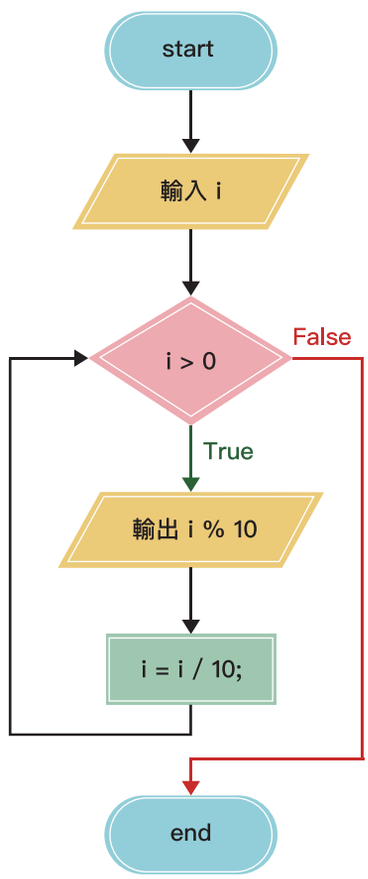


圖3-5-13 範例流程圖

範例3-5-9　找出所有因數

解題過程：

解題前先釐清因數的定義如下：對於正整數 a、b、q，若 b＝a×q，則 a 為 b 的因數，可以記為 a|b，讀作 a 整除 b。例如：24＝3×8，3、8 皆為 24 的因數。

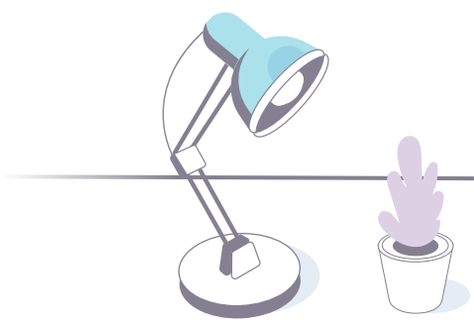
在 Chapter 02 我們有介紹過暴力法的解題策略，若有一正整數 x ，我們可以運用迴圈的方式，將所有可能的因數（y，其中1 ≤ y ≤ x）一一取出，並檢視其是否為因倍數關係。

若 y 為 x 的因數，則 y 可以整除 x，故以取餘數運算子判斷得到的餘數為零者，即為 x 的因數。

```
x = int(input("請輸入數字： "))
y = 1
while(y <= x):
    if(x % y == 0):
        print(y, end=" ")
    y = y + 1
```

測試執行

```
請輸入數字： 92
1 2 4 23 46 92
```



範例3-5-10 找出兩數的最大公因數（輾轉相除法）

解題過程：

當要找出兩數的最大公因數，我們可以使用輾轉相除法的方式解決此類問題，其過程如下：

1.輸入兩正整數a、b（a > b），令a為被除數，b為除數，開始進行兩數相除取餘數。

2.並將本次運算過程的除數和餘數分別作為下一次相除的被除數、除數。

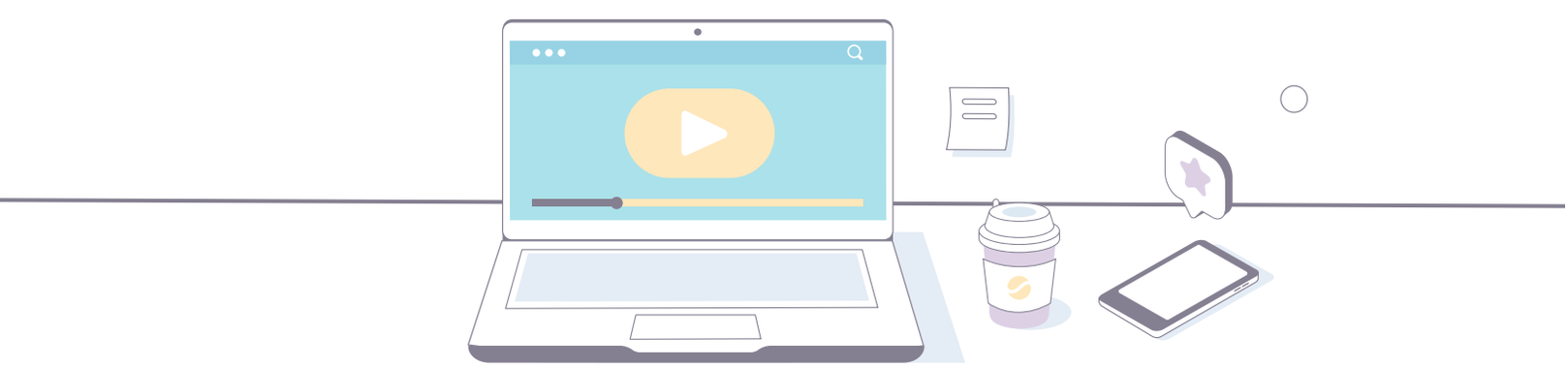
3.運用迴圈反覆進行步驟1、2直到整除為止，整除前的最後一個除數即為最大公因數。

```
a = int(input("請輸入 a 數字 : "))
b = int(input("請輸入 b 數字 : "))
while(a % b != 0):
    t = a % b
    a = b
    b = t
print(b, "為最大公因數")
```

測試執行

```
請輸入 a 數字：52
請輸入 b 數字：36
4 為最大公因數
```

Learn More →



break and continue指令

1.break

在迴圈的執行中，只要穿插break指令，即會跳離所屬之for或while迴圈結構，進行迴圈外的下一個敘述。

2.continue

continue指令同樣也可用來控制迴圈的執行。不同之處在於，break會結束所屬迴圈的執行，而continue只會立即結束當次迴圈，略過剩下的敘述，直接跳到下一次迴圈的開始位置，而非離開迴圈。

範例3-5-11 印出 0，2，4，6，8，

```
for i in range(10):
    if i % 2:
        continue
    print(i, ", ", end="")
```

當i除以2餘1（代表其為奇數）時，進入選擇結構中並執行continue指令後，立即結束當次迴圈，略過下方輸出i的敘述，直接回到更新條件i++，繼續執行下一次迴圈。

測試執行

```
0 , 2 , 4 , 6 , 8 ,
```

執行過程如下：

i 值	i % 2 的值	執行狀態
i＝0	0	輸出 0
i＝1	1	continue，跳到下一次迴圈，進行i=i+1
i＝2	0	輸出 2
i＝3	1	continue，跳到下一次迴圈，進行i=i+1
i＝4	0	輸出 4
i＝5	1	continue，跳到下一次迴圈，進行i=i+1
i＝6	0	輸出6
i＝7	1	continue，跳到下一次迴圈，進行i=i+1
i＝8	0	輸出8

i 值	i % 2 的值	執行狀態
i＝9	1	continue，跳到下一次迴圈，進行i=i+1
i＝10		結束迴圈

範例3-5-12 求 1 至 20 的所有質數

解題過程：

1.質數的定義為除了1和本身兩個因數以外，沒有其他的因數可以整除自己。

2.因此從最小的質數開始逐一檢驗是否為質數，由外迴圈 i 控制2～20，並遞增。

3.再將步驟 2 的 i 值由內迴圈進行檢驗質數的工作，並設定一個旗標 flag，預設值為 0（當 flag＝0 為質數，flag＝1 為合數）。依據質數的定義由內迴圈查驗 2 ～ i-1 中，是否有因數能夠整除當前的 i 值，若有則 flag 設定為 1，並輸出「 i 不是質數」後，直接結束內迴圈，往內迴圈外的下一行敘述執行。

4.當外迴圈的 i 值由內迴圈進行檢驗質數結束後，若其 flag 仍為 0，代表目標數 i 為質數，並輸出「 i 是質數」。

範例3-5-12 求1至20的所有質數（續）

```
max = 20
for i in range(2, max+1):
    flag = 0

    for j in range(2, i):
        if i % j == 0:
            flag = 1
            print(i, "不是質數")
            break
    if flag == 0:
            print(i, "是質數")
```

直接跳離所屬迴圈

測試執行

```
2 是質數
3 是質數
4 不是質數
5 是質數
6 不是質數
7 是質數
8 不是質數
9 不是質數
10 不是質數
11 是質數
12 不是質數
13 是質數
14 不是質數
15 不是質數
16 不是質數
17 是質數
18 不是質數
19 是質數
20 不是質數
```

科技即戰力

請撰寫一猜數字的遊戲程式。遊戲一開始可以先指定並輸入一個介於1～100之間的答案，而玩家在每次猜測並輸入數字的過程中，都會給予提示訊息（「恭喜答對了！」、「太大了！」或「太小了！」）；若輸入負值則結束遊戲，並輸出「您不想玩了嗎？」；輸入大於100則繼續遊戲。

實作解析3-5-2

```
ans = int(input("請輸入答案1-100 : "))
guess = 0
while ans != guess:
    guess = int(input("請輸入你猜的數字1-100 : "))
    if guess < 0:
        break
    if guess > 100:
        continue
    if guess > ans:
        print("太大了！")
    elif guess < ans:
        print("太小了！")
if ans == guess:
    print("恭喜答對了！")
else:
    print("您不想玩了嗎？")
```

輸入負數，停止遊戲

輸入大於100，繼續遊戲

測試執行

```
請輸入答案1-100 : 56
請輸入你猜的數字1-100 : 73
太大了！
請輸入你猜的數字1-100 : 25
太小了！
請輸入你猜的數字1-100 : 68
太大了！
請輸入你猜的數字1-100 : 56
恭喜答對了！
```

3-6 模組化與函式的概念

3-6-1 模組化

在國中時，我們就曾在Scratch中學到了**模組化**的概念，運用自訂函式積木來撰寫功能函式（或稱副程式），將具有相同功能的程式片段，獨立為模組化的函式積木，因此主程式便可以更專注於完成各項任務，而不是處理任務的細節。如下圖，在模組化前，程式積木是循序地由上而下執行；模組化後的程式，將可以畫出五角形的程式積木集，編入五角形的函式積木中。雖然模組化前後兩者的執行結果一樣，但繪製五角形的函式積木，可供程式重複呼叫使用，並且讓主要執行的程式區塊更簡潔易懂。

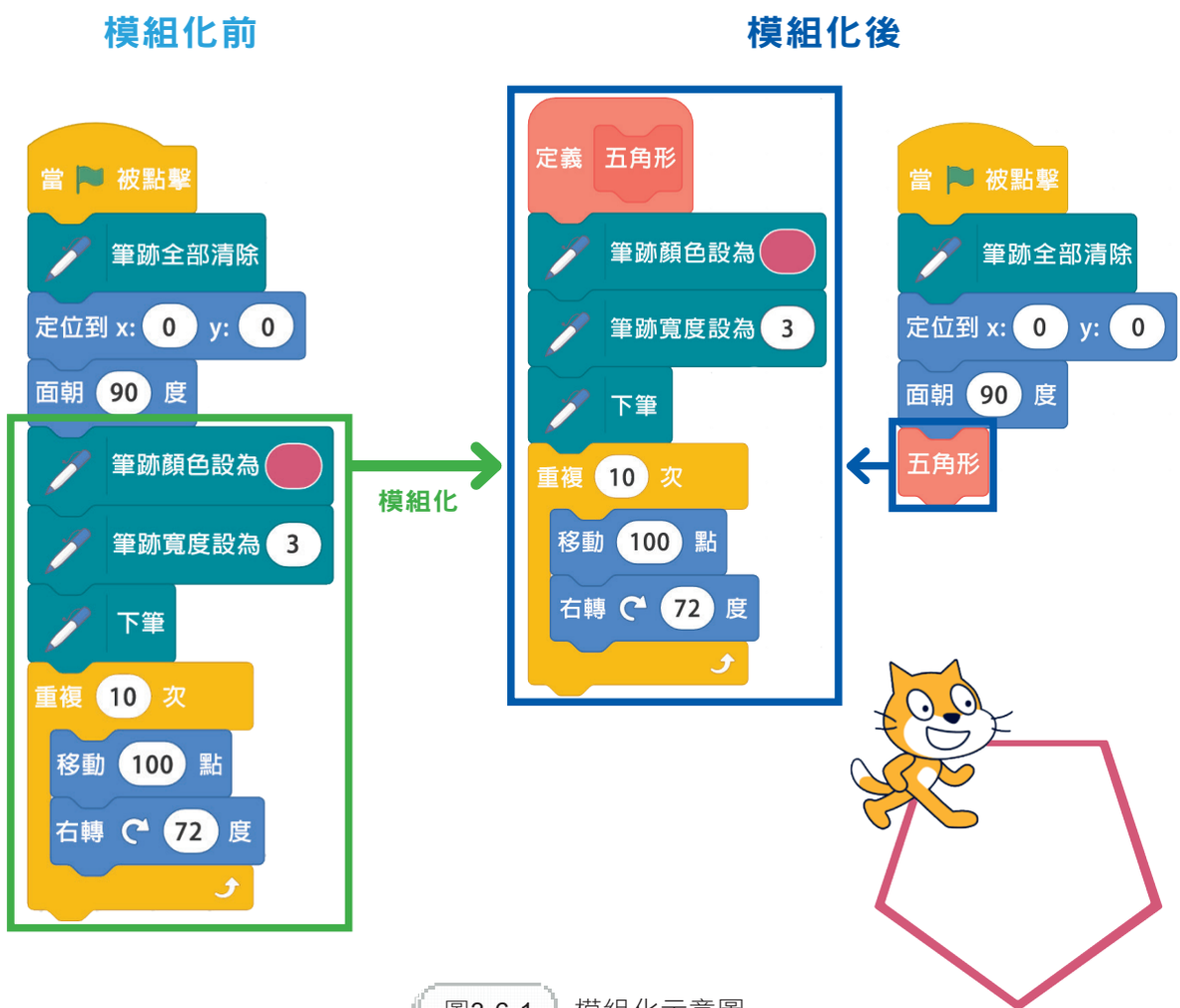


圖3-6-1 模組化示意圖

透過模組化可以達到下列優點：

01 提升開發效率

將開發程式切割成數個部分，交由程式設計師分工合作，以便縮短開發程式的時程。

02 程式重複使用

程式碼僅需開發一次，後續的相同問題可再重複使用此程式碼，節省人力資源與記憶體儲存程式的空間。

03 增加可讀性

透過不同功能模組的開發，明確地將各個模組負責的工作區分開，簡化原來繁冗的程式碼。

04 獨立性

如同汽車維修一樣，將有問題的零件模組替換掉即可。程式設計以模組化進行開發，使得功能相異的程式模組間相互獨立，降低程式除錯與維護的複雜度。

在本章開頭的「Hello, Python!」實作中，print() 就是一種內建的函式，意即將print()中的敘述進行輸出。每一種程式語言皆有其內建的**函式庫**，例如：Python當中計算次方的函式pow（a , b）。我們只要將原料（**參數**、**引數**）放進機器（函式），無須知道機器內部運作的過程，即可獲得正確的結果。而我們也可以使用自訂函式增加程式的彈性，更能增加程式的**再利用**、**模組化**、**分工**等特性。

在上述pow（a , b）的例子裡，**引數**是在呼叫函式時傳遞給函式的實際值；而**參數**是指在該函式定義中，用來接收引數的值。藉由後續將介紹的「自訂函式」範例，我們可以更清楚地看到兩者之間的關係及功能。

3-6-2 函式

常見的內建函式

Python內建函式是指在Python直譯器中內建的函式，不需要額外安裝套件或模組即可以使用。以下是Python常見的內建函式：

`print(*objects, sep=' ', end='\n')`	輸出函式
`input(prompt)`	輸入函式
`len(s)`	回傳參數的長度（元素個數）
`range(start, stop, step)`	產生一個整數序列
`type(x)`	回傳參數的型態
`int(x)`	將參數轉換為整數
`float(x)`	將參數轉換為浮點數
`str(x)`	將參數轉換為字符串
`bool(x)`	回傳一個布林值，即 True 或者 False
`abs(x)`	回傳一個數的絕對值，引數可以是整數、浮點數等
`max(arg1, arg2, *args)`	回傳參數中的最大值
`min(arg1, arg2, *args)`	回傳參數中的最小值
`sum(iterable)`	回傳參數中所有元素的和
`sorted(iterable)`	回傳參數排序後的序列

範例3-6-1 計算x之y次方值（使用內建函式）

```
x = int(input())
y = int(input())
print(pow(x, y))
```

呼叫內建的pow()函式，並回傳x^y的結果

測試執行

```
3
4
81
```

自訂函式

有回傳值的函式

```
def 函式名稱(參數1, 參數2,...):
    函式敘述
    ...
    return 回傳值

函式名稱(引數1, 引數2,...)
```

宣告函式語法

呼叫函式

範例

```
def Add(x, y):
    s = x + y
    return s

print(Add(5, 7))
```

Step1 傳遞引數 a, b 分別至參數 x, y

Step2 回傳運算結果

無回傳值的函式

```
def 函式名稱(參數1, 參數2,...):
    函式敘述
    ...

函式名稱(引數1, 引數2,...)
```

宣告函式語法

呼叫函式

範例

```
def Add(x, y):
    print(x + y)

Add(5, 7)
```

範例3-6-2 計算三角形面積（有回傳值的函式）

```
def area(x, y):
    return (x * y) / 2
a = float(input())
b = float(input())
print(area(a, b))
```

由參數x, y接收來自引數a, b的值

傳回面積的浮點數

Step1 傳遞引數 a, b 分別至參數 x, y

輸入底和高

呼叫area( )函式，並傳遞引數a, b

Step2 回傳運算結果

測試執行

```
5
8
20.0
```

科技即戰力

請設計一個程式，使用自訂函式，計算三角形面積（無回傳值的函式）。

實作解析3-6-1

```
def area(x, y):                    由參數x, y接收來自引數a, b的值
    print((x * y) / 2)             直接在函式中印出計算結果
a = int(input())
b = int(input())                   輸入底和高
area(a, b)                         呼叫area()函式，並傳遞引數
```

測試執行

```
3
5
7.5
```

NOTE

課堂檢測

( ) 1. 下列何者為Python的檔案名稱？
(A).py (B).cpp (C).exe (D).pyt。

( ) 2. 在Python語言中，若程式已進入迴圈敘述並正在執行，假如想讓程式直接跳回更新條件，並檢視是否再次進入迴圈敘述，則下列哪一保留字可以達成此目的？
(A)continue (B)break (C)back (D)goto。

( ) 3. 下列何者是將程式原始碼逐行翻譯為機器碼，且同時直接執行後再繼續翻譯下一段程式原始碼？
(A)編譯程式 (B)組譯程式 (C)直譯程式 (D)載入程式。

( ) 4. 下列哪一項錯誤可透過直譯器發現錯誤位置，進而修正錯誤的程式碼？
(A)語法錯誤 (B)語意錯誤 (C)執行階段錯誤 (D)直譯錯誤。

( ) 5. 若將原先算式 2*x+4 打成 2*x-4 直譯並執行後，會產生何種錯誤？
(A)語法錯誤 (B)語意錯誤 (C)執行階段錯誤 (D)直譯錯誤。

( ) 6. 在執行階段中，若使用者將分母輸入為0，會產生何種錯誤？
(A)語法錯誤 (B)語意錯誤 (C)直譯錯誤 (D)執行階段錯誤。

( ) 7. Python程式的輸出函數名稱為何？
(A)cout (B)print (C)printf (D)export。

( ) 8. 下列變數命名，何者錯誤？
(A)3_sub (B)num (C)Num_3 (D)NUM。

( ) 9. 下列何者為Python中的保留字？
(A)sum (B)add (C)int1 (D)not。

( ) 10. 請問右列程式碼之輸出為何？
(A)9
(B)5
(C)7
(D)無窮迴圈。

```
n = 1
while n != 8:
    n = n + 2
print(n)
```

( ) 11. 將數學不等式 80＜x＜90 表示成Python的條件判斷式：(80<x□x<90)。□內應填入下列何者？
(A)and (B)or (C)not (D)<。

( ) 12. 要判斷 n 為奇偶數時，下列敘述的空格應該填入下列何者？
(A)n/2 == 0
(B)n%2 == 0
(C)n/2 != 0
(D)n%2 != 0。

```
if _____:
    print("偶數")
else:
    print("奇數")
```

( ) 13. 執行右列Python程式片段，試問輸出結果為何？

```
a = 5
b = 6
if a >= 5:
    b = 5
if b >= 6:
    a = b
print(b, a)
```

(A)5 5
(B)5 6
(C)6 5
(D)6 6。

( ) 14. 執行右列Python程式片段，試問 z 值結果為何？

```
x = 6
y = 7
z = 8
if x >= y:
    z = x - y
else:
    z = y - x
```

(A)1
(B)-1
(C)2
(D)-2。

( ) 15. 請問右列程式碼中，sum值為？

```
sum = 0
for i in range(1, 11, 2):
    sum = sum + i
```

(A)55
(B)25
(C)26
(D)56。

( ) 16. 若 m = 0，執行下列程式碼後，m 值為何？

```
m = 0
for i in range(7):
    for j in range(7):
        m = m + 1
```

(A)25
(B)36
(C)49
(D)64。

( ) 17. 請問右列程式碼之輸出為何？

```
a = 10
b = 20
a = b
b = a
print(a + b)
```

(A)30
(B)20
(C)40
(D)50。

( ) 18. a = 3，b = 7，則Python運算式 b // a 的值為何？

(A)1
(B)2.3
(C)2.33
(D)2。

( ) 19. 請問右列程式碼之輸出為何？

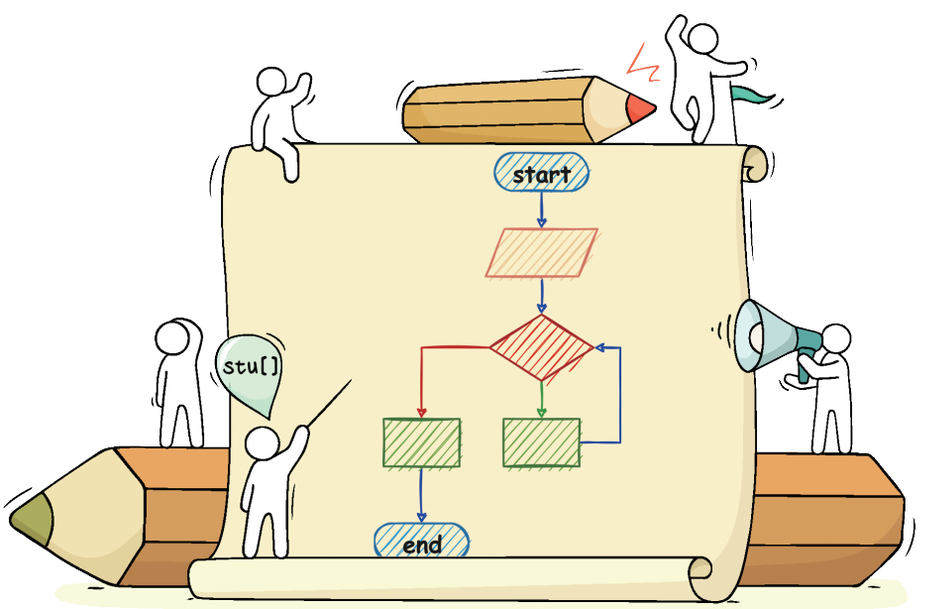
```
n = 567
a = n % 10
n = a // 2
b = 10 % n
print(b)
```

(A)1
(B)2
(C)3
(D)4。

chapter

04 串列資料結構的程式設計實作

在資料結構的單元中，曾提到陣列是運用連續的表格記錄相同型態的資料，可以使得資料的記錄或查詢更為簡便，如班級點名單、班級成績單、班級身高體重紀錄表等資料。在這個章節中，我們以Python程式設計中的串列，來實作有關陣列中資料的儲存與讀取。



4-1 一維串列實作

從Chapter 03的變數概念中可以得知，當班上導師想要記錄30位同學的體重，可能需要宣告30個變數，以記錄每位同學的體重資料，如圖4-1-1。在程式設計中，宣告數量龐大的變數顯然是沒有效率的，而且會占用程式碼整體的版面，因此我們可以選擇專門處理大量相同型態的資料結構——**串列**，取代逐一宣告變數的過程，如下表4-1-1。

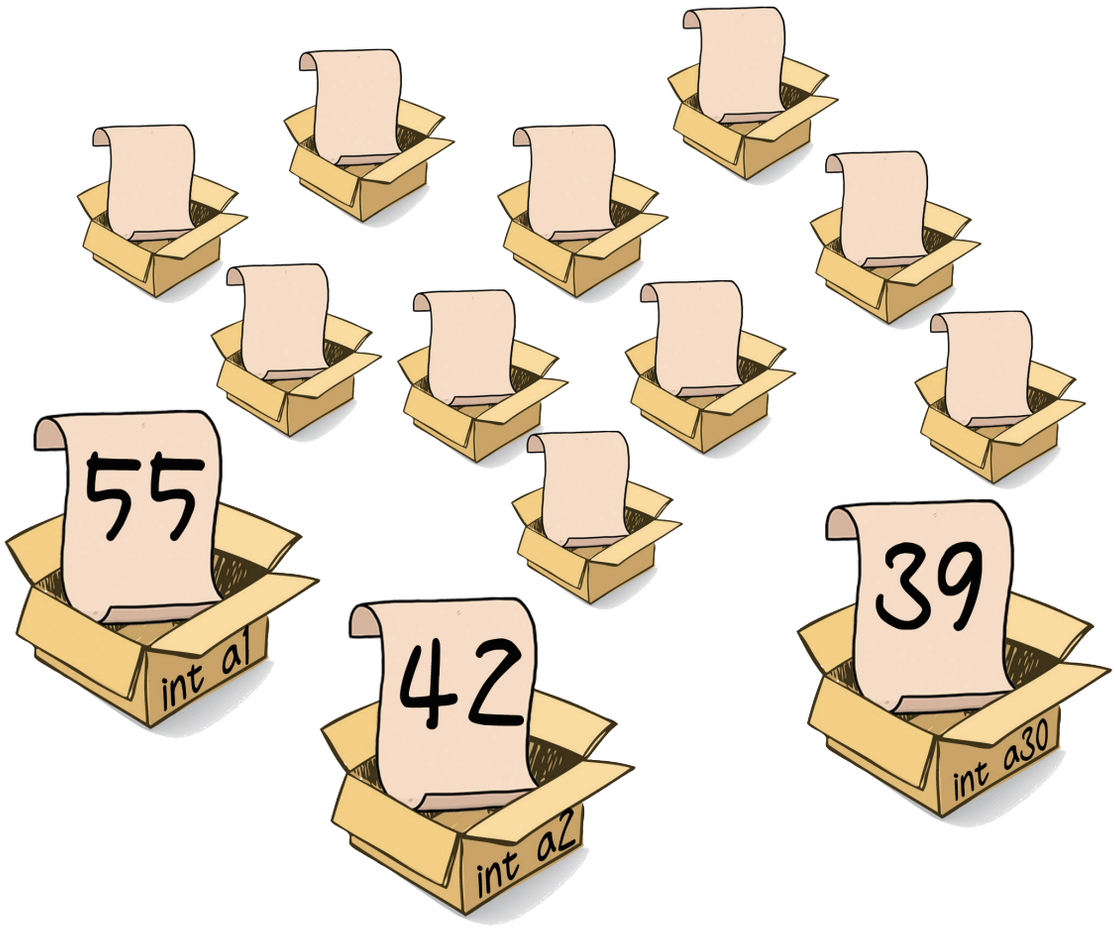


圖4-1-1 一個變數僅能儲存一筆資料

表 4-1-1 同學的體重與座號以串列結構呈現
（此處僅為示例，Python語言的串列索引值是由0開始）

座號（索引值）	1	2	……	30
體重（元素）	55	42	……	39

在Python中，串列的基本概念如下：

1. **串列**可以儲存多個不同型態的資料集合（如前頁體重紀錄圖表），亦可為空集合。可以透過**索引值**（Index，如座號）存取在串列中的每一元素（如學生體重）。

2. 允許的資料型態如**整數**、**浮點數**、**字串**等。

3. 賦予一維串列的值

(1) 相同資料型態的一維串列

```
a = [53, 47, 65]                    a[0] = 53, a[1] = 47, a[2]=65
b = [3.2, 5.6, 8.3, 99.2]           b[0] = 3.2, b[1] = 5.6, b[2] = 8.3, b[3] = 99.2
c = ['Tom', 'Nico', 'Emily']        c[0] = 'Tom', c[1] = 'Nico', c[2] = 'Emily'
```

串列中的每一筆資料稱為元素，並以索引值作為存取元素的依據。如上述串列**a**的**a[0] = 53**，**a[1] = 47**，**a[2] = 65**。

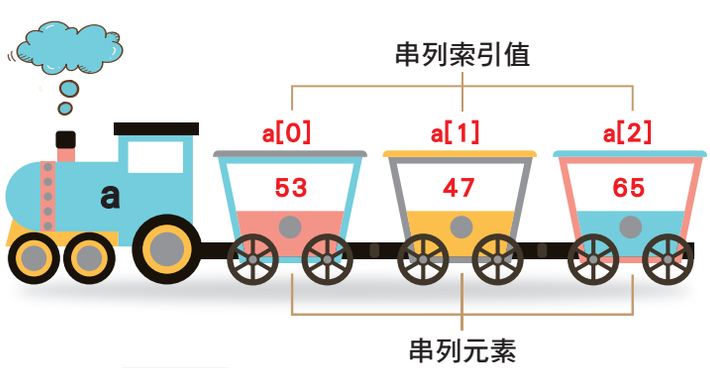


圖4-1-2 以串列型態儲存連續的資料

串列的賦值是以成對的中括號[ ]將元素放入當中，並以逗點作為元素間的間隔。

(2) 不同資料型態的一維串列

```
d = [10, 'Tom', 88.7]    d[0] = 10, d[1] = 'Tom', d[2] = 88.7
```

4. 若串列存在 n 個元素，索引值是從**0**開始，則最後一個元素索引值是n-1。如上述串列**d**的 **d[0] = 10, d[1] = 'Tom', d[2] = 88.7**。亦可用-1、-2的索引值代表從最後數來第1個元素、從最後數來第2個元素。如上述串列**d**的**d[-3] = 10, d[-2] = 'Tom', d[-1] = 88.7**。

5. 串列會依據**元素個數**與所屬**資料型態**，給予**連續的**記憶體空間儲存。其概念如圖4-1-3。

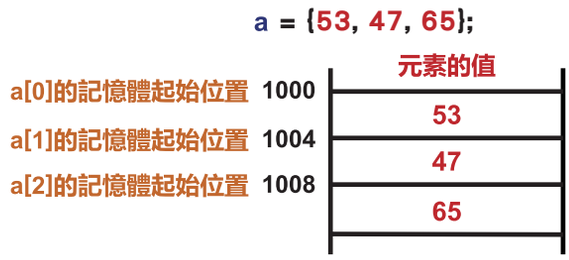


圖4-1-3 串列在記憶體中儲存的方式

6. 若存取超出串列範圍的索引值，直譯器會發出超過存取範圍的錯誤訊息，如下所示。

```
a = [53, 47, 65]
print(a[3])
```

超出範圍的存取

測試執行

錯誤訊息

```
IndexError: list index out of range
```

7. 存取串列元素：

(1) 可以使用print(串列名稱)輸出全部元素。

```
a = [53, 47, 65]
print(a)
```

測試執行

```
[53, 47, 65]
```

(2) 可以使用for迴圈存取個別元素。

```
a = [53, 47, 65]
for i in range(len(a)):
    print(a[i], end=" ")
```

測試執行

```
53 47 65
```

(3) 可以使用append(元素)將元素依序加入空串列。

範例 4-1-1

Q 將下表超額比序的科目次序與分數運用串列儲存後，再以索引值輸出串列分數。

科目次序	科目一	科目二	科目三	科目四	科目五
分數	99	98	97	96	95

```
a = [99, 98, 97, 96, 95]
print("科目一 =", a[0])
print("科目二 =", a[1])
print("科目三 =", a[2])
print("科目四 =", a[3])
print("科目五 =", a[4])
```

測試執行

```
科目一 = 99
科目二 = 98
科目三 = 97
科目四 = 96
科目五 = 95
```

範例4-1-2 運用 for 迴圈輸入後並輸出串列

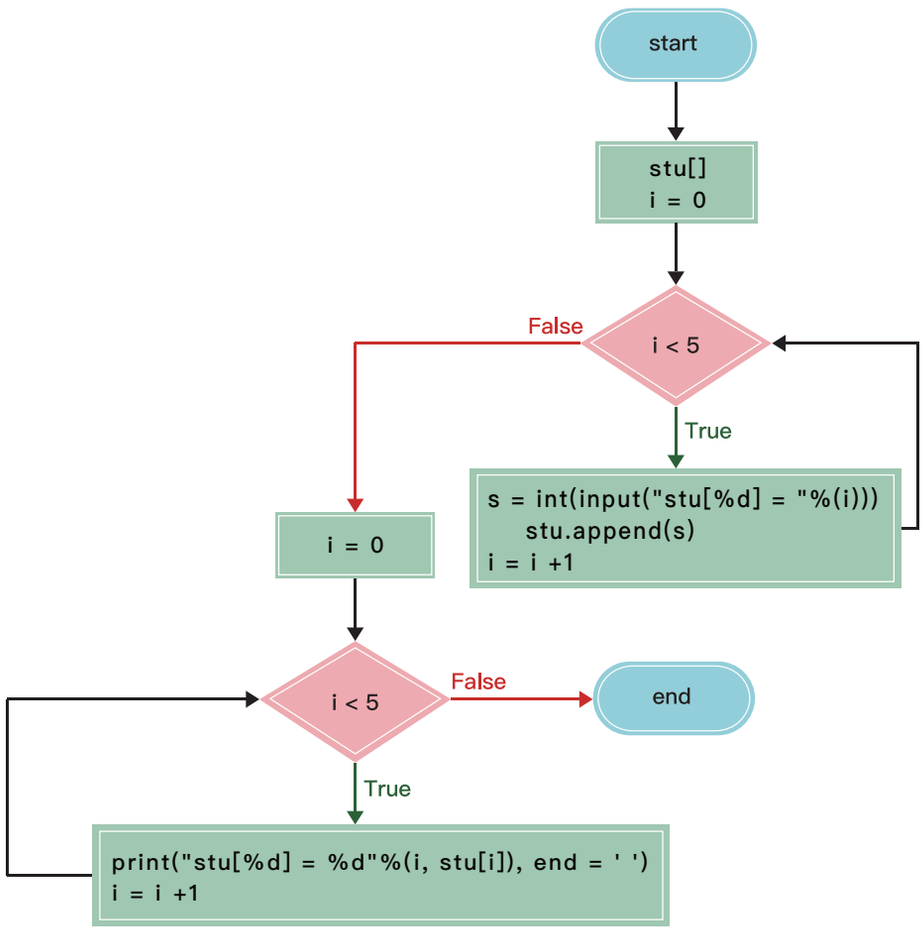


圖4-1-4 範例流程圖

```
stu = []
for i in range(5):
    s = int(input("stu[%d] = "%(i)))
    stu.append(s)

for i in range(5):
    print("stu[%d] = %d"%(i, stu[i]), end = ' ')
```

測試執行

```
stu[0] = 1
stu[1] = 2
stu[2] = 3
stu[3] = 4
stu[4] = 5
stu[0] = 1 stu[1] = 2 stu[2] = 3 stu[3] = 4 stu[4] = 5
```

科技即戰力

請設計一個程式，求串列元素的平均值。

實作解析4-1-1

```
stu = [55, 66, 77, 88, 99]
sum = 0
for i in range(5):
    sum = sum + stu[i]
print("average = ", sum / 5.0)
```

測試執行

```
average = 77.0
```

4-2 二維串列實作

以一維串列來說，若要記錄30位同學的8科成績（表4-2-1），則需要宣告30個一維串列儲存資料，這樣一來又回到了大量宣告的問題。因此為了增加撰寫程式碼的效率，我們可以使用**二維串列**方式替代。

表 4-2-1 學生與成績資料

座號／科目	國文	英文	…	資訊科技
1	98	89	…	96
2	88	82	…	92
⋮	⋮	⋮	…	⋮
29	62	94	…	95
30	96	67	…	96

二維串列較一維串列多增加了一個索引值，例如：下列範例4-2-1有一個3列4行的二維串列a，同樣也是從a[0][0]為開始的第一個元素，而最後一個元素為a[2][3]，前面的索引值2代表第2**列**（橫向），索引值3代表第3**行**（縱向），如下所示。

範例 4-2-1

Q 將下列表格中三位體操選手的四項表演分數以二維串列儲存

項次／選手編號	項目一	項目二	項目三	項目四
1	88	86	85	84
2	80	83	89	92
3	90	93	91	81

宣告串列並給予初始值：

```
a = [[88, 86, 85, 84], [80, 83, 89, 92], [90, 93, 91, 81]]
```

範例 4-2-2

Q 運用雙重迴圈輸出下列二維串列

1	2	3
4	5	6

```
a = [[1, 2, 3], [4, 5, 6]]
for i in range(2):
    for j in range(3):
        print(a[i][j], end = " ")
    print()
```

外迴圈：`for i in range(2):`
內迴圈：`for j in range(3):`
控制列
控制行

測試執行

```
1 2 3
4 5 6
```

下表為串列化的表格，運用 for 迴圈以列為主，依序輸出元素資料。

	第 0 行	第 1 行	第 2 行
第0列	a[0][0] = 1	a[0][1] = 2	a[0][2] = 3
第1列	a[1][0] = 4	a[1][1] = 5	a[1][2] = 6

4-3 串列的應用

「串列」是Python程式語言的基本資料結構之一，我們可以運用其**循序**（儲存在記憶體中連續的空間）的特性，實現更多的應用。而在Chapter 01 所提到的資料結構，多數可以應用串列來實現，以下我們會以圖例與程式實作的方式，說明如何運用串列方式實現各種資料結構的應用。

4-3-1 運用串列呈現堆疊的概念

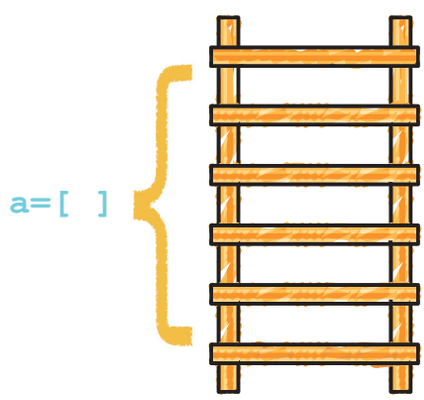
Chapter 01我們曾提到堆疊裡的新增（push）資料與取出（pop）資料兩項動作，在Python實作上，可以運用append(參數)進行串列中元素的新增，以及運用pop(參數)進行元素的取出。

其應用如Chapter 01提過的瀏覽器「回到上一頁」功能，可以將每次瀏覽過的網頁新增至堆疊中，而當點選「回到上一頁」按鈕時，會從堆疊最上層取出網頁；而在文件的編輯中，「還原」的功能也是運用堆疊將編輯的事件新增至堆疊中，待按下「還原」時，則從堆疊中取出上一個步驟進行還原。

接下來，我們以下列堆疊的範例和程式碼，實現串列的應用。

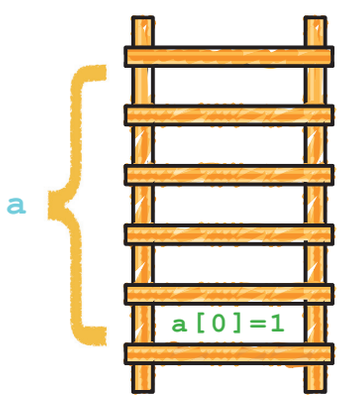
STEP 01

新增一個空的堆疊（串列）。

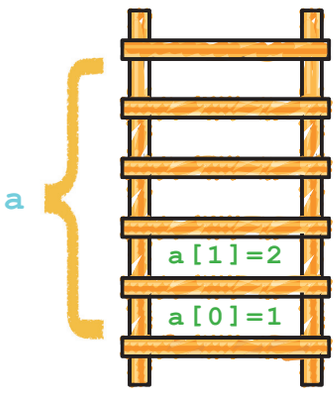


STEP 02

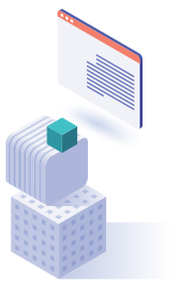
在堆疊中放入第一個資料 1，a.append(1)，a=[1]。

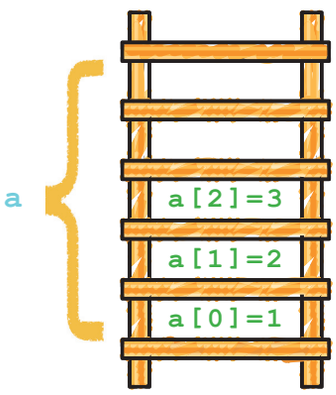


STEP 03 在堆疊中放入第二個資料 2，a.append(2)，a=[1, 2]。

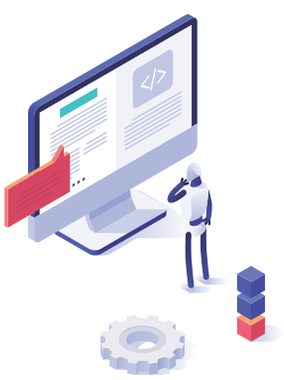


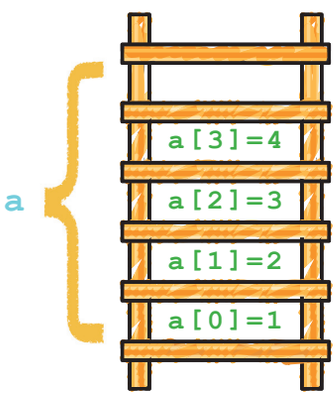
STEP 04 在堆疊中放入第三個資料 3，a.append(3)，a=[1, 2, 3]。



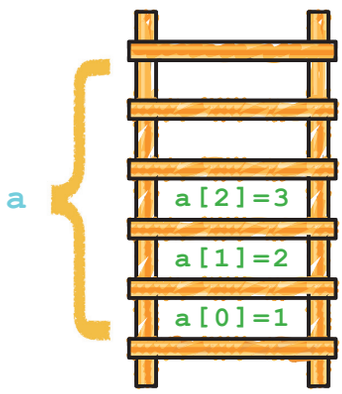


STEP 05 在堆疊中放入第四個資料 4，a.append(4)，a=[1, 2, 3, 4]。





STEP 06 在堆疊中取出最頂端的元素，a.pop，a=[1, 2, 3]。



運用串列實作堆疊的程式設計

```
a = []
a.append(1)
print(a)
a.append(2)
print(a)
a.append(3)
print(a)
a.append(4)
print(a)
a.pop()
print(a)
```

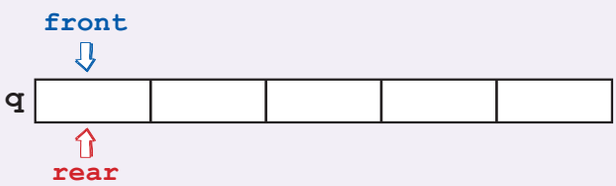
測試執行

```
[1]
[1, 2]
[1, 2, 3]
[1, 2, 3, 4]
[1, 2, 3]
```

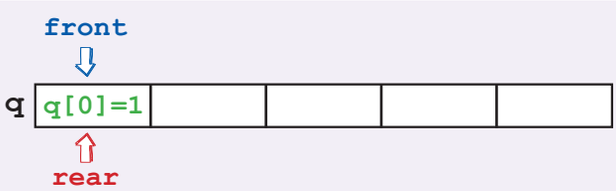
4-3-2 運用串列呈現佇列的概念

在Chapter 01有提到，佇列加入資料是從佇列的**尾端**進行，而若要刪除資料則從**前端**進行刪除，是種**先進先出**的資料結構。其可應用在系統列印排程，將最先進入的列印工作依序輸出至印表機；而客服系統的服務次序也可以運用佇列實現先到先服務的等候排程。接下來我們以下列的佇列的範例和程式碼，實作串列的應用。

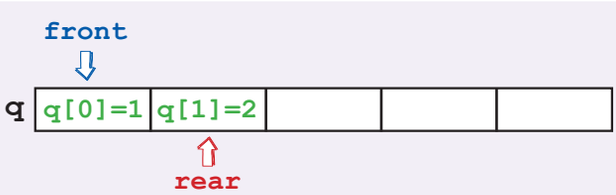
STEP 01 新增一個空的佇列（串列），q=[ ]。



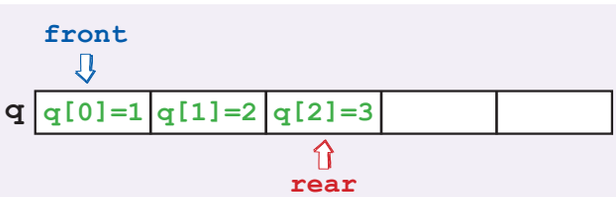
STEP 02 佇列 q 加入第一個資料 1，
a.append(1)。
a=[1]。



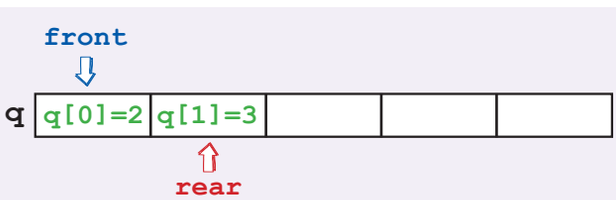
STEP 03 佇列 q 加入第二個資料 2，
a.append(2)。
a=[1, 2]。



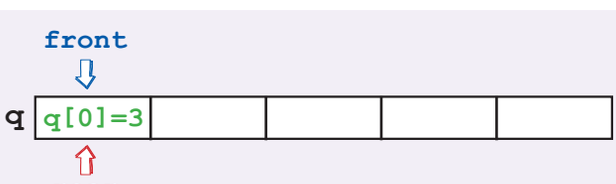
STEP 04 佇列 q 加入第三個資料 3，
a.append(3)。
a=[1, 2, 3]



STEP 05 從佇列 q 刪除最前面的元素，
a.pop(0)。
a=[2, 3]。



STEP 06 從佇列 q 刪除最前面的元素，
a.pop(0)。
a=[3]。



運用串列實作佇列的程式設計

```
a = []
a.append(1)
print(a)
a.append(2)
print(a)
a.append(3)
print(a)
a.pop(0)
print(a)
a.pop(0)
print(a)
```

測試執行

```
[1]
[1, 2]
[1, 2, 3]
[2, 3]
[3]
```

補給站

以串列儲存二元樹的節點值

在資料結構中，我們可以運用串列的方式體現**二元樹**的儲存，透過串列元素儲存每個節點值，並以各元素的索引值，推斷彼此是否互為**父節點**、**子節點**的關係。而在為二元樹的每個節點進行編號（節點儲存在串列中的索引值）的過程中，倘若在同一層有空節點，則必須填補此空節點（此範例空節點以 -1 表示），如此才能還原二元樹的樹狀結構圖型。

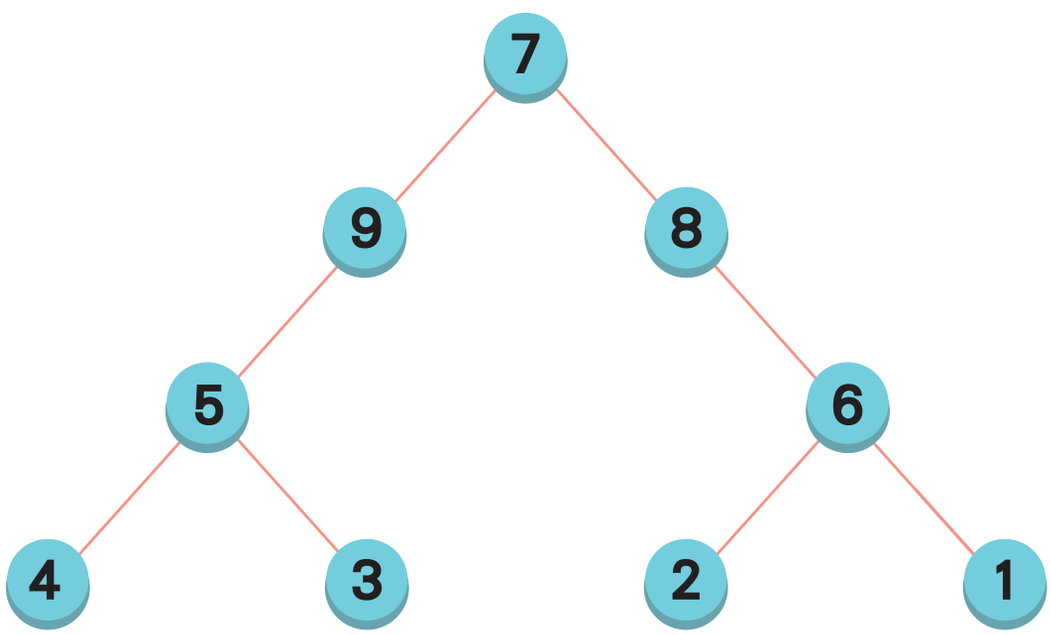


圖4-3-3 範例二元樹

二元樹（圖 4-3-3）以串列儲存的示意圖如下圖 4-3-4，先將同一層所缺的節點補上 -1，如下圖的綠色節點；再從 1 開始，由樹根節點依序編號（括號內對應的是右側串列的索引值），而後再將各節點值依串列索引值填入串列當中。

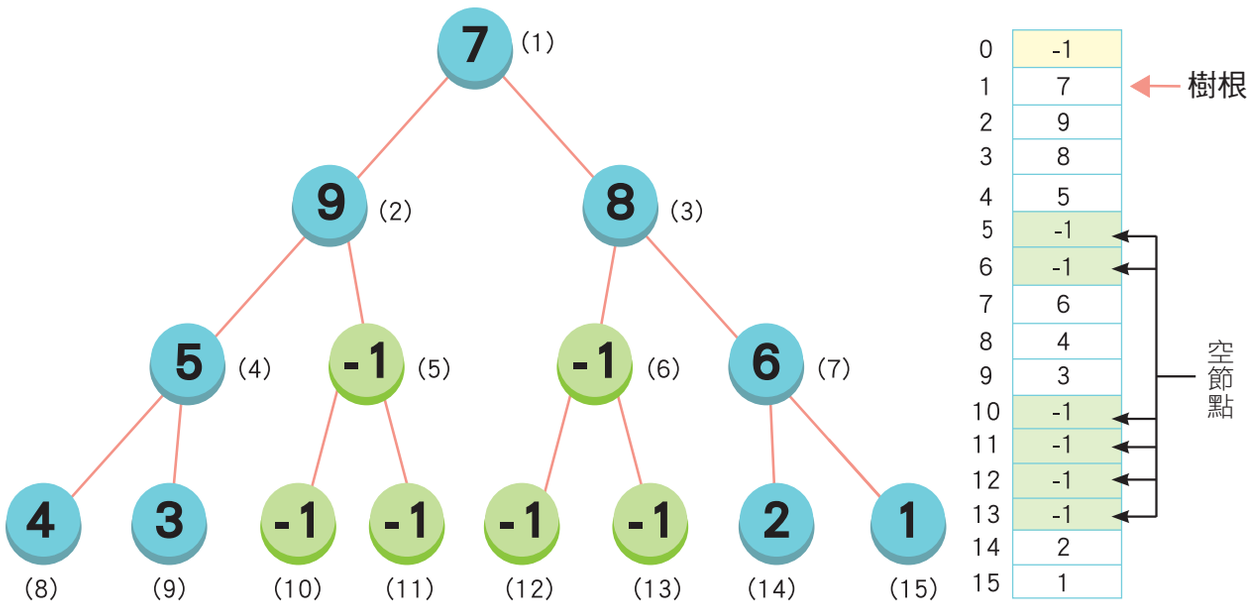
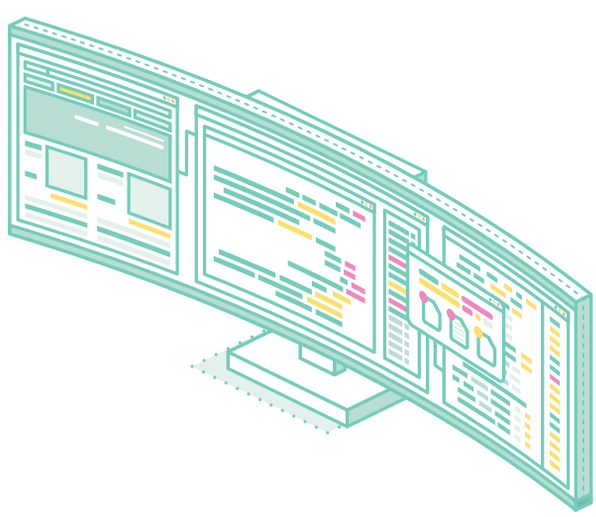


圖4-3-4 以串列儲存二元樹的節點值

在串列元素與樹的節點的相對位置關係，可以藉由以下方式判斷：

若父節點為 i，則左子節點索引值為 2 × i，右子節點索引值為 2 × i + 1。透過上述的判斷，即可將串列還原回二元樹的資料結構圖型。



課堂檢測

( ) 1. 在Python中宣告串列a=[1，2，3，4，5]，試問串列a的最後一個元素為何？
(A)a[3] (B)a[4] (C)a[5] (D)a[6]。

( ) 2. 承上題，試問串列a中的第一個元素為何？
(A)a[0] (B)a[1] (C)a[2] (D)a[5]。

( ) 3. 若A[1]、A[2]，和A[3]分別為串列A的三個元素，下列哪個程式片段可以將A[1]和A[2]的內容交換？
(A)A[1] = A[2]; A[2] = A[1];
(B)A[3] = A[1]; A[1] = A[2]; A[2] = A[3];
(C)A[2] = A[1]; A[3] = A[2]; A[1] = A[3];
(D)以上皆可。

( ) 4. 在Python中宣告串列a=['Tom', 'Peter', 'Wendy', 'Mary', 'Tim']，試問a[5]為何？
(A)Tom (B)Tim (C)Mary (D)超出串列範圍。

( ) 5. 在右列Python程式中，請問最後輸出的值為何？
(A)8 (B)10
(C)12 (D)不可預期。

```
a = [2, 4, 6, 8, 10]
s = a[1] + a[3]
print(s)
```

( ) 6. 在右列Python程式中，請問最後輸出的值為何？
(A)12 (B)14
(C)16 (D)不可預期。

```
a = [2, 4, 6, 8, 10]
s = a[3] + a[6]
print(s)
```

( ) 7. 在Python中宣告二維串列a = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]，試問a[1][1]=？
(A)5 (B)1 (C)7 (D)4。

( ) 8. 在Python中，宣告二維串列a = [[0, 1], [2, 3], [4, 5]]，試問a[2][1]=？
(A)2 (B)3 (C)4 (D)5。

( ) 9. 在Python中宣告二維串列a[5][6]，試問在雙重迴圈的輸出中，a[3][4]的下一個元素為何？
(A)a[4][5] (B)a[3][5] (C)a[4][1] (D)a[4][4]。

( ) 10. 下列有關串列的敘述，何者正確？
(A)可同時具有不同資料型態，且分散存在記憶體空間
(B)具有相同資料型態，且分散存在記憶體空間
(C)具有相同資料型態，且存在相鄰的記憶體空間
(D)可同時具有不同資料型態，且存在相鄰的記憶體空間。

( ) 11. 在Python程式中，若想要加總某次段考8個科目的總分，透過串列實作的程式碼如下，請問空格應填入何值？
(A)6 (B)7 (C)8 (D)9。

```
sum = 0
for i in range(___):
    sum = sum + a[i]
```

( ) 12. 執行下列 Python 程式結果為何？
(A)['a', 'b'] (B)['b', 'c'] (C)['a', 'c'] (D)['c', 'd']。

```
a = ['a', 'b', 'c']
a.append('d')
a.pop()
a.pop(0)
print(a)
```

( ) 13. 有一 Python 程式片段 a = ['a','b','c','d']，試問 a[-2] 的值為何？
(A)b (B)c (C)d (D)超出範圍。

( ) 14. 下列程式片段執行過程的輸出為何？
(A)44 (B)52 (C)54 (D)63。

```
a = [0, 0, 0, 0, 0, 0]
a[0] = 3
a[1] = 7
for i in range(4):
    a[i + 2] = a[i] + a[i + 1]
print(a[5])
```

( ) 15. 下列程式片段執行過程的輸出為何？ (A)44 (B)52 (C)54 (D)63。

```
a = []
for i in range(10):
    a.append(i)
sum = 0
for i in range(1, 9):
    sum = sum - a[i-1] + a[i] + a[i+1]
print(sum)
```

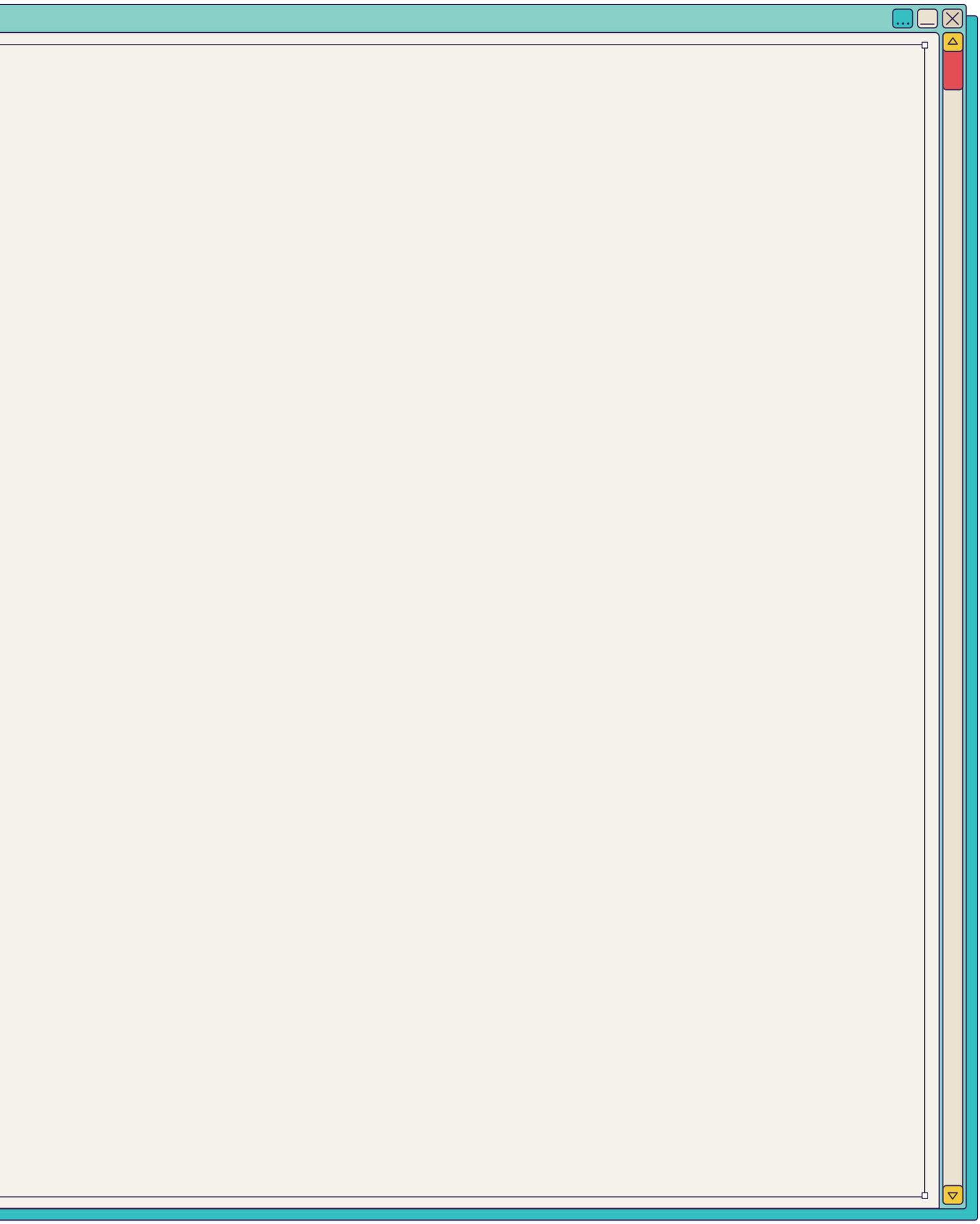
( ) 16. 下列程式片段執行過程的輸出為何？ (A)18 (B)15 (C)12 (D)10。

```
a = []
for i in range(10):
    a.append(10 - i)
print(a[2] + a[5] + a[8])
```

( ) 17. 定義 a 為一 n 個元素的串列，串列元素的索引值為 0 至 n-1。若要將串列中的 a[0] 的元素移到 a[n-1]，下列程式片段空白處該填入哪一項運算式？
(A)n+1 (B)n (C)n-1 (D)n-2。

```
i = 0
n = 5
a = [1, 2, 3, 4, 5]
while i <= _______:
    hold = a[i]
    a[i] = a[i + 1]
    a[i + 1] = hold
    i = i + 1
```

NOTE



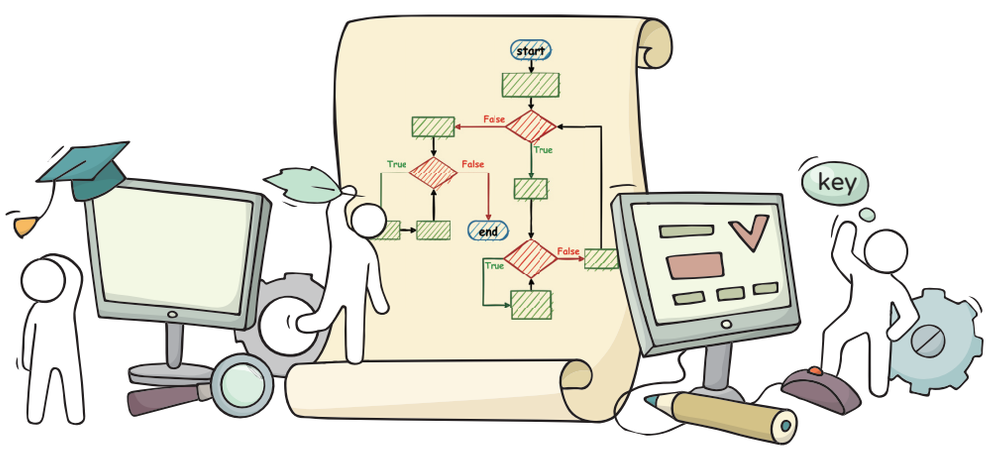
chapter 05

重要演算法的程式設計實作

排序是將一群資料依據特定規則進行重新排列的演算法，通常會以資料的大小依序排列。排序後的資料能讓使用者更有效率地進行分析、閱讀與搜尋。例如：將全校成績進行排序等。

而搜尋是為了在一群資料中，查找某項特定之值或資料而使用的演算法。例如：搜尋引擎可協助我們搜尋想查找的網頁等。

兩者的邏輯概念也常應用在相關的程式演算法中。因此本章節以Python程式語言實作這些常見且重要的演算法，我們將依序介紹：氣泡排序法、選擇排序法、插入排序法、循序搜尋法、二分搜尋法。



5-1 排序法實作

5-1-1 氣泡排序法（Bubble Sort）實作

氣泡排序法為典型的排序方法之一，藉由模仿氣泡由水底上升、浮至水面的現象，發展出類似水中氣泡移動的排序方式。氣泡排序法主要可以透過串列與重複結構實作，每次透過相鄰兩元素進行兩兩相比，若是排序順序錯置，則進行兩數互換。每次大循環的排序結果皆會將該次循環的最大數排列到最後方（由小至大排列），也是該數的最終定位，如同氣泡浮至水面上。重複結構的外迴圈控制大循環，內迴圈控制每次兩兩相比的小回合，反覆進行排序，直到完成所有元素的排序定位為止。

相同 n 筆資料的比較次數不會因為起初順序的不同而有差異，例如：第一次循環 (n - 1) 回合、第二次循環 (n - 2) 回合，直到第 (n - 1) 次循環的 1 回合，比較次數隨著循環數遞減。因此，若有 n 筆資料進行氣泡排序，則所需的次數為 $n(n - 1) / 2$，因此其平均時間複雜度為 $O(n^2)$ 。

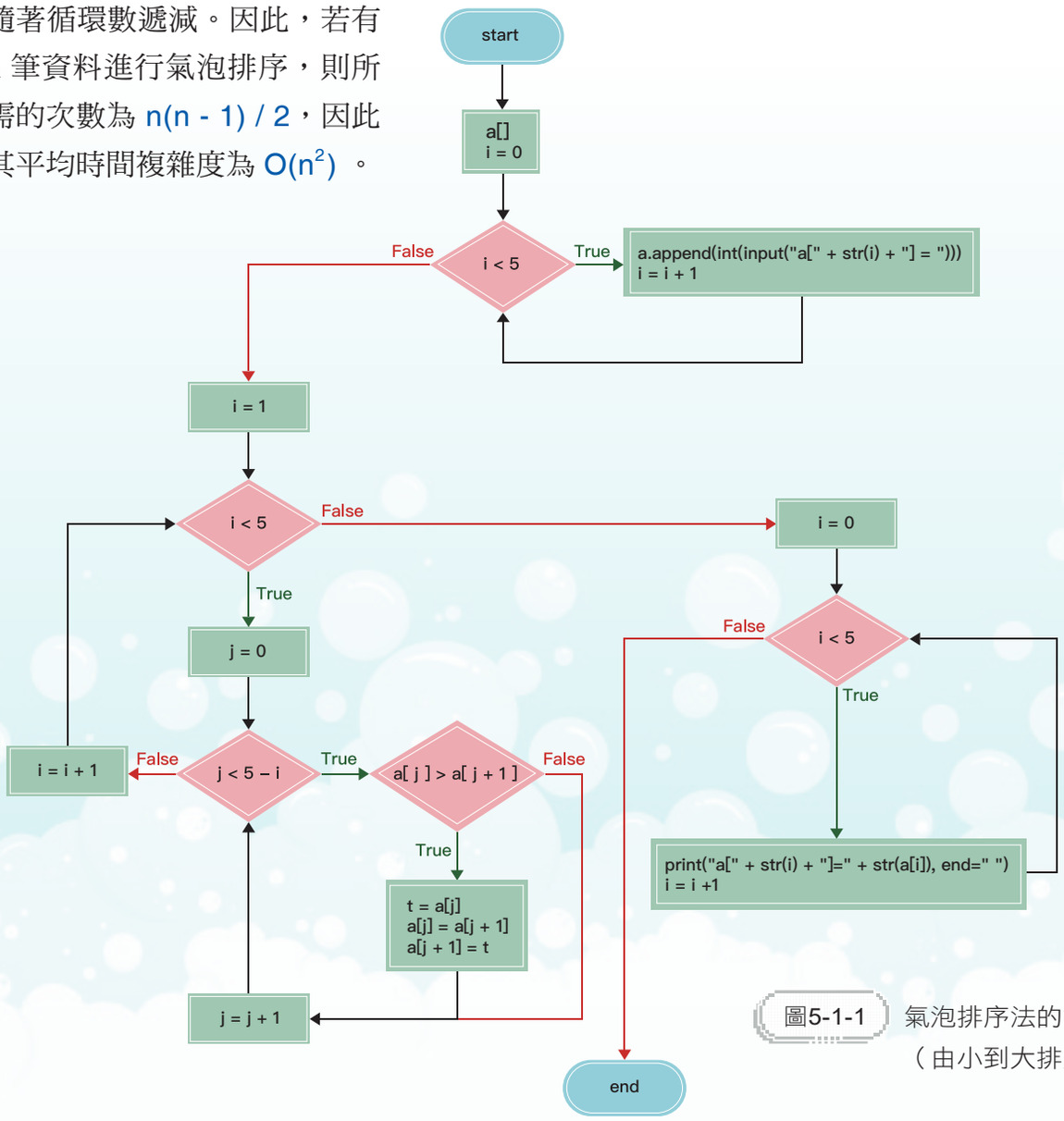


圖5-1-1 氣泡排序法的範例流程圖（由小到大排序）

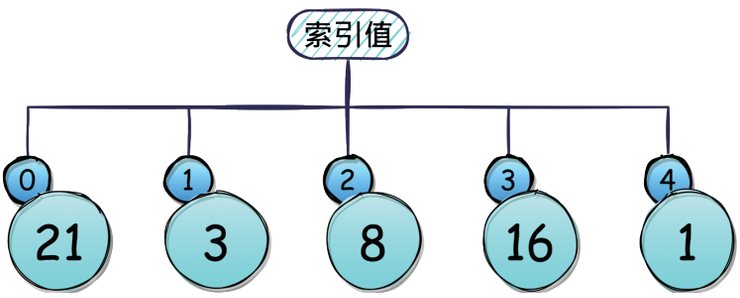
氣泡排序法的程式設計（由小到大排序）

```
a = []
for i in range(5):
    a.append(int(input("a[" + str(i) + "] = "))) 先將串列資料依序輸入

for i in range(1, 5):          外迴圈控制循環數
    for j in range(0, 5 - i)   每回合內兩兩互相比較
        if a[j] > a[j + 1]:    兩數交換條件
            t = a[j]
            a[j] = a[j + 1]
            a[j + 1] = t

for i in range(5):
    print("a[" + str(i) + "]=" + str(a[i]), end=" ") 將最後排序結果輸出
```

測試執行

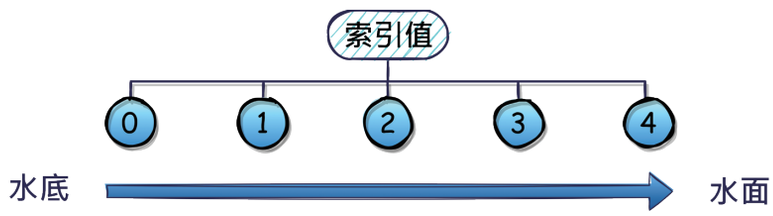
```
a[0] = 21
a[1] = 3
a[2] = 8
a[3] = 16
a[4] = 1
a[0]=1 a[1]=3 a[2]=8 a[3]=16 a[4]=21
```

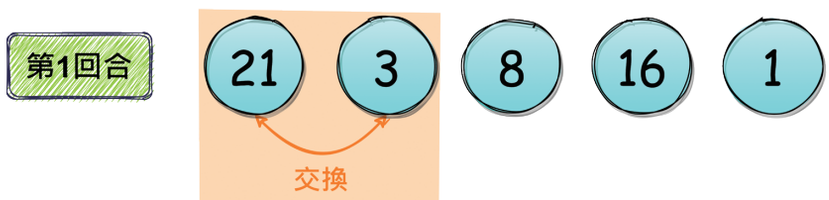
原始資料



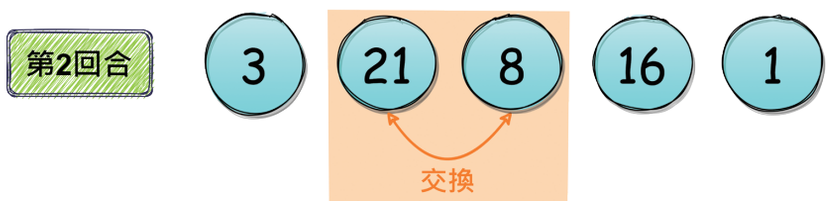
第一次循環

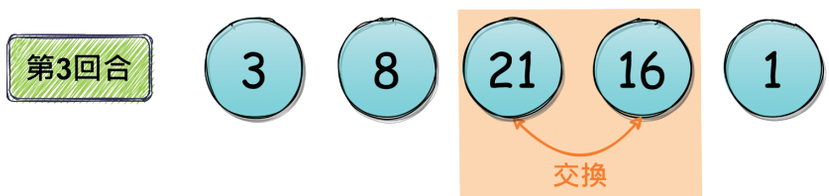
外迴圈控制循環數，若有 n 筆資料，則必須有 (n - 1) 次大循環。



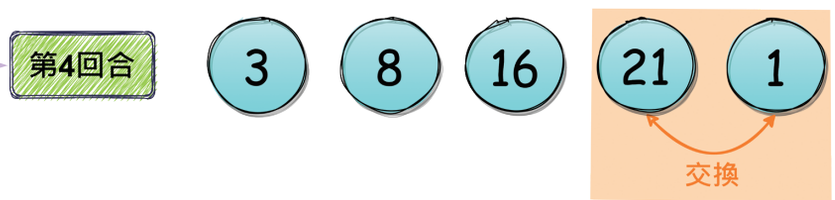


在每一回合，當左側的值較右側的大，則進行兩數交換；反之，則不進行互換，繼續進行下一回合。



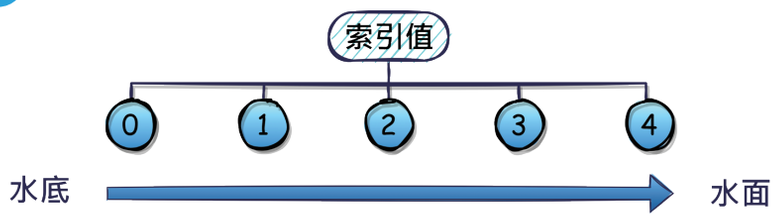


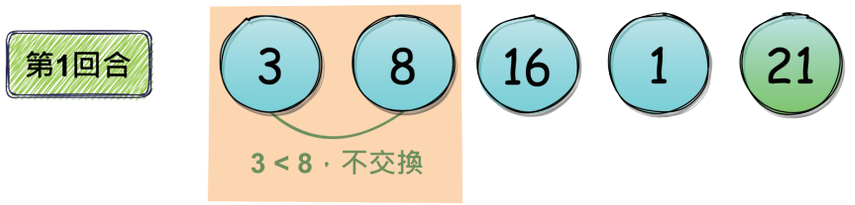
內迴圈控制回合數，若有 n 筆資料進行兩兩相比，共需 (n - 1) 個回合。



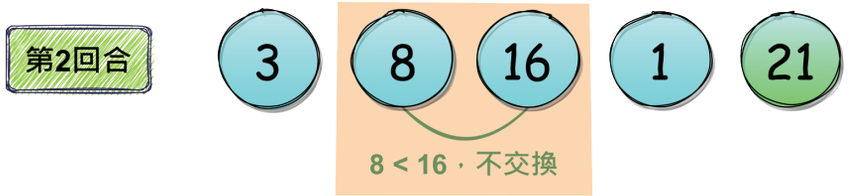
第二次循環

外迴圈控制循環數，若有 n 筆資料，則必須有 (n - 1) 次大循環。

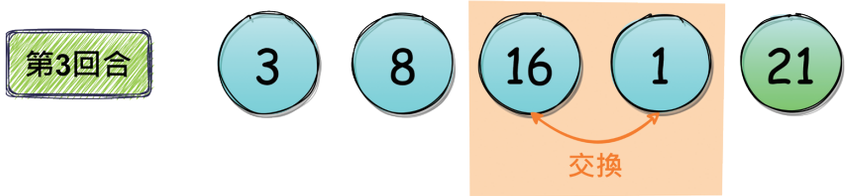




每次大循環會將該次的最大數排序至定位（最後方），因此每一次的循環，會較上次循環的資料數減 1。

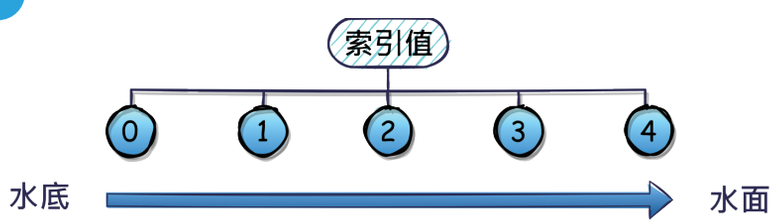


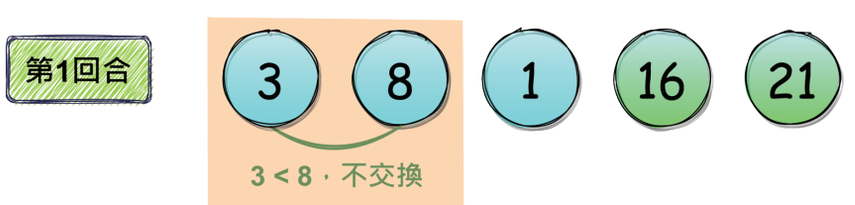
由於每一次的循環，會較上次循環的資料數減 1，因此比較的回合數也會減少 1 次。

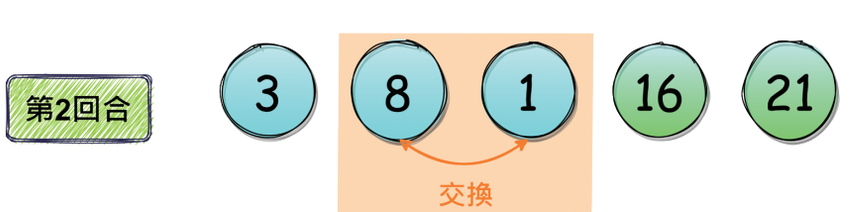


第三次循環

外迴圈控制循環數，若有 n 筆資料，則必須有 (n - 1) 次大循環。

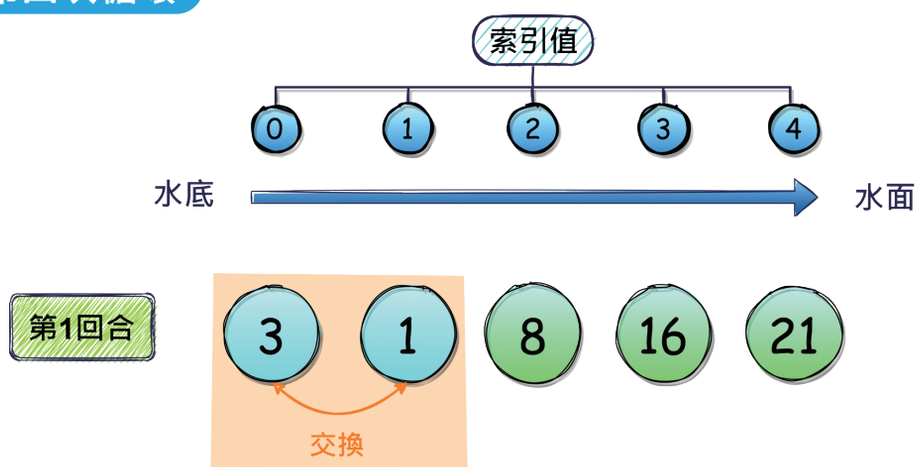




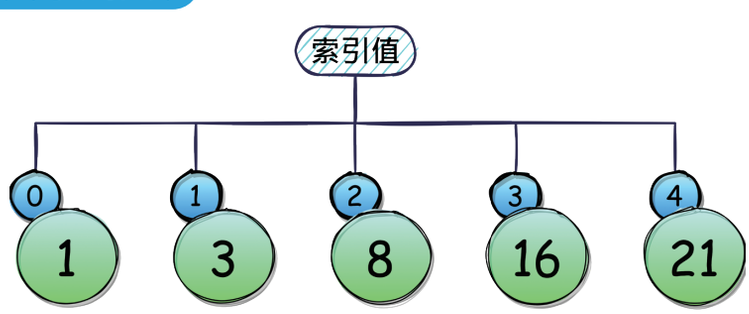


第四次循環

外迴圈控制循環數，若有 n 筆資料，則必須有 (n - 1) 次大循環。



排序結果（由小到大）



科技即戰力

請同學用氣泡排序方式，將數列：65、56、34、79、21由小到大、依照順序排列，並將過程記錄下來。

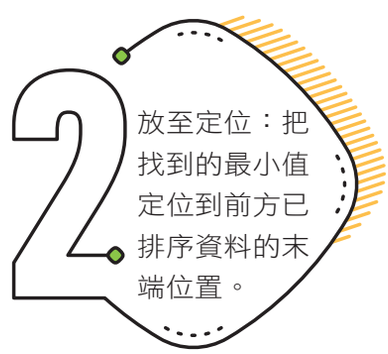
5-1-2 選擇排序法（Selection Sort）實作

選擇排序運作的過程為：先在一群尚未排序的資料（元素）中，比較並得到最小的資料（元素）後，再放到資料群（串列）的起始位置，此即完成第一個元素的定位；接著再從後方尚未排序的剩餘資料中，繼續比較出最小元素，再放到串列前方接續已排序序列的最末端。反覆進行此二步驟，直到所有資料（元素）皆已被排序完成。過程中，若某個元素直接位於正確的最終位置上，則它不會被移動。

選擇排序的比較次數與起初的排列順序無關，其比較次數會隨著循環數遞減，例如：第一次循環 (n – 1) 次、第二次循環 (n – 2) 次，一直到第 (n – 1) 次循環的 1 次。選擇排序法的平均時間複雜度為 $O(n^2)$ 。

選擇排序法的解題過程（以由小至大為例）

選擇排序其實只要反覆執行兩個要點，依序為：



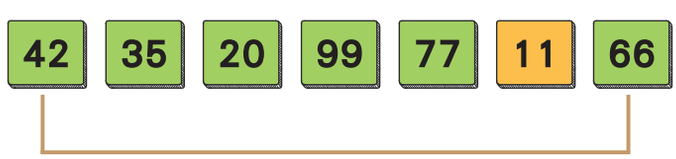
選擇排序法逐步排序過程

```
[42, 35, 20, 99, 77, 11, 66]
```

假設有一個串列如上，由小排至大，以下用圖解方式說明選擇排序法的逐步排序過程。在下列圖表中，我們將尚未排序好的資料用綠色表示，每次循環所比較出的最小值以橘黃色表示，已排序好的資料則以藍色表示。

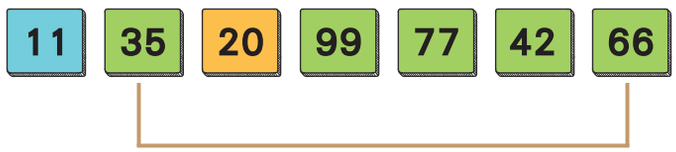
第一次循環

從串列尚未排序過的資料中，比較後得知最小值為 11，並與第一筆資料 42 交換位置。



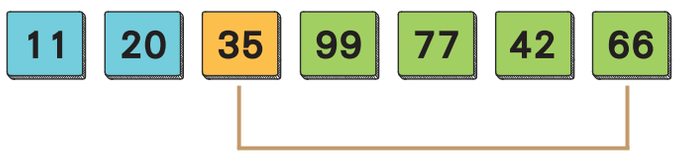
第二次循環

從串列尚未排序過的資料中，比較後得知最小值為20，並與第一筆資料35交換位置。



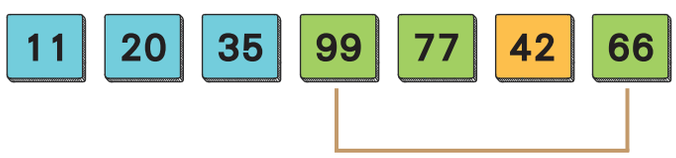
第三次循環

從串列尚未排序過的資料中，比較後得知最小值為35，且35已位於第一筆資料位置，故不須移動。



第四次循環

從串列尚未排序過的資料中，比較後得知最小值為42，並與第一筆資料99交換位置。

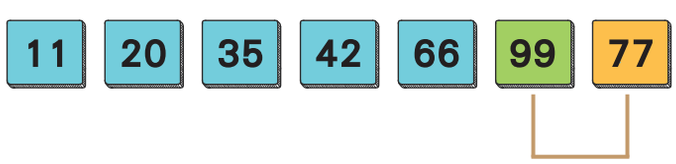


第五次循環

從串列尚未排序過的資料中，比較後得知最小值為66，並與第一筆資料77交換位置。

第六次循環

從串列尚未排序過的資料中，比較後得知最小值為77，並與第一筆資料99交換位置。



排序結果

在未排序的資料最後僅剩一筆99，無須再進行比較，故最終排序結果如下：

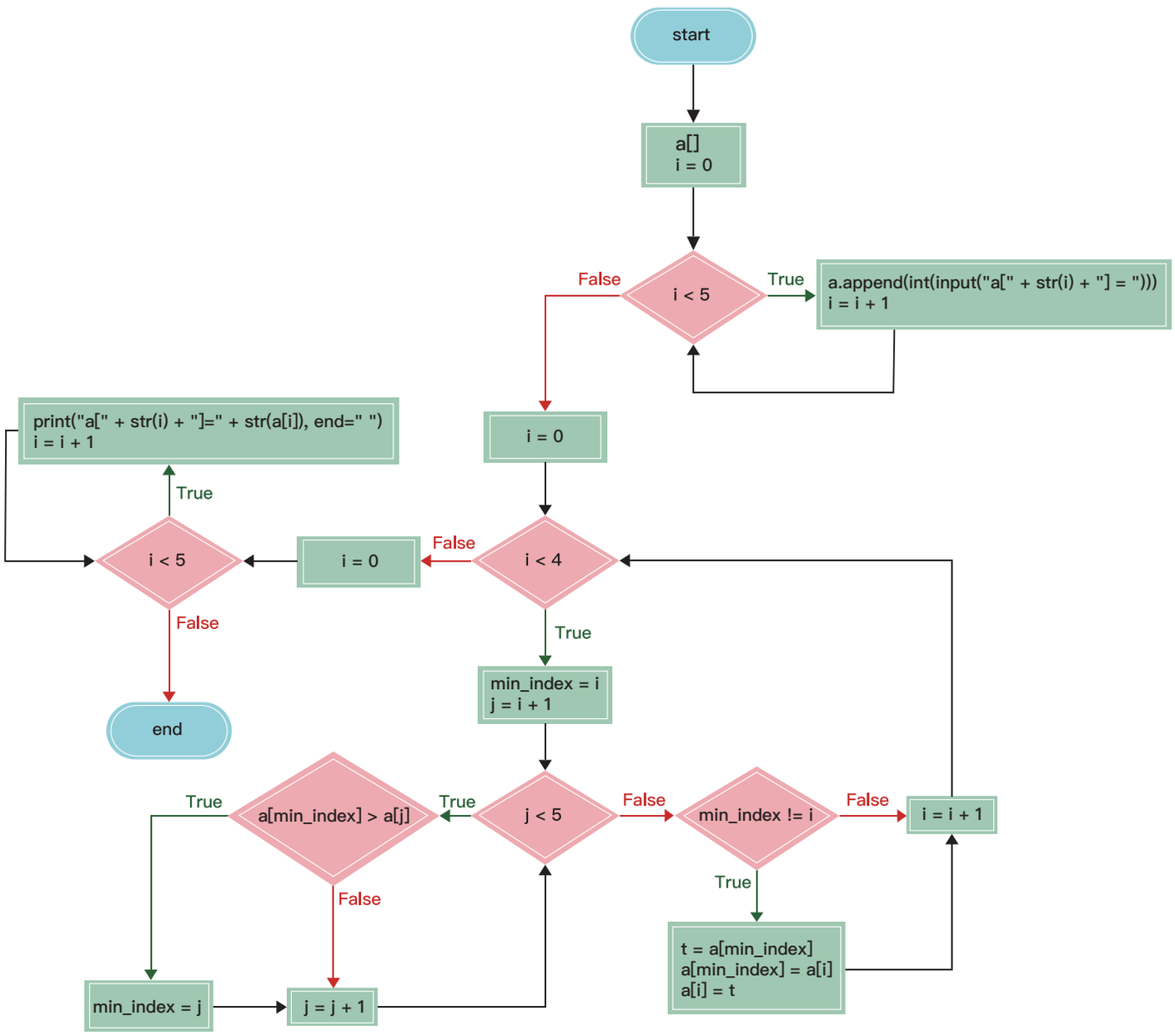


圖5-1-2 選擇排序法的流程圖

選擇排序法的程式設計

```
a = []
for i in range(5):                          先將串列資料依序輸入
    a.append(int(input("a[" + str(i) + "] = ")))

for i in range(4):                          外迴圈控制目前最前端的索引位置
    min_index = i
    for j in range(i + 1, 5):               內迴圈確認最小資料的索引值
        if a[min_index] > a[j]:             若當前最小元素大於後面的
            min_index = j                   元素，則將最小索引值改變
    if min_index != i:                      若最小值不在目前最前端則交換
        t = a[min_index]
        a[min_index] = a[i]
        a[i] = t

for i in range(5):                          將最後排序結果輸出
    print("a[" + str(i) + "]=" + str(a[i]), end=" ")
```

測試執行

```
a[0] = 42
a[1] = 35
a[2] = 20
a[3] = 99
a[4] = 77
a[0]=20 a[1]=35 a[2]=42 a[3]=77 a[4]=99
```

科技即戰力

請同學以上述選擇排序的程式及右側「測試執行」的輸入值為例，修改程式碼，讓「測試執行」能逐步看出排序的過程。

5-1-3 插入排序法實作

插入排序法就如同老師以身高作為依據，替班上同學所排出的隊伍：全班同學一開始是一群尚未排列的隊伍，由老師依序將每位同學從未排序的隊伍中，安插進入已排序的隊伍。

插入排序法逐步排序過程

```
[42, 35, 20, 99, 77, 11, 66]
```

假設有一個串列如上，由小排至大，以下用圖解方式說明插入排序法的逐步排序過程。在下列圖表中，我們將尚未排序好的資料用綠色表示，每循環所要插入的目標資料以橘黃色表示，已排序好的資料則以藍色表示。

在原始資料集（串列）中，取第一筆資料插入已排序數列。

第一次循環

35<42，35 插入 42 前方。

每一循環將未排序的第一筆資料和已排序數列比較，將值插入已排序數列的合適位置。插入時由數列右端（後）往左端（前）比較，直到遇到比正在處理的值更小的值，再行插入。

接下來反覆進行上述的比較步驟。

第二次循環

20<42，20<35，20 插入 35 前方。

第三次循環

99>42，99位置不動。

第四次循環

77 < 99，77插入99前方。77 > 42停止插入。

第五次循環

11 < 99，11 < 77，11 < 42，11 < 35，11 < 20，11最後插入20前方。

第六次循環

66 < 99，66 < 77，66插入77前方。66 > 42停止插入。

排序結果

直到最後一筆資料也插入已排序數列中，則排序結束。

插入排序法若是在每筆資料比較次數最差（多）次數為$n(n - 1) / 2$的情況下，時間複雜度為$O(n^2)$；而在最佳情況（每循環只比1次就插入完成）時，則共需要$(n - 1)$次，此時時間複雜度為$O(n)$。

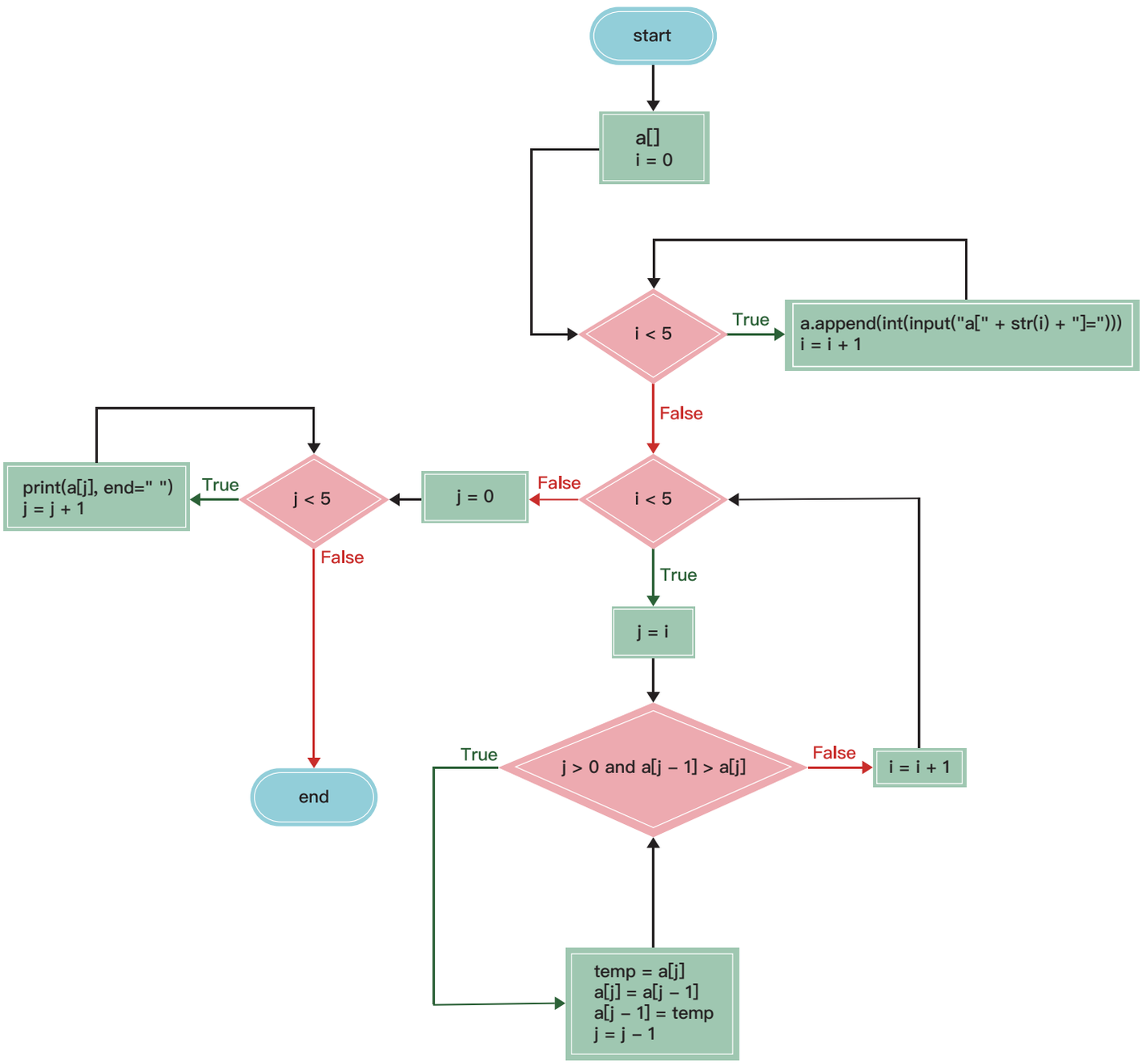


圖5-1-3 插入排序法的流程圖

插入排序法的程式設計

```
a = []
for i in range(5):
    a.append(int(input("a[" + str(i) + "]=")))

for i in range(5):
    j = i
    while j > 0 and a[j - 1] > a[j]:
        temp = a[j]
        a[j] = a[j - 1]
        a[j - 1] = temp
        j = j - 1

for j in range(5):
    print(a[j], end=" ")
```

先將串列資料依序輸入

插入排序即被插入的資料會向後位移

當 j 為正數且正要處理的數小於前一項時，啟動一連串插入且交換的過程

在已排序的數列中由後往前尋找合適的插入位置，直到後項 a[j] > 前項 a[j - 1] 才停止

列出排序結果

測試執行

```
a[0]=42
a[1]=35
a[2]=20
a[3]=99
a[4]=77
20 35 42 77 99
```

請同學以上述插入排序的程式及「測試執行」的輸入值為例，修改程式碼，讓「測試執行」能逐步看出排序的過程。

5-2 搜尋法實作

在日常生活中，我們常常進行資料的搜尋，如透過電話簿尋找電話號碼、在成績單進行名次的搜尋、在搜尋引擎網站查找資料等，在這章節中我們將介紹兩種搜尋法的搜尋過程。

「**搜尋**」即是在一群資料中找出目標資料（又稱**鍵值**，實作時常以key作為其變數名稱）。搜尋法的主要概念為「**資料的相互比較**」，必須透過資料的比較，才能判斷是否搜尋到目標資料。當資料量少時，很容易即可找到目標資料；但若資料量非常巨大時，如何有效地進行搜尋就是一門重要的課題。

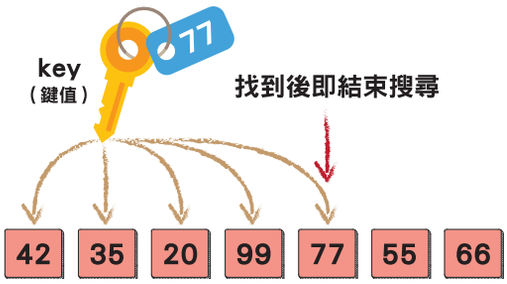
5-2-1 循序搜尋法（Sequential Search）實作

圖5-2-1 循序搜尋法如老師從隊伍的第一位同學，逐一地詢問想要尋找的同學姓名

循序搜尋法的解題過程

在Chapter 02演算法的單元中我們曾介紹過，循序搜尋法即是暴力法的解題策略之一。循序搜尋法是指從串列中第一筆資料開始取出，接著逐一與目標資料（鍵值）相互比對，直到找到所要元素的位置，或所有資料皆尋找完畢為止。其優點是邏輯較為直觀，程式容易撰寫，且資料不需先行排序，可直接進行搜尋（圖5-2-1）；缺點是搜尋效率比較差，假如有 n 筆資料，則平均花費次數為 (n + 1) / 2，不管是否有排序，每次都必須要從頭到尾找一次。

下列為從一串列中找出數字 77，使用循序搜尋法之示意圖：



時間複雜度

1 如果資料沒有重覆，找到資料即可終止，否則要找到資料結束。N 筆資料，最好之情況為 1 次，記為 O(1)。最差之情況下，須進行 N 次比較，記為 O(N) 。

2 假設資料出現與分布之機率相等，平均次數需 (N + 1) / 2 次比較。

$$\text{平均次數} = \frac{(1+2+3+...+N)}{N} = \frac{(N+1)}{2}$$

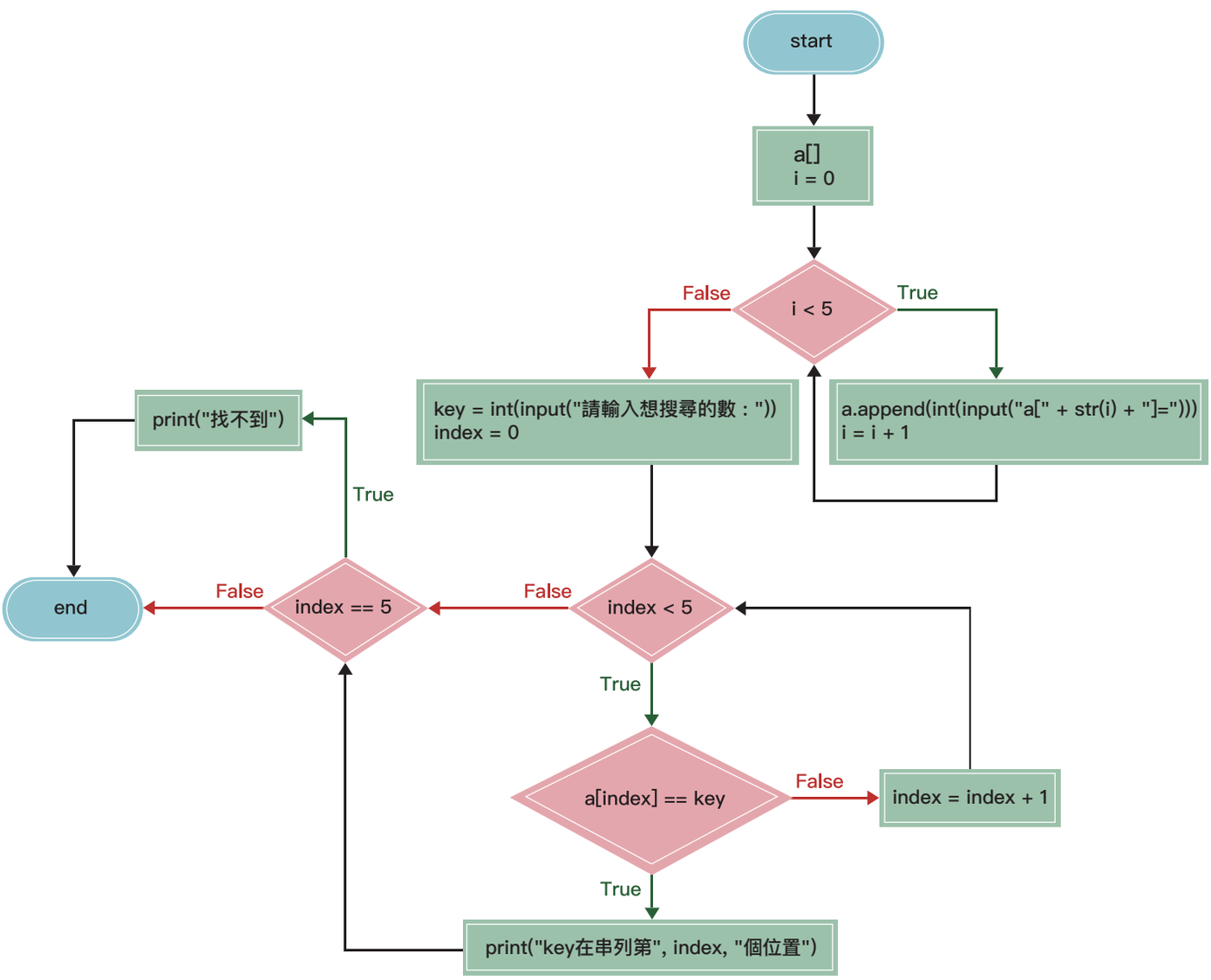


圖5-2-2 循序搜尋法的流程圖

循序搜尋法的程式設計

```
a = []
for i in range(5):
    a.append(int(input("a[" + str(i) + "]=")))

key = int(input("請輸入想搜尋的數 : "))
index = 0
while index < 5:
    if a[index] == key:
        print("key在串列第", index, "個位置")
        break
    index += 1
if index == 5:
    print("找不到")
```

比對串列內的資料是否等於欲搜尋的鍵值

若找到符合條件的資料，就傳回其索引

若搜尋不到，代表迴圈沒中斷，則 index == 5

測試執行 1

```
a[0]=42
a[1]=35
a[2]=20
a[3]=99
a[4]=77
請輸入想搜尋的數：20
key在串列第 2 個位置
```

測試執行 2

```
a[0]=42
a[1]=35
a[2]=20
a[3]=99
a[4]=77
請輸入想搜尋的數：15
找不到
```

5-2-2 二分搜尋法（Binary Search）實作

二分搜尋法的解題過程

二分搜尋法概念上類似我們玩過的猜數字遊戲，例如：先從 1 至 100 之間指定一個數字為答案，再開始猜數字。首先我們會先猜 50，將答案範圍縮小一半，技巧就是每次都選擇所剩區間範圍的中間值，反覆進行上述步驟，即可很快地猜到答案。

在使用二分搜尋法之前，必須先將數列進行排序，接著將排序後的串列資料拆成兩部分，即各一半的資料數，再將**鍵值**（key，**搜尋目標**）與**中間值**比較。假如鍵值與中間值相等，則搜尋成功，並回傳鍵值所在的串列索引位置；若中間值大於鍵值，再往前半部分搜尋；若中間值小於鍵值，再往後半部分搜尋。反覆地將資料切半並比對，直到找到鍵值或沒有資料為止。其優點為搜尋效率佳（若有 n 筆資料，則**平均搜尋次數** $= Log_2 n$），因為每次比較都會比上一次少一半的資料，因此最多只需要比較 $Log_2 n$ 次，而最佳情況是比較 1 次就找到，但限制是資料必須先經過排序。

二分搜尋法逐步搜尋過程

```
a = [11, 20, 35, 42, 66, 77, 99]
```

假設有一個已排序串列如上，以下用圖解的方式說明二分搜尋法逐步的搜尋過程（搜尋鍵值為66）。在下列圖表中，我們將左側邊界範圍以綠色指標 **low** 記錄索引，右側邊界範圍以橙色指標 **high** 記錄索引，中間值以紅色指標 **mid** 記錄索引。

而由於low、high、mid皆為整數變數（索引值），故mid=(low+high)/2的值亦為整數（若有浮點數則取高斯符號）。

步驟一

low = 0

high = 6

mid = (0+6)//2 = 3

42 < 66，故將左側邊界索引 low 向右移動至 mid 右側，low = mid + 1 = 3 + 1 = 4

步驟二

low = 4

high = 6

mid = (4+6)//2 = 5

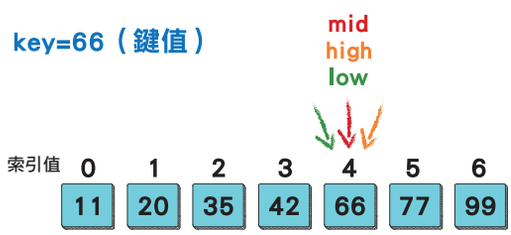
77 > 66，故將右側邊界索引 high 向左移動至 mid 左側，high = mid - 1 = 5 - 1 = 4

步驟三

low = 4

high = 4

mid = (4+4)//2 = 4



a[4] = 66，key = 66，鍵值與中間值相等，搜尋成功且停止搜尋，並回傳鍵值 key 所在的串列索引位置 4，即為所求。

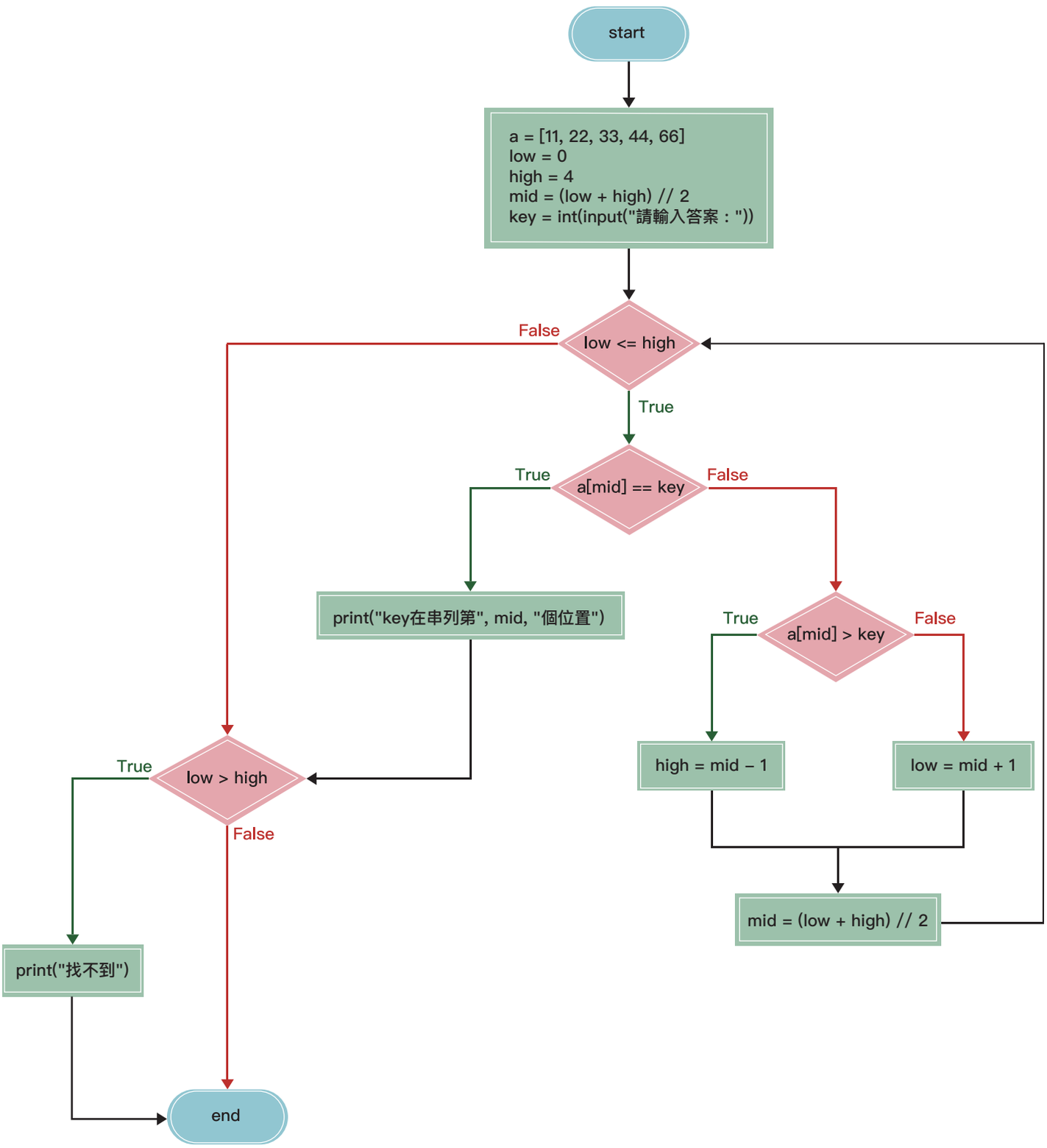


圖5-2-3 二分搜尋法的流程圖（串列元素需先經過排序）

二分搜尋法的程式設計（串列元素需先經過排序）

```
a = [11, 22, 33, 44, 66]
low = 0
high = 4
mid = (low + high) // 2          搜尋中間值的索引位置
透過 low 、 high 兩變數作為指標，不斷與 mid 調整並更新，以此逼近想搜尋的鍵值
key = int(input("請輸入答案 : "))
while low <= high:
    if a[mid] == key:            找到資料
        print("key在串列第", mid, "個位置")   顯示資料位置
        break                    跳出迴圈
    elif a[mid] > key:
        high = mid - 1           將high移動至mid左側位置
    else:
        low = mid + 1            將low移動至mid右側位置
    mid = (low + high) // 2      更新中間值索引

if low > high:                   high < low，表示搜尋不到資料，停止搜尋
    print("找不到")
```

測試執行 1

```
請輸入答案： 33
key在串列第 2 個位置
```

測試執行 2

```
請輸入答案：52
找不到
```

科技即戰力

請同學用二分搜尋法，在數列：4、9、15、19、63、96中尋找資料19，並記錄逐步搜尋的過程。

5-3 遞迴實作

在Chapter 02我們有提到**遞迴**的觀念，即**自訂函數**內部自己呼叫自己的過程，且會有**終止條件**，以結束遞迴函數。以下我們採用階乘函數與最大公因數（輾轉相除法）的求解過程，做為以遞迴解題的實作演練。

5-3-1 階乘函數的解題過程

若以數學算式解析遞迴運作

```
通式 mul(x) = x * mul(x - 1)
終止條件 mul(1) = 1
a = mul(5) = 5 * mul(4)
           = 5 * 4 * mul(3)
           = 5 * 4 * 3 * mul(2)
           = 5 * 4 * 3 * 2 * mul(1)   終止條件
           = 5 * 4 * 3 * 2 * 1
```

5 階乘的程式實作

```
def mul(x):                  自訂函數
    if x == 1:
        return 1             終止條件
    return x * mul(x - 1)    通式

a = mul(5)                   呼叫mul(x)函數
print(a)
```

測試執行

```
120
```

5-3-2 找出兩數的最大公因數（以輾轉相除法）的解題過程

當要找出兩數的最大公因數，我們可以使用輾轉相除法的方式解決此類問題，其過程如下：

1. 輸入兩正整數a、b（a > b），令a為被除數，b為除數，開始進行兩數相除取餘數。
2. 並將本次運算過程的除數和餘數分別作為下一次相除的被除數、除數。
3. 運用迴圈反覆進行步驟1、2直到整除為止，整除前的最後一個除數即為最大公因數。

兩數的最大公因數程式實作（以輾轉相除法）

```
def gcd(x, y):          自訂函數
    if y == 0:
        return x
    else:
        return gcd(y, x % y)

a = input("請輸入a (a > b):")
b = input("請輸入b (a > b):")
a = int(a)
b = int(b)
print("最大公因數為 :", gcd(a, b))
```

測試執行

```
請輸入a (a > b):52
請輸入b (a > b):24
最大公因數為 : 4
```

科技即戰力

請同學以遞迴和迴圈方式，分別實作費氏數列程式 fib(n)，並求得 fib(12) 的值。定義 fib(0) = 0，fib(1) = 1，fib(n) = fib(n-1) + fib(n-2)。

課堂檢測

( ) 1. 利用氣泡排序法將數列資料 [23, 12, 37, 3, 52] 由小而大、依序排列，在第二次循環結束後，數列排列的第二個數字是？
(A)3 (B)12 (C)23 (D)37。

( ) 2. 氣泡排序若有10筆資料進行排序，請問需要幾次循環才能完成？
(A) 10次 (B) 11次 (C) 8次 (D) 9次。

( ) 3. 10筆資料在氣泡排序的第三次循環中，共需比較幾回合呢？
(A) 9次 (B) 8次 (C) 6次 (D) 7次。

( ) 4. 若A[1]、A[2]，分別為串列A[ ]的兩個元素，下列哪個程式片段可以將A[1]和A[2]的內容交換？
(A) A[1] = A[2]; A[2] = A[1];
(B) t = A[1]; A[1] = A[2]; A[2] = t;
(C) A[2] = A[1]; t = A[2]; A[1] = t;
(D)以上皆可。

( ) 5. 利用選擇排序法將數列資料 [23, 12, 37, 3, 52] 由小而大、依序排列，在第二次循環結束後，數列排列的第二個數字是？
(A)3 (B)12 (C)23 (D)37。

( ) 6. 若有8筆資料，請問選擇排序法的總比較次數為多少呢？
(A)7次 (B)14次 (C)28次 (D)56次。

( ) 7. 利用氣泡排序法，將以下資料：10、9、7、3由大到小排列，共需進行幾次比較？
(A)12 (B)4 (C)6 (D)8。

( ) 8. 若有10筆資料，請問插入排序法的最佳次數為下列何者呢？
(A)45 (B)8 (C)9 (D)10。

( ) 9. 承上題，請問插入排序法的最差次數為下列何者呢？
(A)9 (B)10 (C)45 (D)90。

( ) 10. 玩撲克牌時，我們常運用下列何種排序演算法，依大小順序排列我們手中的牌呢？
(A)氣泡排序 (B)選擇排序 (C)合併排序 (D)插入排序法。

( ) 11. 若以二分搜尋法在數列 [12, 29, 36, 48, 61, 82, 100] 中，搜尋資料61，試問需要幾次搜尋呢？
(A)3 (B)2 (C)4 (D)6。

( ) 12. 若有7筆資料採用循序搜尋法，請問平均花費次數為何？
(A)49 (B)21 (C)3 (D)4。

( ) 13. 下列關於二分搜尋法的敘述，何者正確？
(A)不須事先排列 (B)平均搜尋次數較循序搜尋沒效率
(C)主要有4個指標決定搜尋位置 (D)每次比較都會比上一次少一半的資料。

( ) 14. 若有1,000位學生，請問以循序搜尋來說，最差的情況需要幾次比較才能找到該位學生？
(A)1次 (B)500次 (C)999次 (D)1,000次。

( ) 15. 若有1,000位學生，請問以二分搜尋來說，最差的情況需要幾次比較才能找到該位學生？
(A)1次 (B)10次 (C)9次 (D)100次。

( ) 16. 以下是階乘函數的程式，請問空格內應該填入下列何者？
(A) x * mul(x - 1)
(B) (x - 1) * mul(x)
(C) mul(x)
(D) mul(x - 1)。

```
def mul(x):
    if x == 1:
        return 1          終止條件
    return _____________  通式
```

( ) 17. 以下是費氏數列函數的程式，請問空格內應為下列何者？
(A) fib(n) + fib(n - 1)
(B) fib(n - 1) + fib(n - 2)
(C) fib(n) + fib(n - 2)
(D) fib(n - 1) + fib(n - 3)。

```
def fib(n):
    if n == 1 or n == 2:
        return 1
    if n >= 3:
        return _____________
```

( ) 18. 以下是以遞迴實作輾轉相除法求最大公因數的自訂函數程式，請問空格內應為下列何者？
(A) gcd(b, a % b)
(B) gcd(b, b % a)
(C) gcd(a, a % b)
(D) gcd(a, b % a)。

```
def gcd(a, b):
    if b == 0:
        return a
    return _____________
```

( ) 19. 以下是以迴圈實作輾轉相除法求最大公因數的自訂函數程式，請問空格內應為下列何者？
(A) a % b
(B) b % a
(C) a / b
(D) b / a。

```
a = int(input("請輸入a: "))
b = int(input("請輸入b: "))
while b != 0:
    t = b
    b = _______
    a = t
print(a)
```

主題三

系統平台

chapter 06
系統平台之運作原理　168

chapter 07
網際網路運作原理　186

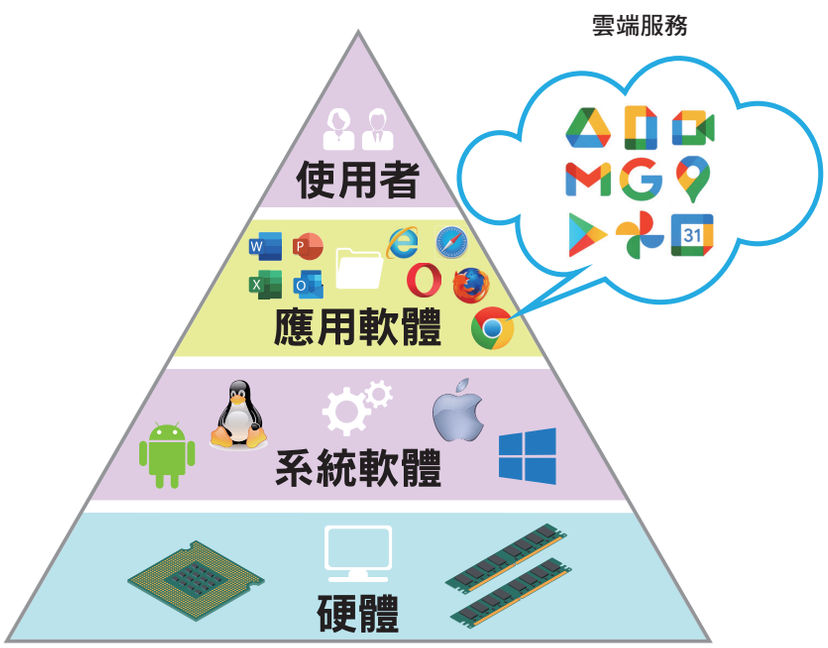
chapter 08
系統平台之未來發展趨勢　204

chapter

06 系統平台之運作原理

早期資訊系統多數是由**硬體**、**軟體**所建構的系統平台，此系統平台提供基礎的架構和環境。如今，拜網路科技的蓬勃發展所賜，許多資訊系統必須透過**網路**的連結，以提供我們進行更多且更為廣泛的延伸與應用。

而系統的硬體資源往往是有限的，因此可以透過**作業系統**，有效率地規劃硬體資源，如記憶體管理、輸出入設備管理等，且能同時處理多個工作排程；透過網路，則可以連結許多資源，如雲端運算所使用到的各類伺服器與硬碟空間，提升系統的運算與儲存能力，創造更多的應用。



6-1 系統平台的概念

系統平台意指可提供程式與軟體執行的環境，而此環境包含了硬體、作業系統等。系統平台發展至今，除了硬體設備體積不斷地縮小，運算速度越來越快之外，軟體也隨著網路技術的發展，從過去將**應用軟體**安裝在作業系統上並執行，轉為只要透過瀏覽器，即可開啟相關的應用軟體，這也實現了軟體跨平台的應用，意即應用程式的執行不再受限於特定的硬體或作業系統的平台。

6-1-1 電腦的硬體

電腦的硬體一共可分為五大單元，分別是**控制單元**、**算術／邏輯單元**、**記憶單元**、**輸入單元**、**輸出單元**，以下介紹硬體各單元主要負責的工作、**控制訊號**與**資料流向**，如圖6-1-1。

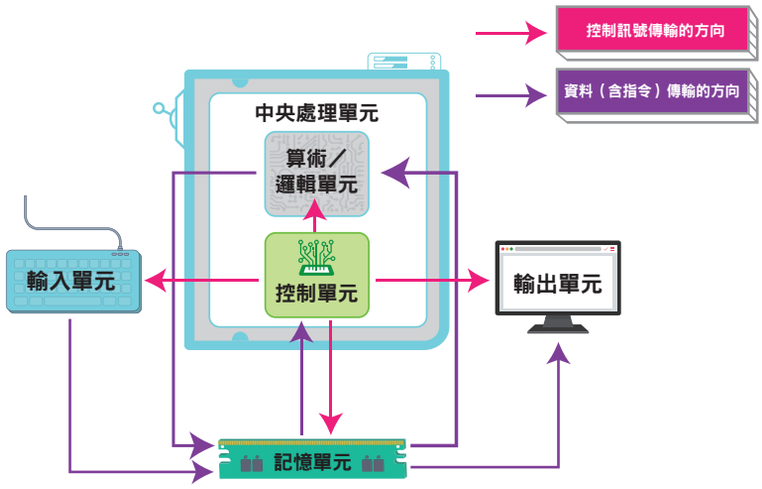


圖6-1-1 電腦五大單元

控制單元（CU）

控制**資料流向**以及**解碼**各項指令，並依指令指揮各單元間的運作與執行，猶如一位站在路口的交通警察。

算術／邏輯單元（ALU）

將來自記憶單元的資料進行**運算**或是**邏輯判斷**等工作，再將運算結果回存至記憶體。**算術／邏輯單元**與**控制單元**合稱為**中央處理單元**（CPU，圖6-1-2）。

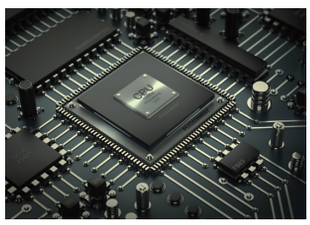


圖6-1-2 中央處理單元CPU

記憶單元（MU）

儲存**指令**或**資料**的位置。如主記憶體（隨機存取記憶體，RAM），輔助記憶體（硬碟、隨身碟）等皆為**記憶單元**的裝置（圖6-1-3）。

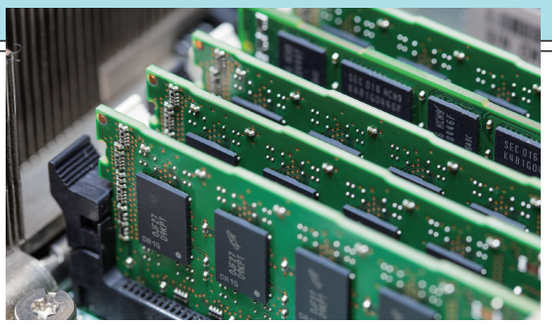


圖6-1-3 主記憶體

輸入單元（IU）

使電腦能接收外部訊息與資料的裝置。如鍵盤、滑鼠、網路攝影機、麥克風等皆屬於輸入裝置。

輸出單元（OU）

將運算結果輸出至外部的裝置。如螢幕（若有觸控功能則同時屬於輸入單元）、3D印表機、喇叭等皆為輸出單元的裝置。

6-1-2 系統軟體（System Software）

系統軟體是協助電腦硬體運行的程式，而當中的**作業系統**（Operating System）可以作為各類**應用軟體**安裝的平台，同時也管理並協調各項硬體的資源配置，如CPU、記憶體、輸出入裝置等。Microsoft Windows、Apple MacOS、UNIX、Linux等皆為**作業系統**。而依據不同用途的系統平台，其所搭配的硬體，往往也會採用相對應的作業系統，如智慧型手機需要搭配低耗電的處理器，以及如iOS、Android等作業系統。

應用軟體開發者也會依據不同的作業系統，開發出能在該作業系統上執行的應用軟體，讓應用軟體無論在哪種作業系統上都能順利執行。

作業系統會協助使用者執行各項工作，使用者完全不用了解背後運作的原理，例如：我們可以同時執行編輯文件和聽音樂等程式，都是藉由作業系統進行**資源控管**和**工作行程**安排。

此外，**公用軟體**（Utility）及**程式發展工具**（Program Development Tool）也被分類為**系統軟體**。公用軟體可協助使用者主動進行部分資源管理的工作，如磁碟掃描、磁碟重組（圖6-1-4）、磁碟清理、檔案壓縮等程式，皆可讓使用者主動進行系統資源的最佳化。

程式發展工具則是協助程式開發的工具，包含編輯介面軟體、整合開發環境、組譯程式（Assembler）、編譯程式、連結程式、載入程式、偵錯程式（Debugger）等，如 DEV-C++ 就是一套提供**整合開發環境**的程式發展工具。

圖6-1-4 磁碟重組工具

6-1-3 應用軟體（Application Software）

應用軟體是人們為了處理或解決某些問題而開發的程式，而市面上有許多被開發來解決多數共同問題或工作的軟體，這類軟體通常極具有商業價值，因此多數大型軟體公司如Microsoft、Adobe等皆會開發相關的軟體進行銷售，例如：Microsoft Office屬於辦公室自動化軟體、Adobe Photoshop屬於影像處理軟體、CorelDRAW屬於繪圖軟體、Adobe Dreamweaver屬於網頁設計軟體、Adobe Indesign屬於排版軟體。

此外，當然也有專門為特定單位或活動所量身訂做的特殊需求軟體，此類型軟體由於需要專案開發，因此通常單價較高，例如：疫苗預約系統、學習歷程檔案系統（圖6-1-5）、財政部報稅系統等。

圖6-1-5 高中的學習歷程檔案系統是屬於量身訂做的特殊需求軟體

科技即戰力

智恩考上了某大學的動畫學系，決定幫自己買一臺用來繪圖的新電腦。她的預算是新臺幣三萬元，請同學上網幫她查詢各種硬體、軟體的與規格，規劃出符合她需求的電腦配備。（可參考各線上查價網站）

硬體與軟體類型	CPU	MU	OU	IU	作業系統	應用軟體
規格						

6-2 程序管理與資源分配

即使CPU的速度再怎麼快，單一核心的CPU一次仍僅能處理一個**程序**（Process，正在CPU執行的工作），但由於CPU可以透過程序管理進行多項工作的**排程**，並期望透過有效率的方式依序處理每一項工作，進而讓使用者感覺CPU是在同一時間內，執行多個程式（圖6-2-1）。

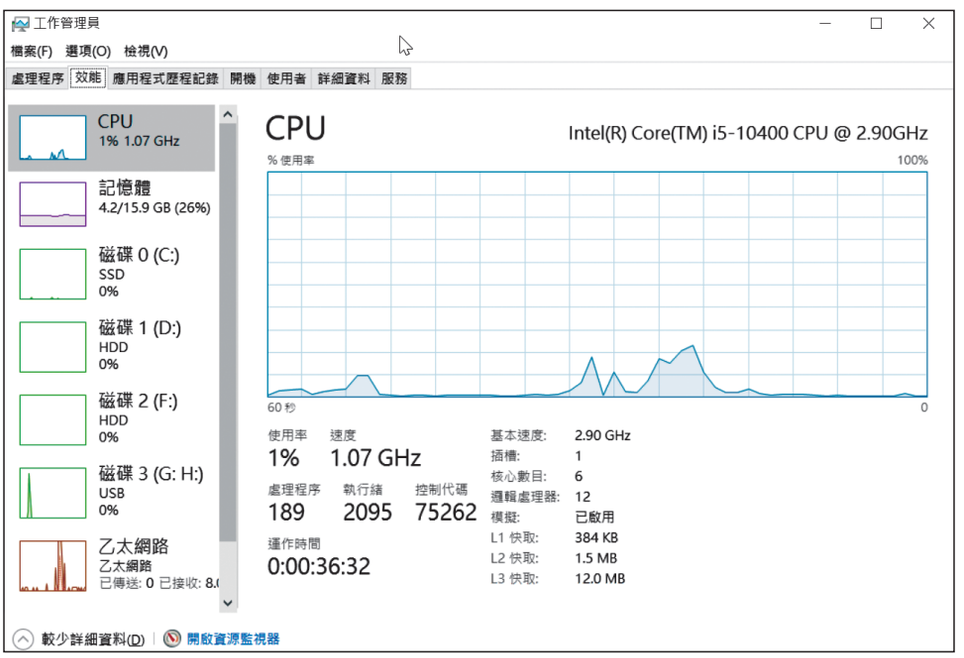


圖6-2-1 作業系統記錄CPU各核心的使用率

6-2-1 程序的狀態

由於同一時間只能有一個程序進入CPU執行，因此我們運用**排程**解決CPU使用權的問題。而一個程序從新建（New）到執行結束（Terminated），如下圖6-2-2，通常會不斷地在「預備」、「執行」、「等待」三種狀態之間切換，直到完成為止。

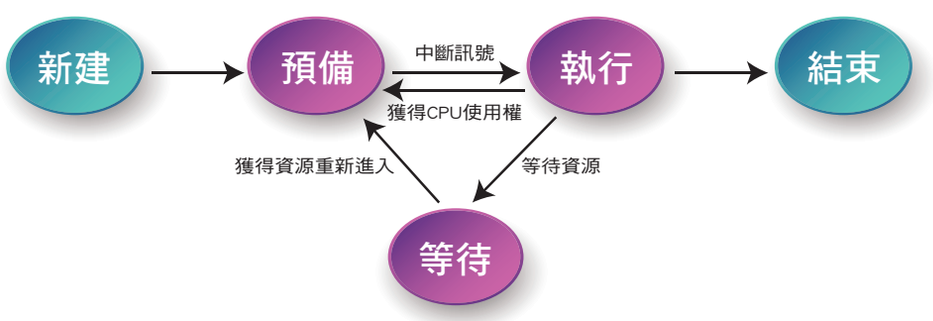


圖6-2-2 程序從新建後會在不同狀態間切換，直到結束。

預備狀態（Ready State）

程序已取得所需的資源。

執行狀態（Running State）

程序正在CPU中執行，但可能因為需要取得某些資源而進入等待狀態。也可能因為系統發出中斷訊號（Interrupt），例如：此次程序獲得的執行時間已結束，而重新回到預備狀態。

等待狀態（Wait State）

執行過程中可能還需要取得某些資源，此時則會重回等待狀態，待程序獲得所需的資源再進入預備狀態。

IT 補給站

死結狀態（Deadlock State）

兩個程序彼此占用對方所需資源且不釋放，造成**循環等待**，使CPU無法繼續往後執行。如圖6-2-3，程序P1占用資源R2，並要求取得資源R1；程序P2占用資源R1，並要求取得資源R2，形成循環等待的**死結**狀態。

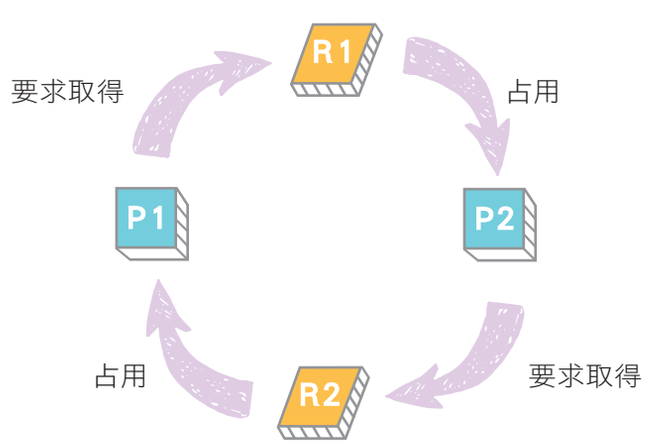


圖6-2-3 死結狀態

6-2-2 程序管理機制——CPU工作排程演算法

在系統運作時，允許多個工作同時處於執行狀態，但是實際上CPU在同一時間內僅能處理一項工作，是由哪項工作能優先獲得CPU執行的資源，而又是哪些工作先進入等待？評估CPU工作排程演算法的效能主要有四項指標：

CPU使用率（Utilization）	工作產出量（Throughput）
CPU若長時間處於等待狀態是系統資源利用效能不高的缺失，因此盡量避免讓CPU閒置。	完成的工作數量。
完成時間（Turnaround Time）	**等待時間（Waiting Time）**
從工作程序進入到系統並執行完畢後的時間區間總和。	進入系統後等待獲得CPU執行的時間區間總和。

接下來將介紹四種常見的CPU排程演算法，分別評估**平均完成時間**與**平均等待時間**兩項效能指標：

先到先服務（First Come First Served，FCFS）

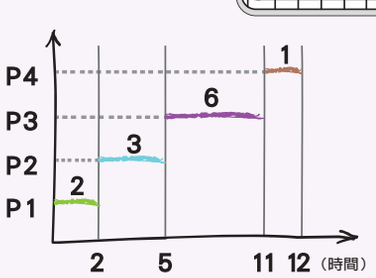
由抵達記憶體的先後次序決定程序獲得CPU資源的順序，這是最簡單且最直接的方式，只要使用先前在演算法章節提過的佇列的機制——**先進先出**，就可以如同排隊一樣依序執行。優點是演算法設計簡單且便於管理，缺點是一旦前幾項程序所需的執行時間較長，則後面程序的等待時間也會相對拉長。

範例6-2-1a 處理器排程（FCFS）

Q 請以先到先服務（FCFS）求解平均完成時間與平均等待時間。

程序代號	執行時間	優先權
P1	2	4
P2	3	1
P3	6	2
P4	1	3

先到先服務（FCFS）



FCFS的平均等待時間：
(0+2+5+11)/4=4.5
FCFS的平均完成時間：
(2+5+11+12)/4=7.5

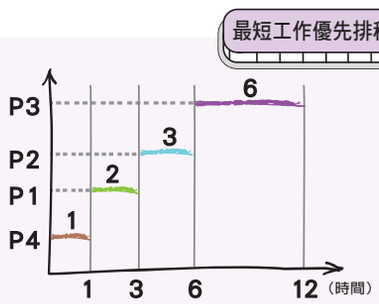
最短工作優先排程（Shortest Job First，SJF）

在記憶體等待執行的程序中，選擇**最短執行時間**的程序優先獲得CPU執行的資源。理論上，此種演算法會得到**最短的**平均等待時間，但由於程序執行的過程中仍有諸多變因，以致於無法準確預測執行時間，例如：需要等待輸入裝置取得資料。另一考量是萬一不斷有執行時間較短的程序進入排程，則較長執行時間的程序會進入無止盡的等待，此種情況稱為**飢餓現象**（Starvation）。

範例6-2-1b　處理器排程（SJF）

Q 請以最短工作優先排程（SJF）求解平均完成時間與平均等待時間。

程序代號	執行時間	優先權
P1	2	4
P2	3	1
P3	6	2
P4	1	3



SJF的平均等待時間：
(0+1+3+6)/4=2.5
SJF的平均完成時間：
(1+3+6+12)/4=5.5

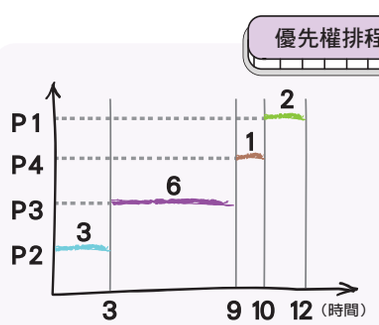
優先權排程（Priority Scheduling，PS）

依照各個程序的**優先權**順序依序執行，而優先權排程也會面臨如同最短工作優先排程的飢餓現象，當不斷有優先權較高的程序進入排程，原先優先權較低的程序可能會不斷受到插隊的排擠，導致需要更久的等待時間，甚至進入無止盡的等待。

範例6-2-1C　處理器排程（PS）

Q 請以優先權排程（PS）求解平均完成時間與平均等待時間。

程序代號	執行時間	優先權
P1	2	4
P2	3	1
P3	6	2
P4	1	3



PS的平均等待時間：
(0+3+9+10)/4=5.5
PS的平均完成時間：
(3+9+10+12)/4=8.5

輪詢排程（Round-Robin Scheduling，RRS）

上述FCFS、SJF、PS三種排程演算法都是等待一個程序執行完成後，再接著執行下一個程序，所以當某一程序占用CPU執行時間過久，或是因程序被排擠而產生飢餓現象，將會導致平均完成時間與平均等待時間的拉長，造成系統效能不彰。

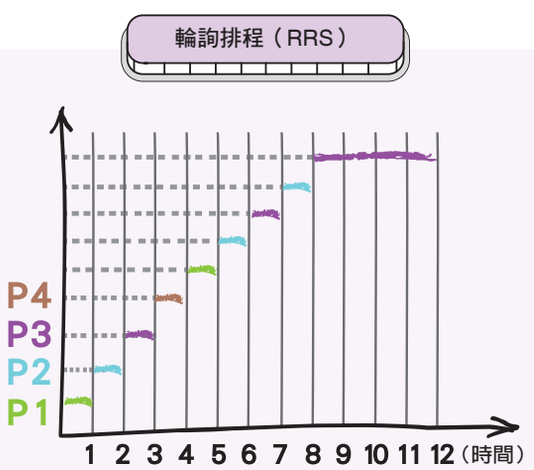
而**輪詢排程**中，無論程序所需時間的長短，每一個程序只會分配到短且固定的**時間片段**，當時間片段一到，系統會中斷該程序的執行且儲存該程序目前的狀態，再由下一個程序接續執行，反覆地進行上述步驟，直到所有程序執行完畢為止。

當切割的時間片段愈短，所有程序就會有同時進行的錯覺，但前提是CPU的效能要夠好，否則使用輪詢排程時，在程序間的頻繁切換會造成系統嚴重的負擔。

範例6-2-1d　處理器排程（RRS）

Q 請以輪詢排程（RRS）求解平均完成時間與平均等待時間。（時間片段＝1）

程序代號	執行時間	優先權
P1	2	4
P2	3	1
P3	6	2
P4	1	3



RRS的平均等待時間：
P1等待的時間為0+(4-1)=3
P2等待的時間為(1-0)+(5-2)+(7-6)=5
P3等待的時間為(2-0)+(6-3)+(8-7)=6
P4等待的時間為(3-0)=3
(3+5+6+3)/4=4.25
RRS的平均完成時間：
(5+8+12+4)/4＝7.25

科技即戰力

期中考將至，數學老師雄雄規劃了整節課協助同學進行一對一解題。請同學使用FCFS、SJF、PS、RRS四種排程方式，計算出下表各同學提問的平均等待時間和解題完成時間。（提問先後次序為云云、潔潔、凱凱，時間片段：3分鐘）

提問同學	需求時間（分鐘）	優先權
云云	23	3
潔潔	11	2
凱凱	16	1

6-3 記憶體的配置與管理

記憶體的配置與管理的目的是為了提高系統對於記憶體的使用效率，並減少主記憶體到硬碟的存取次數，因此對於系統運作的效率有著重要的影響。

6-3-1 分頁（Paging）

很多作業系統都是使用**分頁技術**控管有限的主記憶體資源，如Linux、Windows、MAC OS等。分頁技術是指將主記憶體空間切割成數個固定大小的**頁框**（Frame），再將輔助記憶體（如硬碟）中的資料切成數個固定大小的**分頁**（Page），頁框和分頁的大小必須一樣，等到有需要存取的時候，再替換（Replacement）所需的分頁至主記憶體空間，如下圖6-3-1。

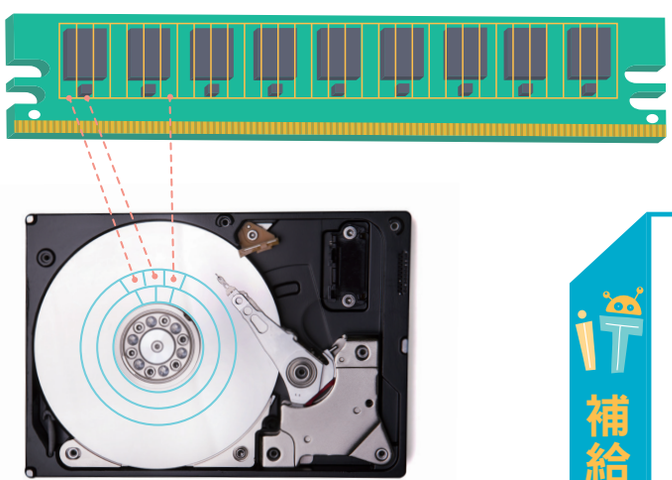


圖6-3-1 將輔助記憶體內的資料切成數個固定大小的分頁，待需要存取時再替換至主記憶體的頁框中

IT補給站

分頁錯誤（Page Fault）

當作業系統要存取某個分頁時，發現這個分頁並沒有在主記憶體的頁框中，即產生**分頁錯誤**，此時就必須進行**分頁替換**（Page Replacement），替這個目標**分頁**清出一個空的**頁框**來放置。

6-3-2 替換演算法

記憶體如何配置和管理記憶體的分頁，首要目標就是盡量降低分頁錯誤的次數，提升作業系統至主記憶體取得資料的**命中率**，以減少到輔助記憶體搜尋資料的次數。以下介紹三種常見的分頁替換演算法：

分頁替換演算法（Page Replacement Algorithm）

以下將分別使用不同演算法，來實際模擬在主記憶體的三個頁框中，連續存取分頁編號「5 6 1 2 6 3 6 4 6 3」的替換過程，並計算分頁錯誤的次數。

1.先進先出替換演算法（First in First out Page Replacement，FIFO）

先進先出替換演算法是將最先載入主記憶體中的分頁，即存在時間最久的最優先被替換，以此種替換方式實作較為直覺且簡單。

分頁載入頁框的替換過程如下表，表格填藍色的部分表示下一次產生分頁錯誤時，預計要進行替換的頁框；表格填紅色表示本次進行替換的頁框與新進入的分頁編號，即產生分頁錯誤。後續兩種方法也依此方式標示顏色。

分頁編號		5	6	1	2	6	3	6	4	6	3
頁框1		5	5	5	2	2	2	2	4	4	4
頁框2			6	6	6	6	3	3	3	3	3
頁框3				1	1	1	1	6	6	6	6
分頁錯誤		是	是	是	是	否，在頁框2找到分頁6	是	是	是	否，在頁框3找到分頁6	否，在頁框2找到分頁3
累計分頁錯誤		1次	2次	3次	4次	4次	5次	6次	7次	7次	7次

→分頁錯誤的累計次數為7次。

2.最近最久未使用替換演算法（Least Recently Used，LRU）

將最近這段時間存在於主記憶體頁框中，且最久未使用到的分頁予以替換。

在以下的範例中，我們除了在三個頁框中記錄分頁編號外，在編號的右上角另標註了存取後的閒置時間。數字由 0 到 2 ， 0 代表此分頁剛被使用過；2 代表此分頁閒置最久，數值 2 作為優先汰換的分頁。

分頁編號		5	6	1	2	6	3	6	4	6	3
頁框1	2	5 0	5 1	5 2	2 0	2 1	2 2	2 2	4 0	4 1	4 2
頁框2	1	2	6 0	6 1	6 2	6 0	6 1	6 0	6 1	6 0	6 1
頁框3	0	1	2	1 0	1 1	1 2	3 0	3 1	3 2	3 2	3 0
分頁錯誤		是	是	是	是	否，在頁框2找到分頁6	是	否，在頁框2找到分頁6	是	否，在頁框2找到分頁6	否，在頁框3找到分頁3
累計分頁錯誤		1次	2次	3次	4次	4次	5次	5次	6次	6次	6次

→分頁錯誤的累計次數為6次。

3.最近最少使用替換演算法（Least Frequently Used，LFU）

將最近最少使用的分頁作為下次預計替換的目標。LFU 和 LRU 演算法的相異之處在於：LRU 的置換原則是依據閒置時間，而 LFU 是依據使用次數的多寡。

在以下的範例中，我們除了在三個頁框中記錄分頁編號外，在編號的右上角另標註了最近使用次數累計。剛進入頁框中記為 1 次，若累計次數相同，則依 FIFO 原則替換；若遇到重新進入頁框的分頁，累計次數仍從 1 開始計算。

分頁編號		5	6	1	2	6	3	6	4	6	3
頁框1	0	5 1	5 1	5 1	2 1	2 1	2 1	2 1	4 1	4 1	4 1
頁框2	0	0	6 1	6 1	6 1	6 2	6 2	6 3	6 3	6 4	6 4
頁框3	0	0	0	1 1	1 1	1 1	3 1	3 1	3 1	3 1	3 2
分頁錯誤		是	是	是	是	否，在頁框2找到分頁6	是	否，在頁框2找到分頁6	是	否，在頁框2找到分頁6	否，在頁框3找到分頁3
累計分頁錯誤		1次	2次	3次	4次	4次	5次	5次	6次	6次	6次

→分頁錯誤的累計次數為6次。

科技即戰力

請同學根據自己的智慧型手機型號，上網搜尋此裝置是運用哪兩種硬體分別儲存「正在執行」及「永久存放」的資料？並比較兩種儲存硬體的特性（容量、速度）。

6-4 分散式系統（Distributed Systems）

分散式系統是由多部獨立的電腦所組合而成的系統，彼此間藉由相互協調以完成相同的工作目標。因此分散式系統的特性，包含了系統間的分散性，使整體系統易於進行擴充，以增加處理與儲存的效能，並提升系統的可靠性。因此分散式系統有助於降低系統成本，並解決系統效能、可靠性以及空間不足等問題。

分散式系統應用在許多不同領域中，例如：在雲端運算領域，可以實現許多大規模的雲端應用，如Google Map即時路況分析等；在科學領域，可以用於處理大量需要計算的數據，如IBM WCG的網格運算；在金融領域則有透過區塊鏈技術實現去中心化帳本的比特幣、乙太幣等。

6-4-1 分散式系統的概念

分散式系統簡單來說就是將一部電腦系統上的工作，利用網路分給一部以上的電腦系統（或稱伺服器）同時分工合作並完成任務，而**雲端運算服務**（Cloud Service）即是分散式系統的應用之一。分散式系統的主要任務就是將**資料儲存**與**運算**的工作，透過網路分配給不同的電腦進行。事實上，相較於單一一臺配備好、效能高的電腦，分散式系統會顯得較有效率（分工合作）且可靠（具有容錯能力）。

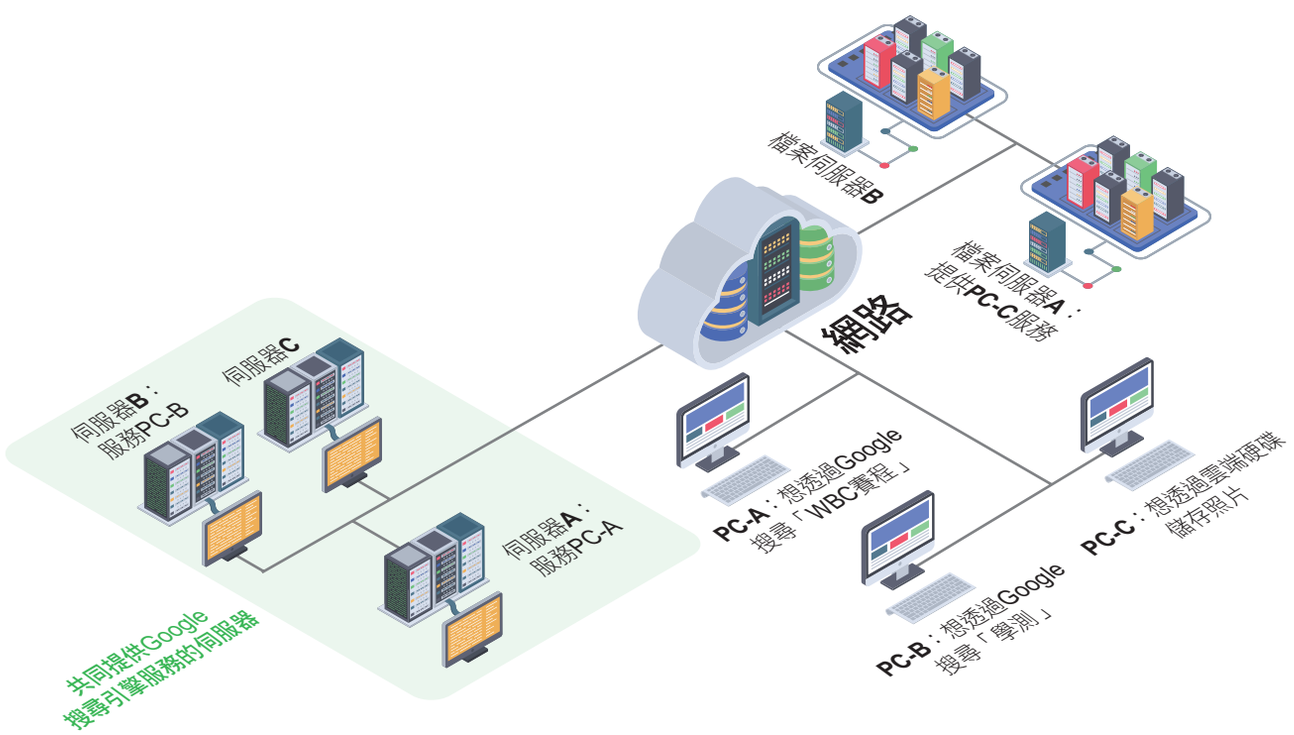


圖6-4-1 分散式系統示意

6-4-2 常見的分散式應用系統

雲端運算

由於近年來的網路發展大幅提升其傳輸速率和增加頻寬，因此我們可以透過網際網路，使用軟體、硬體與儲存空間。例如：Google提供強大的地圖功能，可以選擇所屬的交通方式，並計算出最佳路徑和抵達時間，甚至可以即時了解目前道路的壅塞程度等，其他如雲端硬碟、線上文件編輯、線上影音串流等都融入我們的日常生活與工作中。只要本地端電腦能上網，便可以隨時隨地存取遠端伺服器所提供的雲端服務，而不需要在本地端電腦安裝或下載許多軟體或資料，這些軟體和資料在網路上就可以直接使用。在下個章節我們將更深入地探討雲端運算的特點與應用。

網格運算

在科學或工程計算上常需要解決龐大且複雜的計算問題，我們希望透過網路將全世界大量的閒置電腦資源，提供給需要解決複雜計算問題的電腦，共同分工合作來解決問題，此種應用稱為**網格運算**。解決這類問題需要耗費大量的運算時間，如此龐大的資料量並非是多部電腦或單一部超級電腦就能夠勝任的工作，必須透過網格技術，整合大量分散各地的閒置電腦資源來縮減解決問題的時間。

World Community Grid（WCG）是由IBM公司所主持的一項大規模網格運算計畫——世界社群網格（圖6-4-2），該公司基於公益與協助科學研究的理念，讓全球各地願意加入這項計畫的閒置電腦，共同進行大型科學研究計畫的運算。使用者只要下載並安裝WCG官網提供的軟體即可在電腦閒置時，從遠端下載資料到本地端進行運算，而後再將結果回傳，目前已經成功完成許多大型研究計畫，例如：從數千萬種化合物分子找出能夠抵抗病毒的藥物配方，原先可能需要數百年才能運算完畢的研究結果，縮短為幾天內即可完成。

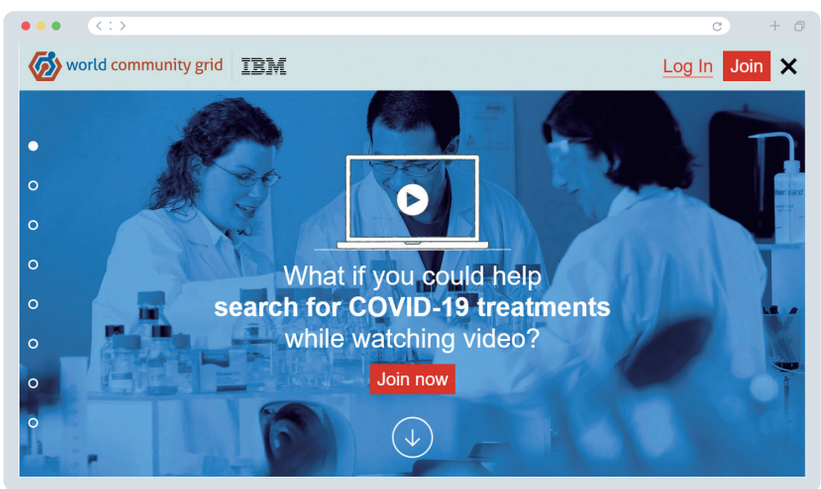


圖6-4-2 WCG對於COVID-19的研究計畫

以前，網格計算多應用於科學與工程領域，雲端運算則傾向於提供個人或商業上的服務；而目前雲端運算的計算能力隨著技術的演進，越來越強大，能將同等級的電腦整合，並提供較為穩定的運算服務。相較之下，網格運算必須處理、整合許多不同等級或系統的電腦，也是其逐漸式微、慢慢被雲端運算技術取代的原因之一。

科技即戰力

請同學至IBM WCG（https://www.worldcommunitygrid.org）的官方網站，找出三項最感興趣的研究計畫，並將研究計畫名稱記錄在下方。

1. _____

2. _____

3. _____

區塊鏈（Block Chain）技術

區塊鏈技術（以虛擬貨幣為例）即為分散式系統的一種應用，目前最成功的實務應用就是比特幣。我們可以將每個**區塊**（Block），視同為分散式帳冊中的每一個**分頁**，每一頁（區塊）可以記錄多筆交易，且分頁（區塊）都會與另一個分頁（區塊）產生鏈結，因而架構起整本帳冊（區塊鏈）。

有別於過去以銀行為中心、記錄每一筆交易的方式，這是一種當區塊鏈有交易產生時，除了雙方的帳本（區塊）有資料異動外，還會透過整個區塊鏈連結在一起的演算法機制（每一個區塊都會記錄上一個鏈結區塊的雜湊值），共同驗證這段連續不斷的交易紀錄檔，且每個區塊中的交易紀錄是無法僅經由其中一個節點來竄改的（圖6-4-3）。

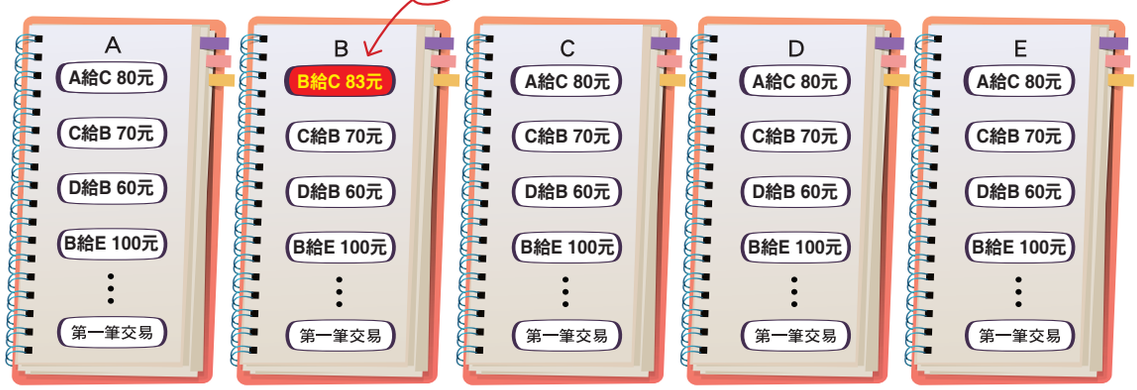


圖6-4-3 每個人都有一個帳本，記錄所有人的交易

IT 補給站

知名的比特幣就是運用區塊鏈技術作為底層技術，有別於過去雙方交易必須透過以銀行為中心的方式，運用分散式帳本技術造就了特點為「去中心化」、「不可竄改性」、「透明」的電子交易體系（圖6-4-4）。

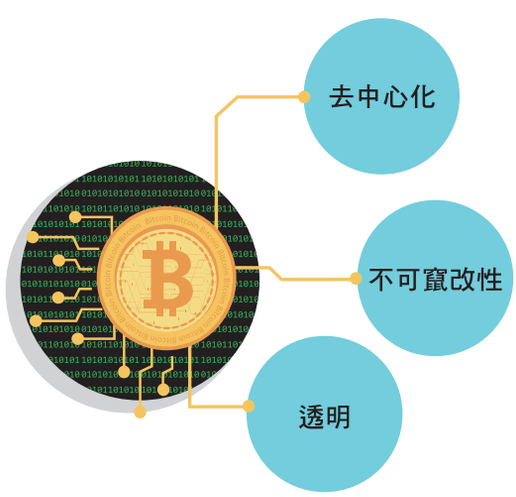


圖6-4-4 比特幣特性

課堂檢測

( ) 1. 下列何者為硬體與應用程式之間的介面平台？
(A)套裝軟體 (B)共享軟體 (C)專案軟體 (D)系統軟體。

( ) 2. 下列何者扮演路口交通警察的角色，控制資料流向以及解碼各項指令，並依指令指揮各單元間的運作與執行？
(A)ALU (B)MU (C)CU (D)IU。

( ) 3. 3D印表機是屬於下列哪一單元的硬體實作？
(A)MU (B)OU (C)CU (D)IU。

( ) 4. 遠距教學中的網路攝影機、麥克風等硬體是屬於下列哪一單元的硬體實作？
(A)MU (B)OU (C)CU (D)IU。

( ) 5. 將來自記憶單元的資料進行運算或是邏輯值判斷等工作，再將運算結果回存至記憶體，是屬於下列哪一單元的工作？
(A)ALU (B)MU (C)CU (D)IU。

( ) 6. 下列何者並非作業系統？
(A)Mac OS (B)MS Windows (C)Linux (D)MS Office。

( ) 7. 下列何者屬於專案開發軟體？
(A)Adobe Photoshop (B)學習歷程檔案系統
(C)CorelDRAW (D)威力導演。

( ) 8. 將一部電腦上的工作，利用網路分給更多電腦同時分工合作並完成任務是指下列何者？
(A)分散式系統 (B)協作平台系統 (C)多核心系統 (D)分時系統。

( ) 9. 下列何者不是使用分散式系統的優點？
(A)可資源共享 (B)可提高計算效能 (C)提高可信度 (D)提升網路效能。

( ) 10. 下列關於「分散式電腦系統」的敘述，何者不正確？
(A)使用者感覺上就像是在使用一部電腦
(B)具有優於大型主機（超級電腦）的延展性
(C)建置成本通常遠高於大型主機（超級電腦）
(D)將多部獨立電腦以網路匯集而成。

( ) 11. 下列哪一項不屬於雲端運算技術？
(A)網格運算 (B)分散式運算 (C)社群網路 (D)雲端硬碟。

( ) 12. 三個程序A、B、C已依序送入電腦等待執行，且它們所需的執行時間分別是10、8、13個單位時間。如果該電腦使用輪詢排程（RRS）演算法來執行該三程序，且每個時間切割為5個單位，則該三個程序的平均等待時間為若干單位？
(A)6.2 (B)14.33 (C)9.33 (D)8.67。

( ) 13. 在系統行程的生命週期當中，不同狀態間的變換不會有以下哪一種？
(A)從執行狀態進入預備狀態 (B)從預備狀態進入執行狀態
(C)從等待狀態進入執行狀況 (D)從執行狀況進入等待狀態。

(　　) 14. 下列五個程序P1、P2、P3、P4、P5的優先權及所需的計算時間如下表所示：

程序名稱	優先權	所需計算時間
P1	3	10
P2	1	1
P3	4	2
P4	5	10
P5	2	5

若作業系統使用優先權排程演算法，則這五個程序的平均等待時間為何？（四捨五入到小數點第一位）

(A)18.6　(B)8.8　(C)8.2　(D)7.2。

(　　) 15. 下列何者並非死結狀態的要素？

(A)不釋放資源　(B)循環等待

(C)可搶奪資源　(D)持有並等待資源。

(　　) 16. 下列哪一種「排程演算法」具有時間配額的設計？

(A)先到先服務（FCFS）　(B)最短工作優先（SJF）

(C)優先權（PS）　(D)輪詢排程（RRS）。

(　　) 17. CPU的排程中，為了讓執行程序有最短的平均等待時間，何者為最適當的排程方式？　(A)FCFS　(B)SJF　(C)RRS　(D)以上皆非。

(　　) 18. 若有三個程序P1、P2、P3都在時間0到達。假設P1、P2、P3之執行時間分別為24、4、2個時間單位。則在最短工作優先（SJF）排程演算法下，三個程序的平均等待時間為何？（四捨五入到整數）

註：一行程的等待時間為該行程到達至其執行結束的過程中，花費在等待其它行程的時間。

(A)2　(B)3　(C)5　(D)6。

(　　) 19. 某一作業系統之CPU排程採用先到先服務排程（FCFS）的方法，假設現在有三個程序分別為P1、P2及P3，其中P1到達時間為2秒、執行時間為16秒，P2到達時間為14秒、執行時間為20秒，P3到達時間為0秒、執行時間為5秒。請問在上述狀況中，三個程序的平均等待時間為何？

(A)13.67秒　(B)5.33秒　(C)3.33秒　(D)2.67秒。

(　　) 20. 假設有一個程序僅能使用三個實體記憶體分頁框，且該程序的分頁1、分頁2、分頁3已依序載入主記憶體中。考慮該程序的記憶體分頁使用順序如下：分頁1、分頁2、分頁3、分頁4、分頁1、分頁2、分頁3、分頁4、分頁1，若採用先進先出分頁替換演算法（FIFO），則產生分頁錯誤的次數為何？

(A)0　(B)3　(C)6　(D)9

chapter

07 網際網路運作原理

在系統平台中，網路所扮演的角色是藉由有線或無線的傳輸媒介，將多部裝置相互連結。如電腦、智慧型手機、印表機等裝置，只要透過網路，就能夠彼此傳輸檔案、傳遞訊息或進行設備共享。而在本章節中，我們將透過網路的架構、不同的通訊技術與通訊協定的介紹，了解網路運作的原理。

7-1 網路的架構

一般來說，網路架構可依據共享資訊的方式不同，分成對等式網路和主從式網路；依據連接方式分成不同的網路拓樸（星狀、環狀、匯流排等）；依據連接的規模亦可分為區域網路（Local Area Network，LAN）、廣域網路（Wide Area Network，WAN）。而網路的傳輸則有賴有線與無線網路傳輸技術的成熟，藉由傳輸媒介與網路通訊設備的開發，使網路傳輸更加穩定。

7-1-1 網路類型

網路共享資訊的方式

1. **對等式網路（Peer-to-Peer Network，P2P）**：對等式網路是一種去中心化的網路架構，每個網路節點同時可作為資源的提供者和接收者，直接和網路中的其他節點進行資源共享，故又可稱為**點對點網路**，如圖7-1-1。

2. **主從式網路（Client-Server Network）**：主從式網路又可稱為**客戶端—伺服器端網路**，是一種中心化的網路架構，客戶端向伺服器端發送資源請求，而伺服器端則提供客戶端所需服務，如圖7-1-2。

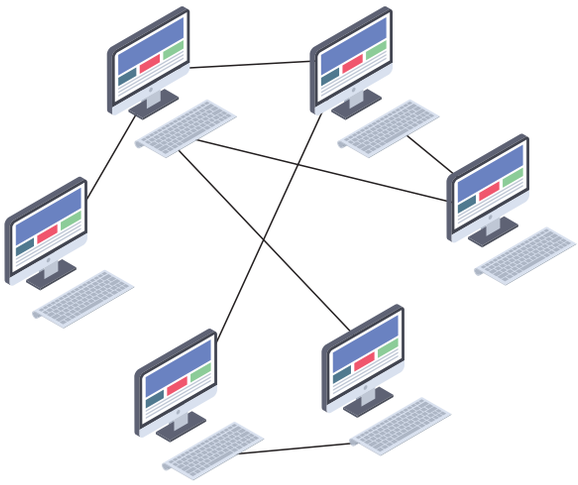


圖7-1-1 對等式網路架構

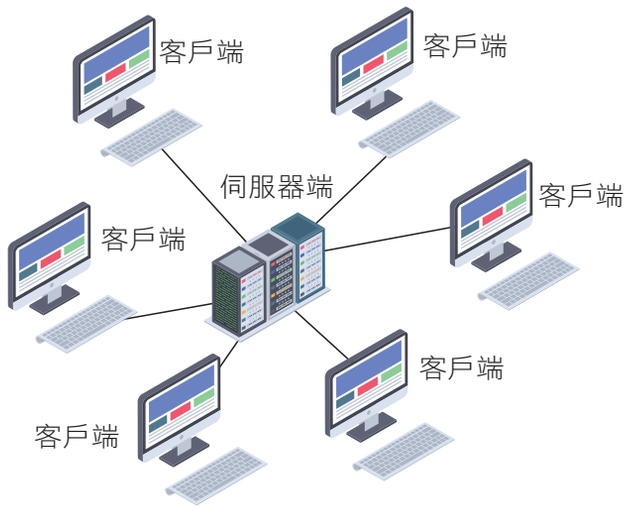


圖7-1-2 主從式網路架構

實體網路的連接方式

01 星狀網路

將所有資訊設備的連線集中於一部集線器，可靠性高，當單一設備故障時，不會影響整體網路的通訊。

02 環狀網路

將所有的設備串接在環狀的線路上，當環狀線路故障時，會導致網路的癱瘓，故常以雙環狀架構建置。

03 匯流排網路

將所有設備連接到一條主要的通訊線路，當主線路故障時，會導致整體網路的癱瘓。

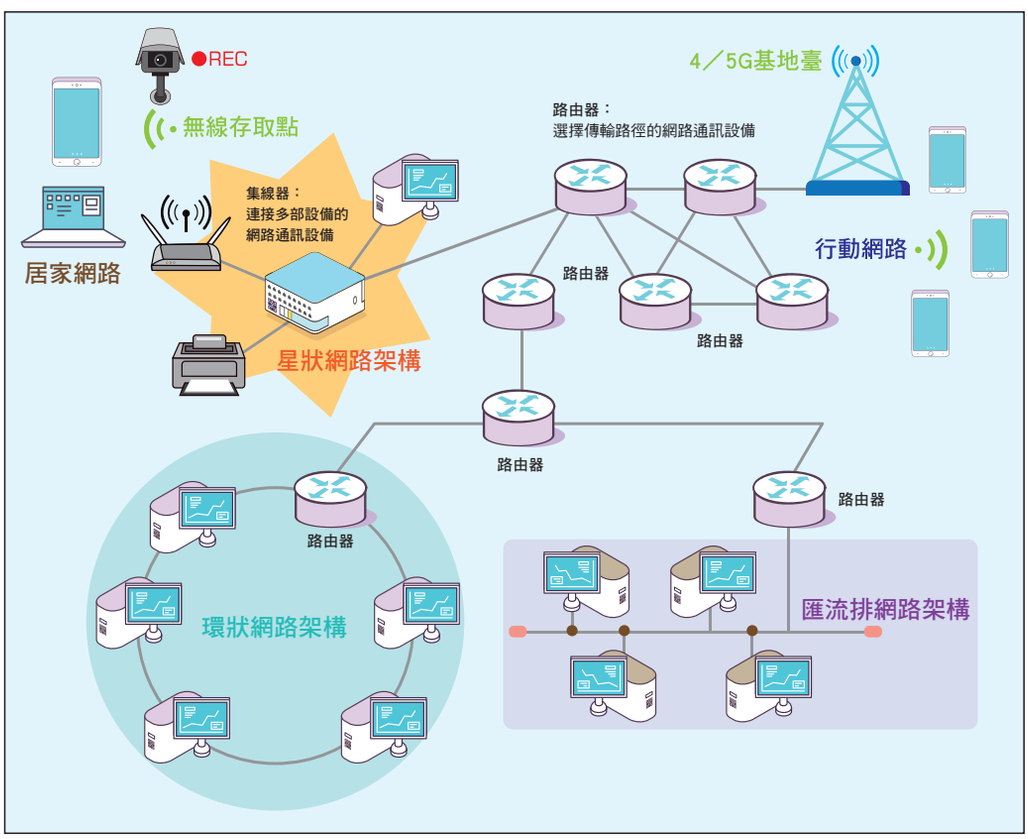


圖7-1-3 依據連接方式分成不同的網路拓樸

網路的範圍與規模

區域網路

一般規模不大的網路系統的範圍稱之為**區域網路**，例如：家庭、學校、商辦大樓等較小型的網路架構。

廣域網路

廣域網路是透過遠距離的方式，將位於不同地理位置的區域網路彼此連接。如海底電纜（圖7-1-4）等，其距離能夠跨接不同城市、國家甚至橫跨各大洲、各大洋，最終形成一種跨國界、涵蓋範圍廣闊的大規模網路建設。

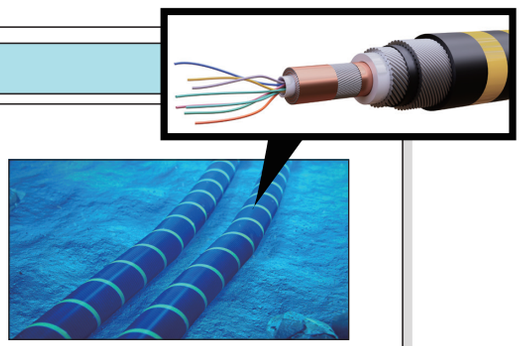


圖7-1-4 構成廣域網路的海底電纜

網際網路（Internet）是一種超大型的網路集合，其彙整世界各地不同規模的網路，如臺灣就有臺灣學術網路（TANet）提供教育機關與各級學校網路服務，而一般的企業與民眾則可透過中華電信、台灣大寬頻等固網業者（網際網路服務提供者，Internet Service Providers，ISP）所提供的連線服務與網際網路相連接。

7-1-2 網路的傳輸

網路的傳輸可藉由有線傳輸媒介連結不同的傳輸節點或設備，如雙絞線、光纖等，作為傳輸訊號乘載的媒介，而在傳輸過程中，常見的設備包含了交換器和路由器，分別負責訊號轉送與傳輸路徑選擇的功能。而隨著無線通訊技術的發展，無線網路可透過廣播無線電波、微波等作為傳輸訊號乘載的媒介，而在傳輸過程中，常見的設備包含了無線存取點和通訊衛星，進行訊號的接收與傳送（圖7-1-5）。

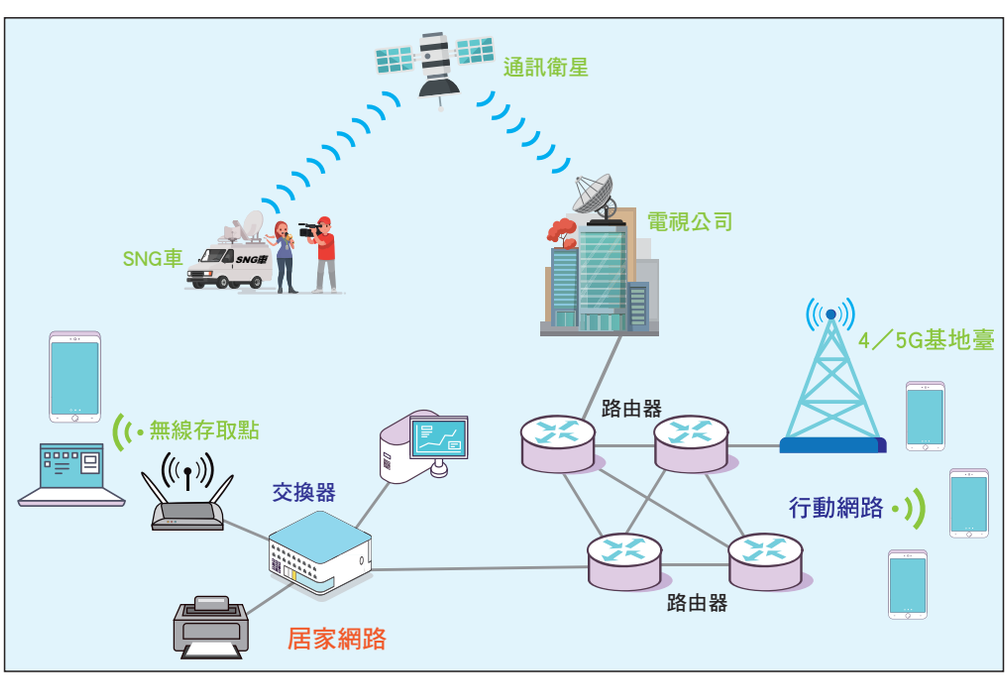


圖7-1-5 有線／無線網路傳輸的媒介與相關設備示意圖

有線網路的傳輸媒介

使用電纜線，如雙絞線、同軸電纜、光纖（圖7-1-6）等，連結可以彼此傳輸通訊的裝置，由於是使用實體線路建置，因此訊號較不受外在因素影響，是網路系統架構的基礎建設，如乙太網路等。過去幾年普遍認為有線網路頻寬速率較快，但近年全球專注於發展無線網路技術，兩者的頻寬速率已經不相上下了。

圖7-1-6 （左）RJ45 接頭與雙絞線；（中）BNC 接頭與同軸電纜；（右）ST接頭與光纖

雙絞線

01 線材裡面有八條分別被絕緣體包覆的銅線，且兩兩對絞，最外層再包覆絕緣體作為外皮，並以RJ45作為接頭，是一種在區域網路中常見的線材，佈線成本較低。

同軸電纜

02 線材裡面有一條銅線，銅線的外面有一層絕緣體包覆，絕緣體外又有一層較薄的網狀金屬導體，最外層再以絕緣體作為外皮，並以BNC作為接頭。常見於區域網路與有線電視業者佈線的線材，佈線成本較雙絞線高。

光纖

03 線材最內層是由玻璃或塑料所製成的纖維，而光在纖維當中以反射方式傳輸訊號。微細的光纖包覆在塑料中，使纖維能夠彎曲而不至於斷裂，並以FC、ST等作為接頭。常用於較長距離的骨幹網路建置。佈線成本通常為最高，但具有傳輸速度快、品質佳的特性。

有線網路連線設備——交換器與路由器

交換器可連接多部設備，目前已取代大多數的傳統的集線器，常用來建立網路，使其能夠相互進行資料的交換（圖7-1-7）；而路由器主要是用以連接網路，並且替資料選擇傳輸路徑的設備（圖7-1-8）。

圖7-1-7 交換器可用來連結設備，建立網路

圖7-1-8 家用的路由器多數也含括了AP的功能

無線網路的傳輸媒介

1. **廣播無線電波**：由於其波長傳送距離短但穿透性佳的特性，適合在建築物內或城市中架設基地臺通訊，例如：在家中建置無線網路基地臺（Access Point，AP），手機接收電信基地臺訊號等。
2. **微波**：由於其波長傳送距離遠但穿透性不佳的特性，必須在高樓或空曠處架設基地臺收發訊號，且常透過衛星轉送訊號，例如：播送即時新聞的SNG車（Satellite News Gathering）對通訊衛星進行連線，手機接收通訊衛星發送的GPS（Global Positioning System，全球定位系統）訊號等。

無線網路連線設備——AP

AP一般稱為「無線存取點」或「無線基地臺」，是一種無線網路訊號的發射與接收平台，主要作為有線網路與無線網路之間銜接的橋樑。現有的網路連線設備中，有些也整合了AP和路由器等功能。

科技即戰力

請回家確認看看，家裡是使用哪一間ISP業者的有線網路？牽到家裡的是哪一種線材？家裡面有哪些連線設備？

7-2 通訊技術的演進

古時的人們透過飛鴿傳書、烽火臺、旗語等傳統通訊方式，無需面對面溝通也能順利傳遞訊息；而近代通訊技術也一直是科技領域發展的關鍵要素，從早期的電報、電話，一直發展到現代的行動通訊、網路通訊等。在這個不斷發展的領域，從類比訊號進入到數位通訊時代，行動通訊更是成為發展重點，尤其是智慧型手機的普及，以及物聯網、雲端運算、人工智慧等應用的結合，都隨著**第五代行動通訊系統**（5G，5Generation）的發展，不斷地擴展並改變我們的日常生活。

7-2-1 通訊系統的演進

不同於有線網路使用的實體線路，只要透過無線網路基地臺的建置，就能將有線網路的訊號轉為無線網路的電磁波，使訊號更為彈性地延伸，讓筆記型電腦、手持裝置、車載系統等各類行動裝置的應用與發展，皆可隨著無線網路技術的成熟而更為便利及廣泛。

Wi-Fi

Wi-Fi（圖7-2-1）是由IEEE組織所定義的802.11的短距離無線傳輸規範，其技術主要是將有線網路連接無線存取點（AP）後，透過AP發送5GHz及2.4GHz的無線電波頻段來連接設備（如筆記型電腦、智慧型手機、平板電腦、無線印表機等）。而這些設備只要是位於無線網路涵蓋的區域內，都可以自由地挪動位置且持續維持網路連線。

Wi-Fi的主要優點包括了方便建置和連線、不需佈線、速度快且支援多種設備等；然而同時也存在一些缺點，例如：訊號穩定性不如有線網路，安全性較低。因此，在使用Wi-Fi時，使用者常需要設定安全措施來保護自己的網路和數據傳輸。

圖7-2-1 支援IEEE組織的無線網路規範之標章

行動通訊系統

行動通訊系統使用無線電波或衛星等無線傳輸技術，其涵蓋範圍較Wi-Fi廣闊，並能實現語音、簡訊、多媒體和影音串流等傳輸服務。依據技術的演進，分類如下表7-2-1所示：

表 7-2-1 1G至5G的通訊系統

第一代行動通訊系統（1G）	第二代行動通訊系統（2G）	第三代行動通訊系統（3G）	第四代行動通訊系統（4G）	第五代行動通訊系統（5G）
透過類比訊號傳輸，傳輸速度慢，主要用於語音通話。	開始採用數位訊號傳輸，可提供語音通話和簡訊傳輸的服務。	能提供圖片傳輸和網路瀏覽服務，傳輸速度較上一代快。	傳輸速度較上一代快，能夠讓用戶流暢地觀賞串流影片、進行視訊通話等。	傳輸速度較上一代快，能夠傳輸高畫質的串流影片且實現許多即時服務，如物聯網（Internet of Things，IoT）等。

「物聯網」相關內容將於後續8-3節詳加介紹

現代的行動通訊技術已完全融入我們的生活，藉此也發展出更多元化的應用與品質更佳的雲端服務，為人們提供更加便利且高效率的通訊服務。

此外，從2015年至今，SpaceX 公司更透過低軌道衛星群，提供覆蓋全球的高速網際網路存取服務，即為 Starlink 計畫（圖7-2-2）。

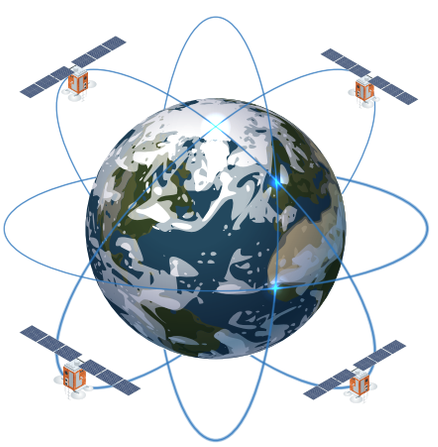


圖7-2-2
建置低軌道衛星群，提供覆蓋全球的網際網路存取服務。

7-2-2 無線感測網路（Wireless Sensor Networks，WSN）

無線感測網路可透過許多**感測節點**所構成的無線網路系統，進行環境的偵測與數據的蒐集。感測節點包含**感測器**和**無線通訊模組**，可偵測環境中的目標物，並運算蒐集到的數據，將處理過後的資料內容，以無線通訊模組送到雲端的資料匯集中心。感測器具有**低成本**、**低耗電**的特性，因此很適合在各種環境下大量布署建置，如農業上的使用（圖7-2-3）。

圖7-2-3 農業使用無線感測器

無線感測網路最初是應用在美國軍事科技的研究計畫，透過微小的裝置，能在不驚擾敵軍的情況下，如同蒼蠅般地深入敵方陣地，攝錄畫面等情資，並即時回傳影像資料。

而除了軍事上的應用外，也發展出更多元的應用，如監測環境、生理數據監測、農業感測、航空系統控制等，如臺灣將部分主要橋梁安裝了位移感測裝置，藉以即時提醒用路人橋梁的安全與否；在農業上，也透過感測器進行全天候即時的溫濕度監測，並將大規模的數據透過無線傳輸，匯入資料庫。伺服器將蒐集到的數據分析後，主動進行相對應的措施，進而達到集約且智慧化的農業（圖7-2-4）。隨著無線感測網路有越來越多的創新應用，也更能達到規模經濟，增進生活上的便利性。

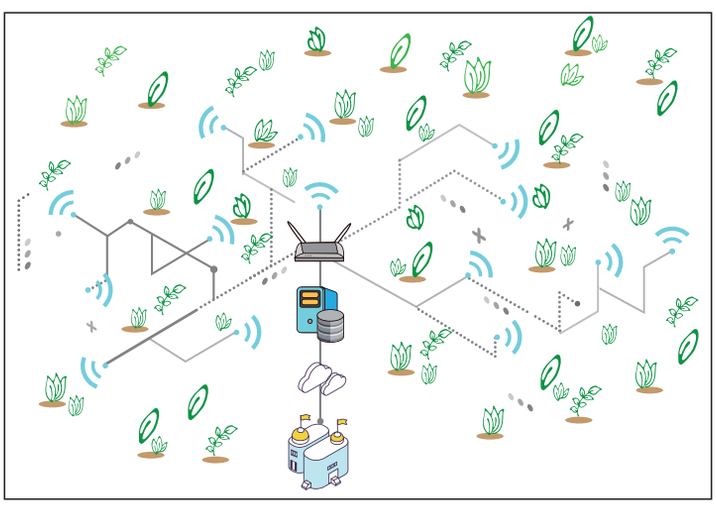


圖7-2-4 無線感測網路架構圖——以農業為例

科技即戰力

1. 你的智慧型手機是使用哪一家ISP業者的通訊網路？是屬於第幾代通訊網路呢？
2. 想一想：居家生活中有哪些用來傳送資料的無線設備？這些發出訊號的設備通常會分別使用到哪些頻段呢？

7-3 通訊協定

通訊協定是指使不同裝置間能彼此進行通訊的規則，例如：兩國元首會面前，必須先約定好時間、使用語言，以及討論議題等。而網路技術發展至今，全世界得以透過共通的規則使彼此互通有無，所仰賴的也是網路上的通訊協定，常見的通訊協定組合如目前應用最為廣泛的**TCP／IP協定組合**，與ISO組織規範的**OSI七層模型**，透過七層的規範，提供並區分相對應的通訊協定準則。

7-3-1 封包（Packet）

封包廣義上是指在網際網路上傳遞的資料訊息，而狹義的封包定義是指在TCP／IP協定中，傳送至**網路層**階段的資料訊息。

7-3-2 TCP／IP通訊協定組合

TCP／IP通訊協定組合指的是一群通訊協定的組合，此規範一共將其協定組合劃分為四個層級，由上而下分別為**應用層**、**傳輸層**、**網路層**、**連結層**。

應用層（Application Layer）

定義應用程式的規範，如超文本傳輸協定（Hyper Text Transfer Protocol，HTTP）、簡易郵件傳輸協定（Simple Mail Transfer Protocol，SMTP）、檔案傳輸協定（File Transfer Protocol，FTP）、虛擬終端機模擬協定（TELNET）等。

傳輸層（Transport Layer）

負責兩節點間的資料傳輸，如傳輸控制協定（Transmission Control Protocol，TCP）、使用者資料協定（User Datagram Protocol，UDP）等，可將應用層的資料切割成大小相同的區段，並且在傳輸後確認區段已送達對方。而傳遞的資料在這一層可稱為**區段**（Segment），先將每一個區段進行編號，目的端收到後再依照這些編號重新組合資料。

網路層（Internet Layer）

負責選擇封包傳輸的**最佳路徑**，如網際協定（Internet Protocol，IP）。而傳遞的資料在此層可稱為**封包**。

IT 補給站

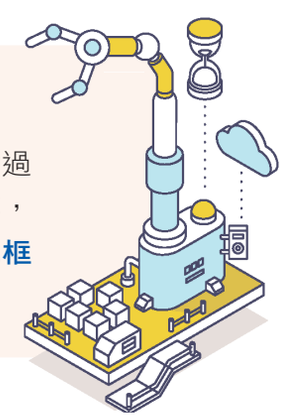
IPv4 與 IPv6 的差異

IP位址（IP Address）為網際網路上的通訊地址，每個能夠連上網際網路的裝置都必須擁有一個IP位址才能與外界進行資料的傳輸。傳統的IPv4提供了2^{32}個IP位址，但由於進入了物聯網時代，需要大量的IP位址提供給眾多裝置，IPv4位址的數量漸漸不敷使用，IPv6也應運而生。IPv6提供了2^{128}個IP位址，可以解決IPv4位址不夠的問題。而目前IPv4透過許多IP位址整合與劃分機制，如supernet、subnet等技術，節省了許多IPv4位址的浪費，目前路由機制介於IPv4與IPv6並行運作的時期。

	IPv4	IPv6
IP位址數量	2^{32}個	2^{128}個
表示方式	4組十進位數字(0~255)，如210.60.36.204	8組16進位數字，（連續兩個冒號代表冒號中間有連續1組以上的0）如fe80::dc77:45e2:da4c

連結層（Link Layer）

連結層是負責將網路層的封包從本地裝置傳送到目的端裝置的網路層的過程，也是負責將封包轉為實際傳輸訊號的規範，如乙太網路、Wi-Fi等，也定義了傳輸媒介與硬體裝置的規範。而傳遞的資料在此層可稱為**訊框**（Frame）。



科技即戰力

請將下列(A)～(D)選項依TCP/IP四層協定的規範，分別填入右側表格相對應的空格中：

(A)IPv6　(B)光纖纜線　(C)Instagram APP　(D)將封包編號

應用層	
傳輸層	
網路層	
連結層	

7-3-3 OSI模型（Open System Interconnection Model）

OSI模型是由國際標準組織ISO所提出的概念模型，將其定義為七層的架構，由上而下分別為**應用層**、**呈現層**、**會議層**、**傳輸層**、**網路層**、**資料連結層**、**實體層**（表7-3-1），再透過這七層的概念與架構發展出相關的通訊協定。

表 7-3-1 OSI 七層網路架構

層級	名稱	功能
第七層	應用層（Application Layer）	規範使用者所接觸到的應用程式，如網頁、遠端連線、檔案傳輸、電子郵件，資料庫管理系統等。
第六層	呈現層（Presentation Layer）	資料的壓縮／解壓縮、加密／解密、編碼／解碼等。
第五層	會議層（Session Layer）	管理兩端點間的連線。同步／非同步等。
第四層	傳輸層（Transport Layer）	資料傳輸錯誤的監測和還原，資料封包的編號。
第三層	網路層（Network Layer）	在網路封包傳輸時，網路層負責封包的「路由」，也就是選擇封包傳輸的路徑。因此像網路通訊設備中的路由器，就是透過這一層來進行規範。
第二層	資料連結層（Data Link Layer）	規範流量控制、錯誤偵測、傳輸媒體存取控制等。
第一層	實體層（Physical Layer）	規範實際傳輸的硬體設備，如線材、端點接頭、訊號類別等。

OSI模型的分層僅止於概念上的定義與規範，並非如TCP／IP是一群已制訂且實作完成的通訊協定組合（圖7-3-1），但OSI模型的存在，對於學習網路架構或是分層定義上的釐清，具有指標性的意義。

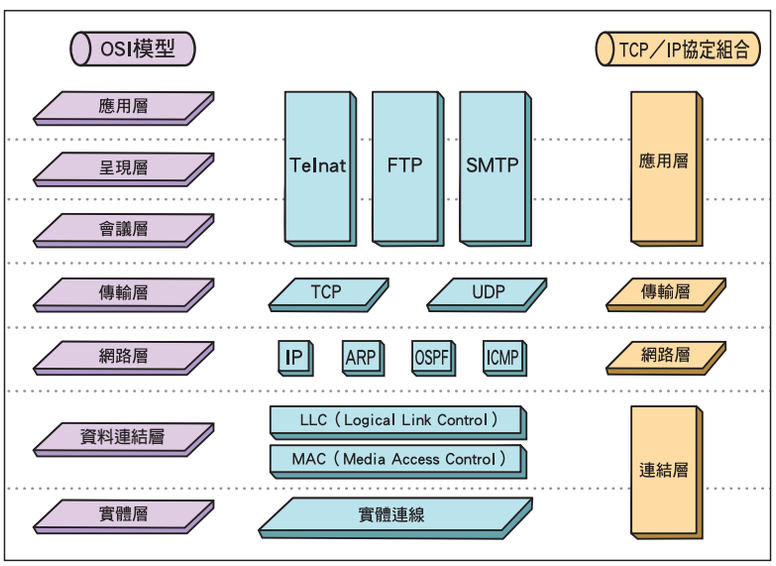


圖7-3-1 OSI模型與TCP／IP協定組合對應圖

7-4 網路路由

網路路由是在網路上傳送封包的技術，當封包傳送至網路時，路由器會檢查其目的地位址，並決定如何傳送封包到目的地的路徑。透過這種方式，路由器可以幫助封包在網路上傳送，進而實現網路通訊。

階層式的路由方法

郵局替我們傳遞國際信件時，並非從寄件人這端點直達收信人信箱，而是在寄件人至地方郵局投遞信件後，先轉送至本國國際郵件處理單位，依序再轉送至對方所屬的國家郵政總局、地區郵局，郵差再進行最末端的派送（圖7-4-1）。而路由也是如此，網路中的路由指的就是封包（信件）傳輸路徑，封包自來源端發送後，透過一部以上的路由器（郵局）選擇傳輸路徑再進行轉送，最後抵達目的端。

階層式的路由方法是將整個網際網路中的路由器，依所屬層級的不同，大致上可分為三層級（最低層級、中間層級、最高層級），分別負責各層封包路由的選擇，視網路如一個倒立的樹狀結構，最低層級的區域網路內會有一部路由器負責封包的轉送，如各縣市內部信件轉送；若有需要往區域網路外部傳輸，向上還有中間層級的路由器，負責不同區域網路間的封包轉送，如各縣市間信件轉送；而如果網際網路的規模甚大，我們還需要更高層級的骨幹網路路由器連結中間層級的各地路由器，負責中間層級間封包的轉送，如國際信件轉送。

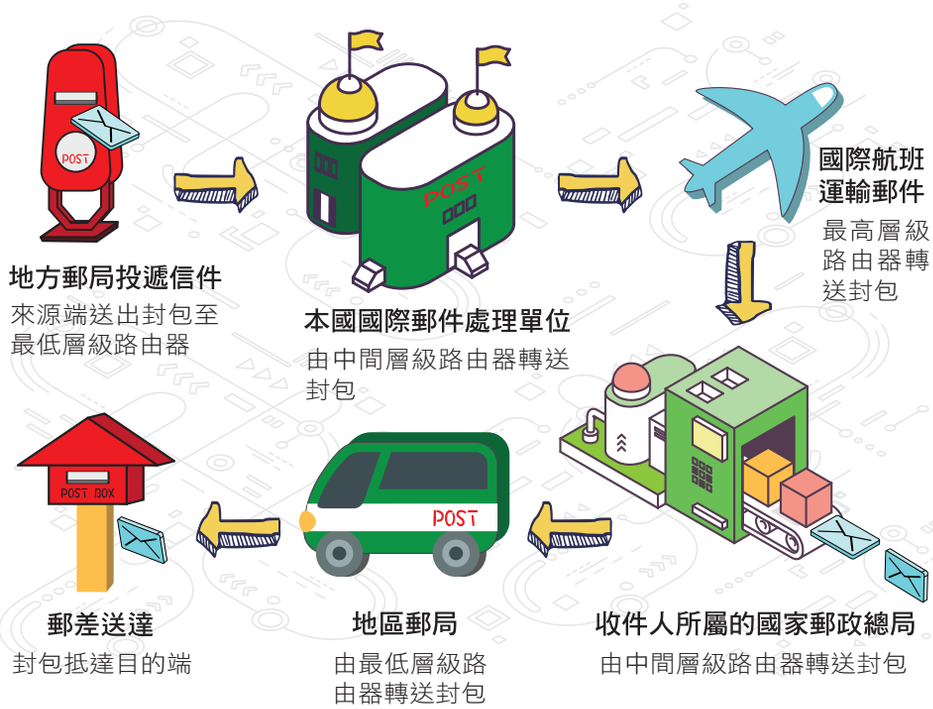
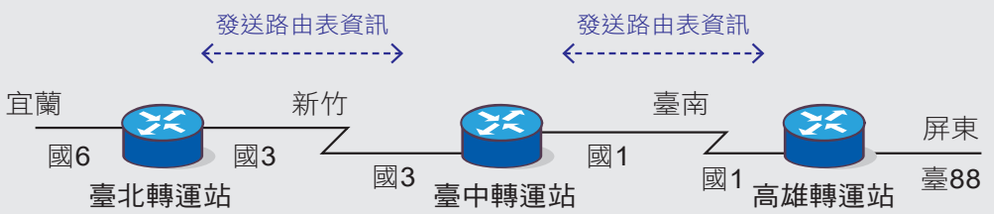


圖7-4-1
國際信件傳遞與跨國封包轉送的過程

常見的路由演算法有RIP、OSPF、BGP等，說明如下：

RIP 路由協定

RIP（Routing Information Protocol）**路由協定**是目前最為廣泛使用的一種路由協定。RIP路由協定是**距離向量**（Distance Vector）演算法的一種應用，其所選擇最佳路由的決定方式，是計算來源端至目的端的傳輸方向所經過路由器的數量（成本，如圖7-4-2），同時每30秒會把自己的路由表（Routing Table）資訊發送給相鄰的路由器，也因為方法直覺且簡單，因此設備實作容易。但是當傳遞路徑所經過的路由器數量較多時，即使此路由路徑的頻寬速率較快，也會被認為是效能較差的選擇，較適合用在規模較小的網路架構。



臺北轉運站路由表		
目的地	道路	轉運次數
宜蘭	國6	0
新竹	國3	0
臺南	國3	1
屏東	國3	2

臺中轉運站路由表		
目的地	道路	轉運次數
新竹	國3	0
臺南	國1	0
宜蘭	國3	1
屏東	國1	1

高雄轉運站路由表		
目的地	道路	轉運次數
臺南	國1	0
屏東	臺88	0
新竹	國1	1
宜蘭	國1	2

圖7-4-2 以RIP路由協定為例，各轉運站路由表的轉運次數即為其成本。

OSPF 路由協定

OSPF（Open Shortest Path First）則是透過一套較為全面的演算法，除了考量路由器數量外，也加入了各路徑的頻寬速率（成本，如圖7-4-3）等因素，計算出兩端點間各個路由的連線品質以建立路由的樹狀結構（圖7-4-4），並在設備更動時才發布路由表訊息給鄰近路由器節點，所以OSPF的路由效率較RIP佳，且適合規模較大的網路架構。

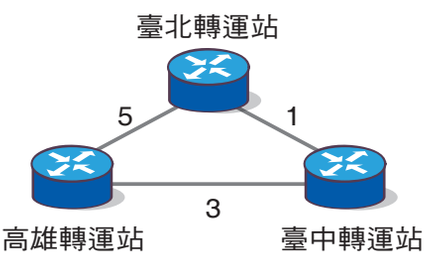


圖7-4-3 各轉運站間的成本示意。成本越大代表速度越慢。

轉運站名	臺北	臺中（臺北→臺中）	高雄（臺北→臺中→高雄）
成本	0	1	1 + 3 = 4 < 5（臺北—高雄），故選擇經過臺中轉運站後再到高雄的路線較快。

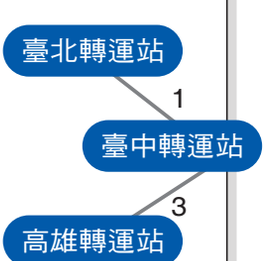


圖7-4-4 以OSPF建構臺北轉運站至其他轉運站的路由樹狀結構（可先以Dijkstra's Algorithm求得最短路徑，轉運站間的數字為兩站間的最短路徑）。

BGP 路由協定

邊界閘道協定（Border Gateway Protocol，BGP）是屬於**網域（自治系統）**和另一網域（自治系統）之間的路徑選擇協定，在實務上是網際網路服務提供者（ISP）之間的路由選擇協定，是一種**外部閘道協定**（Exterior Gateway Protocol，EGP），負責在一群路由器所屬的自治系統範圍外的路徑選擇（圖7-4-5）。因此在單一自治系統內（ISP之內）的路由選擇是透過**內部閘道協定**（Interior Gateway Protocol，IGP）即可進行路由選擇，也就是在一群路由器所屬的網域（自治系統）範圍之內進行路由選擇，如前面介紹過的RIP、OSPF等。

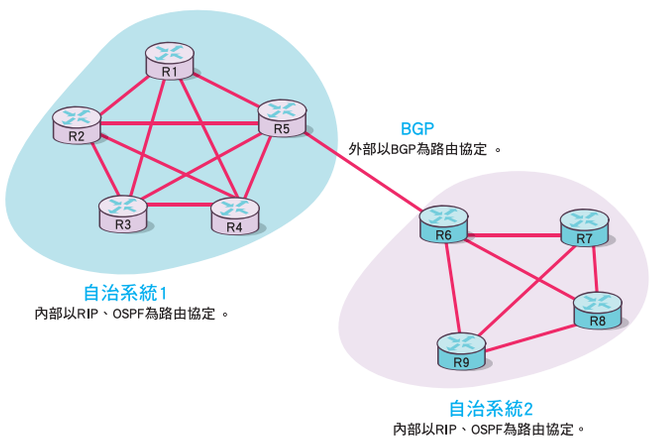
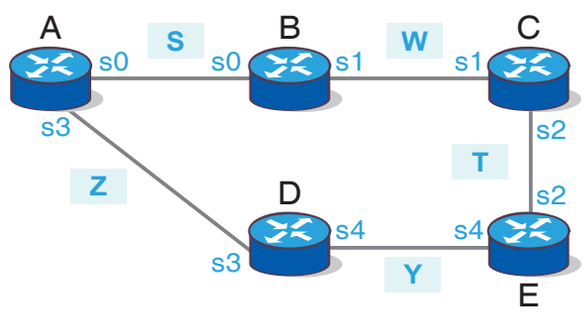


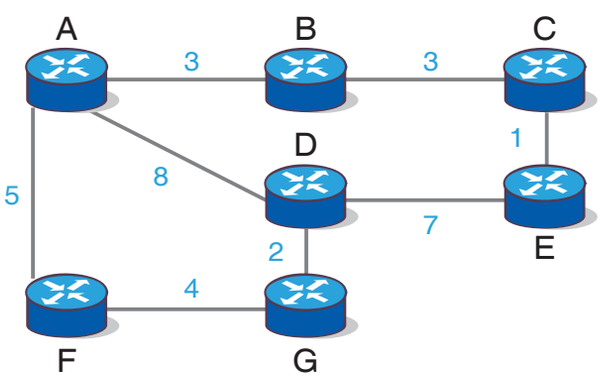
圖7-4-5 自治系統內外部路由協定

科技即戰力

1.右圖有五座城市S、Z、W、Y、T，而s0～s4為城市間的道路名稱，A～E為轉運站。請以RIP路由協定，分別列出A～E的路由表。



2.右圖有七個轉運站A～G，而數字為轉運站間的成本（成本越高，代表乘車耗費的時間越長）。請以OSPF路由協定，分別列出A～G的路由樹狀結構。（可參照圖7-4-4先運用Dijkstra's Algorithm求得最短路徑）



3.請同學查出家裡或學校的電腦是透過哪些網路相關設備連線至網際網路，並將連接網際網路所經過的設備與路線繪製下來。

課堂檢測

（　　）1. 下列何者屬於ISP業者？
(A)Instagram　(B)YouTube Premium
(C)中華電信　(D)KKBOX。

（　　）2. 樂樂是某公司資訊部門的人員，想替公司建構一區域網路，於是分析不同線材的單位成本，並將成本分析提供給公司主管作為決策的評估項目之一。試問下列何者為正確的成本順序？
(A)光纖＜同軸電纜＜雙絞線　(B)光纖＜雙絞線＜同軸電纜
(C)同軸電纜＜雙絞線＜光纖　(D)雙絞線＜同軸電纜＜光纖。

（　　）3. 恒宇傳遞訊息給遠在美國的家人，訊息內容提到「將文字化為電壓訊號，並透過海底電纜傳送到遠在美國的你。」請問引號中（劃線處）的敘述較屬於TCP／IP的哪一層規範？
(A)應用層　(B)連結層　(C)傳輸層　(D)網路層。

（　　）4. 若想要在家中建置無線區域網路，請問需要下列哪一項設備？
(A)Acess Point（AP）　(B)Cellphone Point（CP）
(C)Gateway Point（GP）　(D)Broad Point（BP）。

（　　）5. 在智慧型手機中，如「健保快易通」、「Instagram」、「台鐵e訂通」等APP是屬於下列哪一TCP／IP層級的規範呢？
(A)傳輸層　(B)連結層　(C)應用層　(D)網路層。

（　　）6. 易易剛才透過通訊軟體傳了多則訊息出去，卻發現部分訊息傳送失敗。易易請廠商查驗了一下，發現網路連線並未中斷，且路由器皆正常運作，請問這最有可能是TCP／IP的哪一層出了問題？
(A)應用層　(B)呈現層　(C)傳輸層　(D)會議層。

（　　）7. 下列何者除了考量路由器數量外，也加入了各路徑的頻寬速率等因素，計算出兩端點間的各個路由的連線品質，建立路由表？
(A)RIP　(B)OSPF　(C)BGP　(D)IGP。

（　　）8. 電信業者所開發的第五代通訊系統，即5G，請問「G」指的是下列何者？
(A)Giga Bytes　(B)Generation
(C)Giga Hz　(D)Giga Bbits。

（　　）9. 佩佩在2022年初購買了同年剛生產上市的智慧型手機，作為自己的生日禮物。請問這支手機最可能不支援下列何種訊號？
(A)5G　(B)3G　(C)5GHz　(D)2.4GHz。

（　　）10. 阿佑趁著暑假到叔叔的「達達大寬頻」公司實習，叔叔說他們這些地區性的有線電視臺業者，最常拉＿＿＿＿線材到客戶的家裡。請問空格內最有可能填入的是？
(A)光纖　(B)雙絞線
(C)電話線　(D)同軸電纜。

( ) 11. 無線感測網路常用於感測周遭環境，試問感測器節點具有下列哪些特性？
(A)低耗電、低成本、可支援少量感測節點
(B)能以遠端控制、高成本、可支援少量感測節點
(C)低耗電、低成本、可支援大量感測節點
(D)能以遠端控制、高成本、可支援大量網路節點。

( ) 12. 凱凱喜歡征服世界各地人煙罕至的險峻高山，假如他在登山時，想用智慧型手機連接網際網路，請問下列何種技術最為合適？
(A)無線網路 (B)衛星定位系統
(C)第五代通訊系統 (D)低軌道衛星通訊技術。

( ) 13. TCP／IP協定組合分為四層，簡單郵件傳輸協定（SMTP）屬於下列哪一層？
(A)應用協定層 (B)傳輸協定層
(C)跨越網路協定層 (D)實際協定層。

( ) 14. 有關TCP／IP網路架構的網路層，下列敘述何者錯誤？
(A)網路層負責將封包從來源端電腦送至目的端電腦
(B)網路層用到的位址稱為IP位址
(C)網際網路協定是網路層使用的一個協定
(D)網際網路協定保證同一訊息的封包能依原寄送的順序抵達目的地。

( ) 15. 下列有關IP之敘述，何者錯誤？
(A)IPv4位址中間用「.」隔開
(B)IPv4全長共64個位元
(C)IPv6全長共128個位元
(D)203.72.100.24是正確的IP位址。

( ) 16. 在OSI模型中，負責決定訊息傳輸路由者是哪一層？
(A)實體層 (B)連結層 (C)網路層 (D)傳輸層。

( ) 17. 在OSI模型的網路架構下，假如碰到瀏覽器跳出「無法選擇合適的編碼規則進行解碼」的錯誤訊息，則最有可能是哪一層發生了問題？
(A)資料連結層 (B)應用層 (C)呈現層 (D)會議層。

( ) 18. 下列何者利用計算來源端至目的端的傳輸方向所經過路由器的數量，決定路由路徑？
(A)RIP (B)OSPF (C)BGP (D)IGP。

( ) 19. 某公司內部所使用的網路路由協定，只會在設備更動時才發布路由表訊息給鄰近的路由、進行資訊更新。請問該路由協定最有可能是下列何者？
(A)RIP (B)BGP (C)OSPF (D)IGP。

( ) 20. 在不同的ISP（如臺灣學術網路、中華電信光世代等）之間，是以下列何者做為彼此間的路由協定呢？
(A)RIP (B)OSPF (C)AGP (D)BGP。

chapter

08 系統平台之未來發展趨勢

在系統平台中，由於硬體設計的突破、軟體技術的創新和網路效能的提升，讓許多科技的應用得以延伸並且持續發展。例如：**人工智慧**（Artificial Intelligence，AI）是讓電腦系統能效法人類思考、解決問題，並具有自主學習、決策判斷的能力；雲端運算（Cloud Computing）即分散式系統的應用與實踐，使我們得以透過網路，即時取得軟體、開發平台、基礎設施等服務的應用；而物聯網技術的應用，更讓科技與生活的連結愈加緊密，為人類的日常生活帶來更多的效益。

8-1 AI 人工智慧

日常生活中，我們每天都在使用人工智慧，如影音平台Netflix以人工智慧推薦下一部電影時；Siri或Google助理以人工智慧回應我們的命令時；當iPhone以人工智慧擷取到我們的臉部特徵並解鎖時，或是Facebook正確地推薦朋友在照片上標記等，這些都是人工智慧的應用。

人工智慧的發展並非一開始就如現今能進行許多複雜條件的決策判斷，而是在專家系統問世後，大致可分為三個時期的發展，才有當前的成果，且正在持續演進中。

第一階段——專家系統

工程師試圖將定義好的**特徵**（Pattern）和**模型**（Model）作為提供系統決策的依據，但是每當遇到例外情況，系統便無法進行正確的決策判斷，因為有些問題是無法以數值衡量的方式解決，這時期的代表系統為**專家系統**，例如：地震的預測，目前尚無確切的證據能預測地震即將到來。

第二階段——機器學習（Machine Learning）

工程師開始嘗試建立各種機器學習的模型，並以**監督式**、**非監督式**[1]等方式訓練模型從而解決問題，其概念是在人類的思維架構下，輸入大量的資料樣本（帶有**標籤**或不帶標籤[2]）到系統中，讓系統依據**模型**中的特徵自行判斷，透過這種方式，人們意外地發現機器能衍生合適的模型，且不斷地藉由現象學習調整決策，這就是**機器學習**的概念。巨量資料是提供訓練模型的來源之一，機器學習是資料的分析方法，從資料中萃取出複雜的規則，讓電腦根據這些規則發展出近似人類智慧的行為，人工智慧自此有了重大的突破。

由於機器學習的設計是為了解決特定問題而開發的演算法，通常是需要透過人類在特定領域的經驗，把大問題細分成許多子問題，待一個個解決後再重新組合起來，汲取特徵並建立模型，但此種模型仍有過多的人類主觀因素所建立的特徵，因此思考方式並無全面性，於是，人工智慧開始邁入了第三階段發展時期。

第三階段——深度學習（Deep Learning）

深度學習與機器學習的不同在於研究學者開始發展**深度類神經網路**（Deep Neural Network）的模型，並以巨量資料集訓練這些模型。以往的機器學習在學習特徵方面有其能力的上限，但研究學者發現透過深度學習的方式，可以在許多領域顯著提升巨量資料進行特徵學習的效果。由於其模型內的特徵也是透過大量的資料樣本學習而產生的，這種做法可以給模型更多自我調整的機制，且不需要人工介入處理，即可建立模型，因此也稱為**特徵學習**（Feature Learning）。

1.・監督式學習（Supervised Learning）：
在訓練的過程當中，輸入大量的樣本資料進入模型，同時也提供樣本資料的答案，此答案即為「標籤」。例如：先後把各500張照片輸入模型，這些照片分別標有狗狗與貓咪的標籤，並且在訓練後，使用尚未輸入過的照片進行模型測試。

・非監督式學習（Un-supervised Learning）：
訓練資料沒有標準答案、不需要以人力事先輸入「標籤」，所以機器在學習時並不知道分群結果是否正確。 在訓練過程中，只向機器提供資料範例，它會自動從這些範例中找到潛在的規則。

2.標籤（Label）猶如提供參考答案給系統，例如：提供兩張圖，一張標籤為西瓜，一張標籤為火龍果。

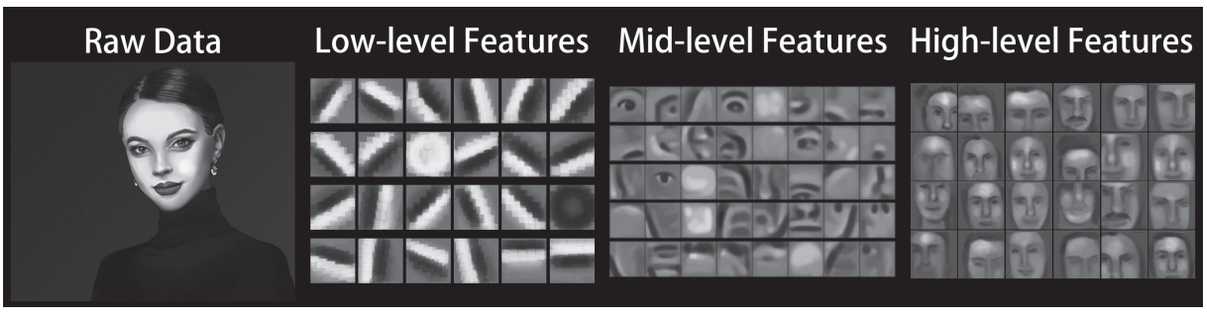


圖8-1-1 使用深度學習進行人臉辨識，從基礎的特徵如邊緣、角落、斑點、線條等，到中段的紋理特徵，如眼睛、鼻子、耳朵等等，到更宏觀的特徵——辨認是否有人臉的輪廓，最後透過此特徵判斷圖片是否有出現人臉。

參考資料：Rene Meyer, Deep Learning Smarts Up Your Smart Phone, AMAX (2015)

2016年，AlphaGo的誕生與其強大的對弈能力，讓全世界都為之驚艷，使用的就是深度學習的技術。因此，深度學習就成為近幾年機器學習中成長最快、最受矚目的技術。如智慧型裝置的語音辨識系統、文字翻譯、影像辨識、智能客服、垃圾郵件分類，都是深度學習實務上的應用。

在人工智慧發展的第一階段——專家系統，需要由人類定義特徵和模型；到了第二階段人工智慧發展的機器學習，只需要人類定義特徵，電腦自己會找出模型；而深度學習的出現，將模型、特徵的制訂都交由人工智慧系統自行演算和推論。

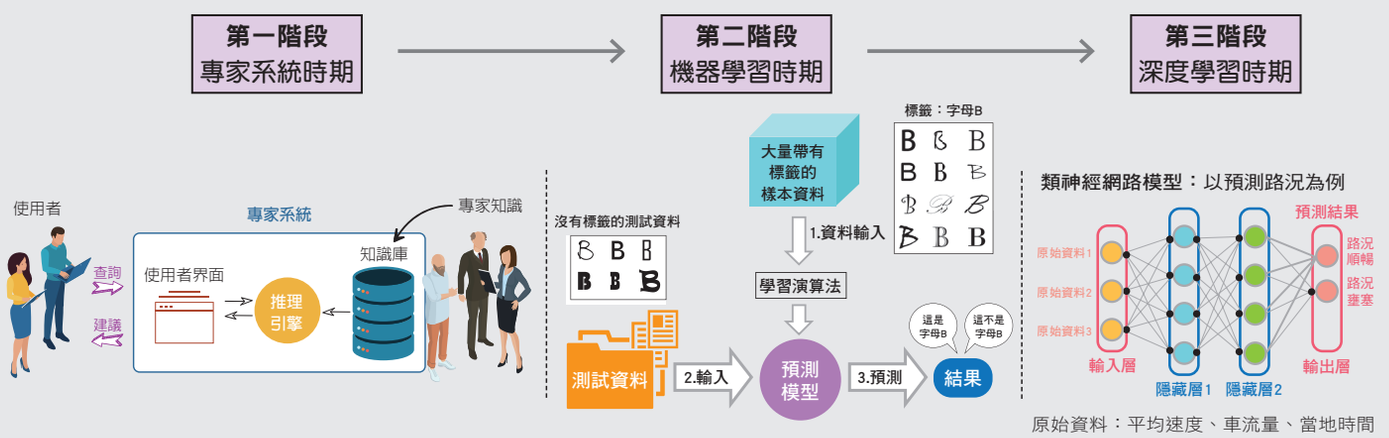


圖8-1-2 人工智慧演進的三階段與其相關分析模型

無論是從商業、醫療保健乃至於教育等，人工智慧的相關應用發展愈發迅速。政府與產業基於人工智慧開發上的公眾考量，也必須制定相關原則與承諾，而這些開發原則被稱為「負責任的人工智慧實踐（Responsible AI Practices）」。其包含人工智慧系統中，所有關於公平性、可解釋性、隱私性和安全性的相關議題與規範，相關說明如下。

參考資料：Google AI（https://ai.google/）

公平性 Fairness

隨著人工智慧對社會各領域的影響與日俱增，對所有人開發一個「公平」與「包容」的系統，就顯得至關重要。例如：運用人力資源系統挑選經理人時，人工智慧可能會從過去的企業主管面試資料中學習，導致其偏愛男性候選人。

可解釋性 Interpretability

要如何向人類說明人工智慧所做出的決策，將會是一大挑戰。即使是機器學習的專家，對於複雜的決策——例如：以深度學習（類神經網路）判斷癌症的復發——也可能需要花費相當多的時間，才能了解人工智慧的整體決策過程。

> 關於「機器學習」，Chapter 10 將會有更詳盡的說明。

隱私權 Privacy

使用「具敏感性的資料」訓練或輸入機器學習模型時，需要妥善保護隱私權。例如：人工智慧在辨識腫瘤細胞時，要向病患蒐集醫療影像，因此必須在符合法律和社會的規範下，發展出符合社會期望的智慧醫療系統。

安全性 Security

辨識潛在威脅，並且透過不斷地模擬攻擊、進行相對應的防範措施，進而維持人工智慧的系統安全，防範試圖欺騙人工智慧系統的攻擊者。例如：下圖8-1-3，攻擊者在圖片中添加一些干擾數據，人類視覺上無法察覺圖片的變化，卻能使人工智慧系統誤判其圖片分類。

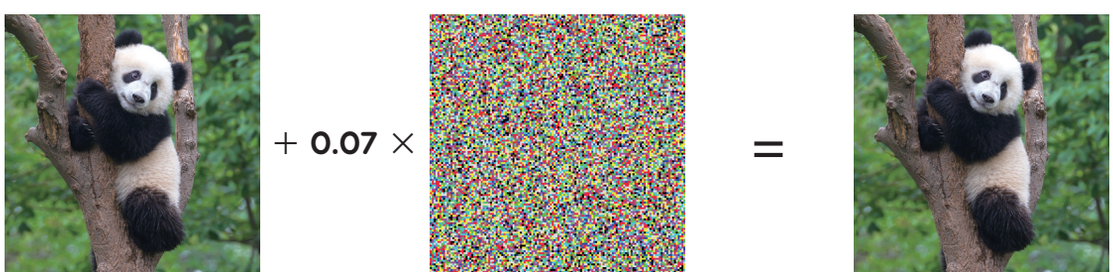


圖8-1-3 將X原圖貓熊（panda）加上線蟲（nematode）作為干擾數據後，使得系統誤判其為長臂猿（gibbon）。

參考資料：
Ian Goodfellow and Jonathon Shlens and Christian Szegedy, Explaining and Harnessing Adversarial Examples. International Conference on Learning Representations, 2015.

科技即戰力

請同學透過電腦或智慧型手機連上Quick, Draw! 網站（https://quickdraw.withgoogle.com/），動手畫畫看。除了將自己的繪製結果分享給同學，也可以進入該網站資料庫，觀看資料庫中的塗鴉圖形與繪製過程。

※這個繪圖遊戲是藉由類神經網路的概念研發出來的，只要在時間內繪製簡單的圖形，系統將透過先前所累積的圖形經驗，猜測使用者所繪製的圖形。因此愈多人在這個網站繪製圖形，系統猜對的機率便會增加，這就是運用資料樣本提供機器進行學習並調整模型的應用。

8-2 雲端運算

雲端運算泛指的是在網路上取得的應用與服務，而這些應用與服務包含軟體和硬體等資源。有別於以前的人們如果要使用軟體，就必須先將軟體安裝到作業系統上；或是碰到容量不足的問題時，必須額外購買硬碟、增加儲存空間。現在我們只要透過瀏覽器，大多都能直接取得這些軟硬體服務了。

8-2-1 雲端運算的應用模式

網路發展在近年來大幅提升了傳輸速率和增加頻寬，因此我們可以透過網路使用更多元化的運算服務和儲存空間，如 Google 提供的 Gmail 、雲端硬碟、線上文件編輯、線上影音串流等。只要本地端電腦能上網，即可以享受許多遠端伺服器所提供的雲端服務，而不需要在本地端電腦安裝或下載許多軟體或資料，這些軟體和資料都在網路上就可以直接使用。

雲端運算可以衍生出下列三種應用模式：

軟體即服務（Software as a Service，SaaS）

軟體即服務——應用軟體通常在購買後，必須先安裝於作業系統上才能使用，現在有了雲端運算的發展，我們可以透過作業系統中的平台環境，如 Chrome 瀏覽器等，直接於網際網路上購買並使用所需的應用軟體（圖8-2-1），如微軟的 Microsoft 365、Google Apps、iCloud 等。對於軟體發行端來說，透過網際網路直接提供使用者所需的應用軟體，可以對此軟體進行使用權限的控管、更新作業、客製化服務等。

圖8-2-1 Google 提供許多雲端服務

對於使用端來說，不用事先安裝軟體，可以更彈性地選擇所需的服務，且降低內部系統或資訊人員維護上的負擔。目前許多學校的校務行政系統也朝著向上集中管理的目標邁進，透過專業的向上集中化管理更可提升資訊安全的防護，而教師們也可以將課程內容上傳到雲端學習系統平台。例如：使用 Google Classroom，讓師生都能在 COVID-19 疫情的影響下，透過雲端平台進行遠距課程。

平台即服務（Platform as a Service，PaaS）

平台即服務——PaaS 提供使用者開發軟體的平台與環境，使用者不需要在本機建置撰寫程式與資料庫管理的環境，即可迅速在網際網路上進行開發與測試，如同提供一間繪畫教室，裡面的畫紙、顏料應有盡有，畫家只需要專心繪製畫作。而透過PaaS的平台，更可以整合並加速軟體（SaaS）的開發。如 Google App Engine（圖8-2-2）、Microsoft Azure、VCloud Air 可以支援多種程式語言的開發，在 APP 開發完成後迅速發布，倘若開發的 APP 很受歡迎，同時有很多使用者連線使用，此平台也會自動將流量進行有效的分流。

圖8-2-2 Google App Engine提供了開發者實作的線上平台，支援許多熱門的程式語言，並提供許多開發工具

基礎設施即服務（Infrastructure as a Service，IaaS）

基礎設施即服務提供使用者穩定的網路連線、儲存設備、運算服務等基礎架構與設施（圖8-2-3），例如：提供伺服器的硬體環境給所需的使用者或公司，使用者也不需要擔心規劃的機器能否承擔高流量等風險，僅須進行簡單的設定，調整與部署基礎設施，如儲存空間配置等，並依使用量彈性付費。如 Amazon EC2 ，即提供許多知名網站如 Netflix、Airbnb、Expedia基礎設施的服務。

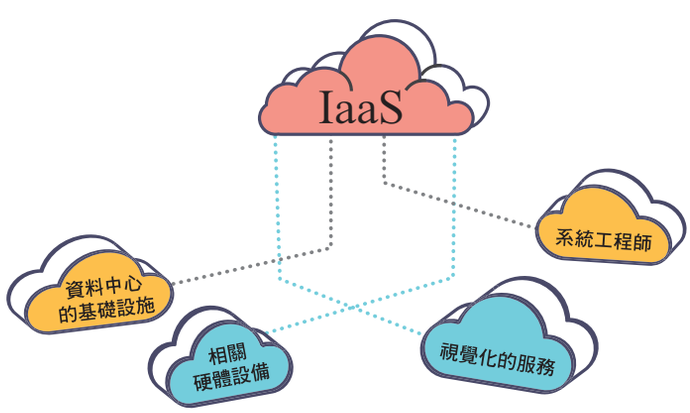


圖8-2-3 IaaS所提供的服務

雲端運算的三大應用如下：

	軟體即服務（SaaS）	平台即服務（PaaS）	基礎設施即服務（IaaS）
提供對象	一般使用者	程式設計師	程式設計師 系統管理師
提供服務	線上應用軟體	程式開發平台 程式測試平台	伺服器 網路連線
提供業者	Google Apps、iCloud、Microsoft 365	Google App Engine、Microsoft Azure	Amazon EC2、OpenStack、Rackspace

8-2-2 雲端運算的五大特性

無所不在的網路存取（Broad Network Access）

無論身在何處都可以運用手機、平板電腦、筆記型電腦和桌上型電腦等多種設備，連接網路並存取、使用所需的服務。在任何地方都可以使用雲端資源，無需維護機房，只要能連接網路就能使用雲端服務。

迅速且彈性（Rapid Elasticity）

使用者可以隨時彈性地選擇自己此時所需要的空間或頻寬流量等。對組織內部或外部進行快速且彈性的調整，以滿足資源的分配、所提供的服務內容等需求。

可計量的服務（Measured Service）

提供可以被監控與記錄的服務，讓每項應用和服務所使用的空間、流量都能被精確的計算出來。主要目的是藉由以量計費，實現用多少付多少（Pay As You Go）（圖8-2-4）。

隨選即用服務（On-demand Self-service）

使用者可依自己的需求，即時地自助選用所需服務和應用，無需透過供應商的人工交易。提供給使用者最適切的需求，客戶要什麼就給什麼，且使用者無需了解雲端中的基礎設施，不需進行底層控制設定即可開始使用其運算資源。

資源集區（Resource Pooling）

雲端供應商將應用服務資源集中管理，並在多用戶的情況將運算資源按需求動態分配。

圖8-2-4 Amazon EC2用多少，付多少的傳輸費用

8-2-3 資料中心的概念與應用實例

資料中心提供組織存放重要應用程式及資料的硬體設施，包含專業且穩定的運算與儲存的資源及環境。資料中心建置的主要設備涵蓋路由器、交換器、防火牆、儲存系統、伺服器等硬體。

而要架設一座大規模的資料中心（圖8-2-5），首要條件就是要有充足的電力資源、節能的散熱系統，以及占地廣闊的土地，中心內除了架設用來儲存資料的硬碟外，還需要大量的伺服器主機提供資料運算、儲存、分析等服務。在巨量資料時代，有更多的應用如人工智慧、物聯網等皆需要雲端運算服務，資料中心扮演著這些應用基礎設施的幕後功臣。

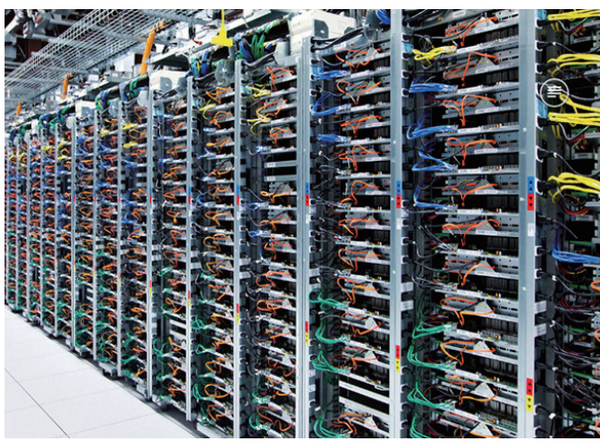


圖8-2-5 資料中心

超大型資料中心（Hyperscale Data Center）

超大型資料中心是近年來成長最為迅速的資料中心類型，我們平常所使用的雲端服務大多也是由此提供，如 Amazon、Microsoft、Google（圖8-2-6）、Meta、IBM 等企業皆有建置。除了公司自己提供一系列的雲端服務給使用者外，更可提供伺服器主機、儲存空間、網路連線的租賃方案給一般使用者或企業使用，讓一般使用者或企業不須自行建置資料中心環境、維護伺服器主機、備份儲存資料等。資料中心也可以依需求，彈性配置硬體的資源，提高伺服器與硬碟的使用率，一般企業也能節省建置與維護成本，縮短整體建置時間，讓服務快速上線。若硬體資源不敷使用時，也能迅速擴充。

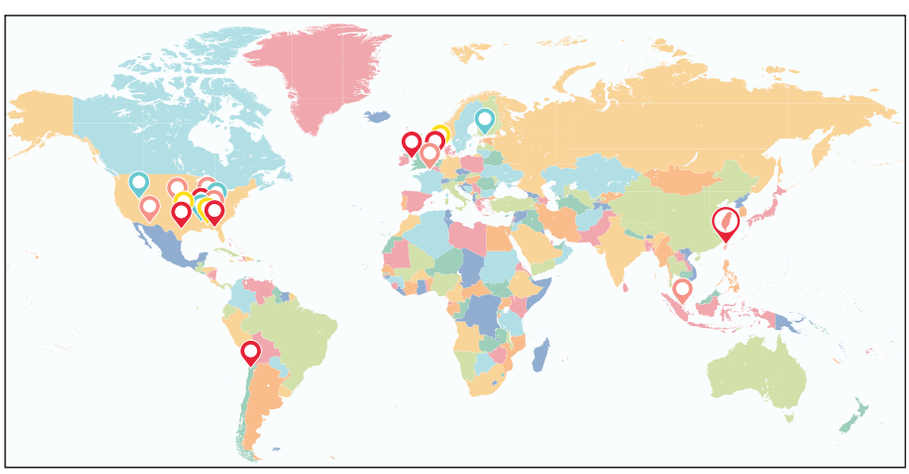


圖8-2-6 Google資料中心分布

科技即戰力

身歷其境——資料中心的內部環境

請同學連線到中華電信資料中心IDC網站（https://www.idc.hinet.net/vr/），在3D畫面中移動自己的所在位置，並同時觀察資料中心的環境。說說看，你看到了哪些設備？這些設備分別提供了資料中心哪些功能呢？（本學習活動可使用VR設備，若無VR設備亦可用滑鼠操控前進方向）

8-3 IoT 物聯網

將各種裝置、設備運用有線或無線網路連接在一起，進行不同裝置間的傳輸與功能連動，這種使物品相連的網際網路即為**物聯網**，也是讓科技與生活更密切結合的應用（圖8-3-1）。

在物聯網中，使用者可以即時透過網際網路進行物聯網裝置的監控、管理與設定，亦可結合人工智慧的演算法技術，實現**人工智慧物聯網**（Artificial Intelligence of Things，AIoT）的應用，如智能監控、智慧家庭、智慧交通等服務。

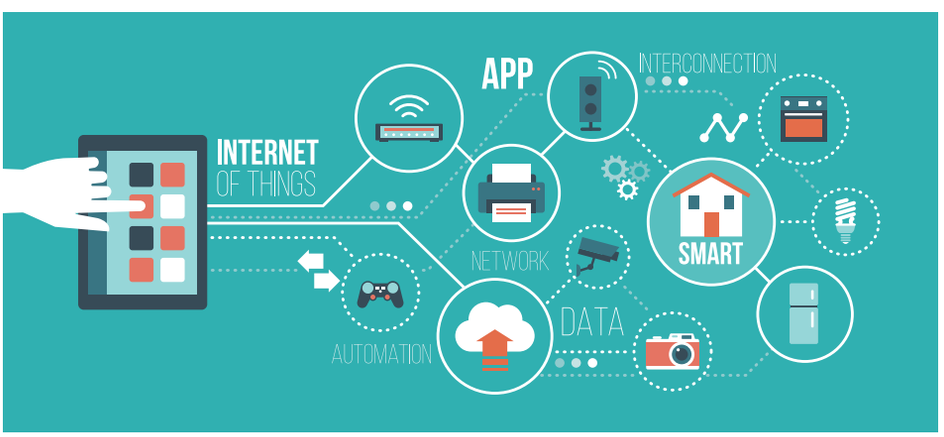
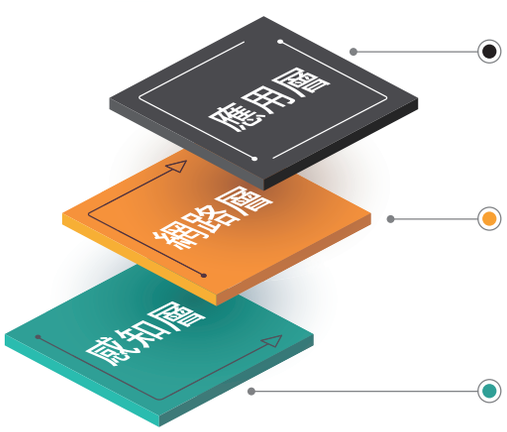


圖8-3-1 物聯網概念

8-3-1 物聯網的架構概念



雲端伺服器主機接收到網路層所傳來的大量資料，進行分析、處理，將資料轉成有意義的資訊或知識，並提供最適切的服務和訊息給使用者或其他裝置運用。

>> 智慧農業
>> 智慧工廠
>> 智慧家庭
>> 智慧醫療

當上述感測元件產生數據資料後，就必須藉由有線網路或無線網路（4G、Wi-Fi、藍牙、Bluetooth Beacon、ZigBee等）傳送到可以運算和分析的裝置上。如雲端系統進行資料處理，由感測元件（感知層）到資料處理（應用層）的裝置，這段過程即為網路層。

>> 藍牙
>>Zigbee
>>Wi-Fi
>>RFID
>>NFC

裝置或設備能夠透過許多感測元件，偵測周遭環境的變化，而偵測裝置會產生座標定位、溫度、溼度、空氣品質等數位化資料，並即時將這些資料經由網路層向外傳輸。在家電、工廠、農業、生態研究、醫療等領域都有廣泛的應用。

>> 溫溼度
>> 氣體監測
>> 動作感測
>> 衛星定位

圖8-3-2 物聯網三層架構

網路層如同人體的神經系統；而感知層如同皮膚的感覺受器，能將外在的變化傳遞到大腦，進行訊息處理；應用層則扮演大腦的角色，最後做出相對應的因應（圖8-3-2）。

從感知裝置產生數據資料，透過網路傳送至雲端伺服器主機，再經由資料分析後，自動進行相對應的反應處理，此為物聯網的三大架構概念。物聯網在各個領域發展出許多應用，可以自動處理特定事件，或運用在大量的資料分析上，如資料採礦正是應用層的重要發展。此外，將資料傳送到雲端上也延伸出一系列雲端服務的商機。

8-3-2 物聯網的應用

智慧家庭

這裡所說的「智慧」是提出並解決使用者在日常生活中的問題，或是使生活更為便利，提升生活品質的方案。每個人對於生活品質的訴求皆不同，因此使用者可以彈性選擇自己所需的智慧家庭組合，並調整裝置設定以符合自身生活習慣。例如：早上由智慧型手機的鬧鐘喚醒後，透過物聯網觸發窗簾開關、將窗簾自動拉開；而夜燈也會因為亮度感測器偵測到自然光而自動關閉；智慧音響則開始播放使用者喜愛的音樂頻道；掃地機器人也在使用者出門並將智慧門鎖上鎖後開始自行運作，同時也啟動門窗震動感測器與監視器，以防宵小進入。一連串裝置的觸發事件，都是讓生活更加便利的應用。在目前智慧家庭（圖8-3-3）應用中，業者提出了許多方案，關注的重點大多是在安全、便利、節能、娛樂這四大方面。

圖8-3-3 智慧家庭示意圖

智慧汽車

圖8-3-4 智慧汽車示意圖

新世代的車輛配備了許多感測器，例如：微波雷達感測器、攝影機影像辨識、光敏感測器、溫度感測器、胎壓感測器等裝置（圖8-3-4）。現今智慧汽車的感測系統，透過感測元件收集駕駛環境變化的資料，並經由行車電腦分析後，主動驅使車輛各模組執行動作，進而讓車輛本身具備主動調整的功能，減輕駕駛專注力的負擔，更可以達到科學化的主動安全防護機制，例如：車道維持系統、雨滴感應式雨刷、雷達感應式車距維持定速系統、自動煞車輔助系統、胎壓偵測系統、空調恆溫系統等。期許將來能夠藉由這些感測系統，普及無人駕駛車輛。

工業4.0

工業4.0一般的定義為「**工業技術數位化**」的興起，藉由多種數位化的技術，達到製程最佳化，並藉此改善產品的良率，以提升產量為目標。工業4.0的核心是**連接性**（Connectivity），在工廠的製造設備上架設感測器，管理人員可即時監控機器製造的過程，彈性調配生產排程，偵測機器運作狀態，進一步將上、中、下游所有機器透過網路相互連線，使設備自動進行資料的溝通，此即為物聯網的應用。工業4.0也將零組件與產品的生命週期進行追蹤記錄、即時分析生產數據，並搭配九種科技（巨量資料、雲端平台、自動化、系統整合、物聯網、網路安全、積層製造、擴增實境、以及模擬，圖8-3-5），確保生產端的運行能維持在最穩定的狀態。工業4.0不僅改善了過往工業上效率不足、成本過高、生產線不靈活等缺點，協助推動傳統製造業的數位轉型，更是將更快的網路速度、更強大的運算能力以及多種科技革新做全面性的整合運用，不僅能讓更多高度自動化的產線效率提高，也讓工廠系統的生命週期管理與服務成為可能，進而促使生產、服務，甚至是商業模式的創新。

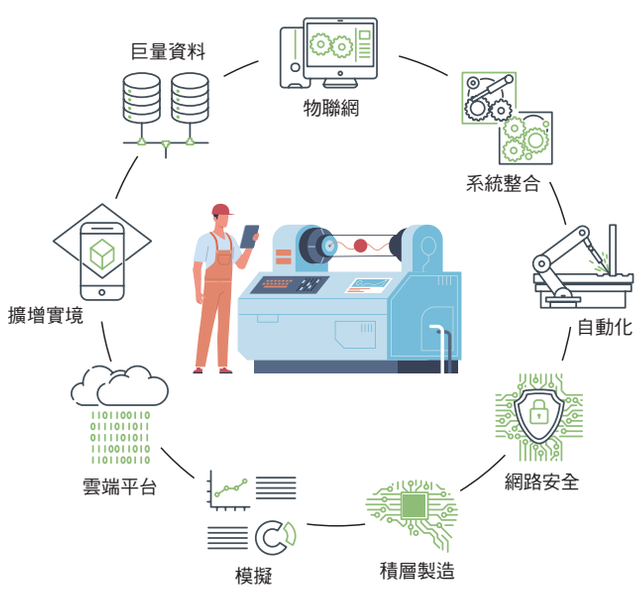


圖8-3-5 工業4.0的九大科技

醫療

許多智慧型穿戴裝置如智慧手環、智慧手錶等，都具備有睡眠監測、壓力與呼吸監測、血氧飽和度監測、心率監測等感測器記錄生理狀態，並將量測到的數據經由藍牙傳輸至智慧型手機，智慧型手機中的APP再連線至雲端伺服器主機進行分析和處理，最後將數值與結果回覆給使用者。雖然無法與醫院儀器的精準度相比，但對於現代醫療來說，這也表示預防醫學的重要性日益提升。

在醫學中心往往可以見到許多物聯網的應用，例如：將許多醫療儀器、設備無線連網化，可以讓這些儀器更機動地前往醫院各個場域，也可以讓監控感測器連上醫院網路，監測患者的生理狀態如脈搏、血氧、血壓、體溫等，除了可以擴大臨床數據的蒐集時間，降低醫護人員的負擔，更能提昇醫療品質。患者不用再像以前一樣從身上連接一堆線材或感測器到電腦，即可將訊息自動雲端化，建構出更有效率的醫療平台（圖8-3-6）。

圖8-3-6 智能病房建置床邊照護系統，提供零時差的安全醫療服務

智慧交通

在交通方面，透過物聯網整合駕駛、車輛、道路等系統，可提出許多解決方案及服務，有利於改善交通狀況，對於地狹人稠的臺灣而言更是顯得格外重要。例如：智慧停車場以感測器連結燈號的切換，可快速引導駕駛找到空位，減少尋找車位時所消耗的能源以及排出的廢氣；也能藉由將許多車輛的即時動態（如車輛的定位座標）傳輸至共同的伺服器主機，再透過演算法分析這些巨量資料與地理資訊系統，了解該路段當下車流的動態。如 Google Map 可以即時顯示所有路段的車流狀態是順暢或壅塞，並提早建議駕駛改道，提升道路的使用效能，改善交通運輸環境。

我國政府也積極地導入道路建設的智慧化基礎設施，例如：在道路設置車流量偵測器（圖8-3-7）或車牌辨識系統，即時監控道路的車流量，並透過遠端調整交通號誌疏導壅塞路段。臺北市也在近年推行智慧動態號誌（圖8-3-8），透過人工智慧影像辨識系統觀測路口當時車流情形，並主動以動態方式調整紅綠燈號誌，降低路口停等時間，使車流更加順暢。

圖8-3-7 車流量偵測器

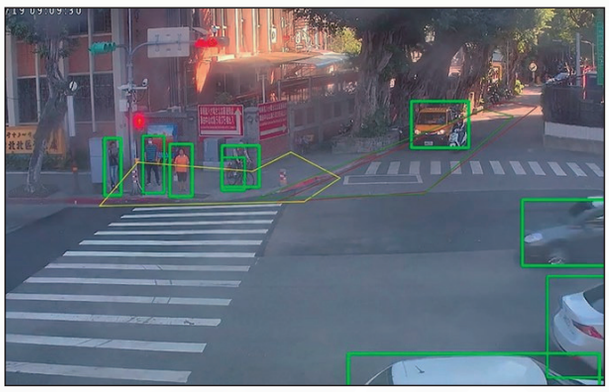


圖8-3-8 臺北市智慧動態號誌

在鐵道運輸上，也可以運用 5G 與 AIoT 結合的技術，預防諸多意外事故。例如：透過沿線的傳感器和 5G 系統的結合，縮短緊急狀況自動回報的時間，替駕駛以及行控中心爭取更多的反應時間與空間。

在時速超過 300 公里的高速鐵路上，我們會發現上網的連線品質經常受到影響，這是因為 4G 通訊系統雖然理論上能支援到時速 350 公里的運輸載具，但高鐵除了速度快以外，還有地形的屏蔽和隧道的影響，容易造成通訊過程中斷。然而，5G 最快可支持到時速 500 公里的移動載具，只要基地臺的密度夠高，就可以在高鐵上享受不間斷的網路通訊體驗（圖8-3-9）。

圖8-3-9 以臺灣高鐵為例，中華電信在全臺12個車站（包括地下化）與48個隧道段，建置了強化訊號的洩波電纜（Leaky Cable）。

農業4.0

農業非常適合運用物聯網的技術監測各項環境數據，並且自動做出因應措施，如溫溼度感測器、光照強度偵測等，若土壤溼度太低，則啟動自動灑水系統，讓智慧灌溉系統僅在特定條件下灑水運作，進而節約用水，讓土壤維持最佳溼度狀態，透過物聯網技術與人工智慧的結合，提高單位面積的生產量。

而農民也可以運用感測器，把蒐集到的資料上傳至雲端伺服器，並在遠端監控這些數據與裝置；也有農民以無人機進行化學肥料或農藥的噴灑，可避免人體直接接觸化學藥劑。透過科學化的方式結合巨量資料、物聯網、人工智慧的應用，農業也走向了以數據化精準控制作物生產過程的階段（圖8-3-10）。

圖8-3-10 高雄毛豆與智慧農業管理系統

物聯網發展下的資訊安全

當生活上的物品與服務都陸續連接至物聯網時，家電、智慧手錶、監視器等物品彼此之間都能傳遞資料、互相連動。依據不同目的而開發的智慧家居、智慧汽車、智慧交通、智慧城市等新型態的科技應用，也使我們的生活更加便利。與此同時，更多應用也正不斷地被開發出來。

但面對擁有巨量資料的政府或企業，當我們在生活上許多的隱私（如起床時間、監視器畫面等資訊）被搜集、同步上傳雲端後，也難防有心人士利用這些資料進行商業獲利或其他不法行為。因此除了開發業者在硬體層、軟體層、通訊層三個層面都必須層層把關，替使用者盡到最完善的資訊安全防護之外，使用者也應具備保護自己隱私的資安素養。

科技即戰力

請試著把班上的教室規劃成一間能運用物聯網設備運作的智能教室（例如：當老師打開投影機時，電燈就會熄滅、窗簾會自動拉上），並和同學們一起討論、分享你的想法。

課堂檢測

( ) 1. 將定義好的特徵和規則作為提供系統決策的依據，是屬於人工智慧的哪一發展階段？
(A)機器學習 (B)深度學習
(C)專家系統 (D)廣度學習。

( ) 2. 不使用標準答案提供系統解決問題，而是在人類的思維架構的模型下，輸入大量的現象到系統中，讓系統依據模型中的特徵自行判斷，是屬於人工智慧的哪一發展階段？
(A)機器學習 (B)深度學習
(C)專家系統 (D)廣度學習。

( ) 3. 能夠從巨量資料中自動萃取特徵，這種做法可以給模型更多自我調整的機制，且不需要人工介入處理，即可建立模型，是屬於人工智慧的哪一發展階段？
(A)機器學習 (B)深度學習
(C)專家系統 (D)廣度學習。

( ) 4. 下列何者非物聯網架構層級？
(A)感知層 (B)應用層
(C)網路層 (D)傳輸層。

( ) 5. 下列哪一種雲端應用提供了微軟的Office360、Google Apps、iCloud等服務？
(A)SaaS (B) IaaS (C)PaaS (D)MaaS。

( ) 6. 下列哪一種雲端應用提供了Google App Engine、Microsoft Azure等服務？
(A)SaaS (B) IaaS (C)PaaS (D)MaaS。

( ) 7. 下列哪一種雲端應用提供了穩定的網路連線、儲存設備、運算服務等基礎架構與設施，如Amazon EC2、OpenStack等服務？
(A)SaaS (B) IaaS (C)PaaS (D)MaaS。

( ) 8. 在任何地方都可以使用雲端資源，無需維護機房，只要能連接網路就能使用雲端服務，此為雲端運算的哪一項特性？
(A)無所不在的網路存取 (B)資源集區
(C)迅速且彈性 (D)可計量的服務。

( ) 9. 使用者可以隨時選擇此時所需要的空間或頻寬流量等，此為雲端運算的哪一項特性？
(A)隨選即用服務 (B)資源集區
(C)迅速且彈性 (D)可計量的服務。

( ) 10. 每個服務和應用都可以被精確計算出所使用的空間或流量，此為雲端運算的哪一項特性？
(A)隨選即用服務 (B)資源集區
(C)迅速且彈性 (D)可計量的服務。

( ) 11. 資料中心僅提供代管服務，並未提供伺服器系統的維運，企業也擁有完全屬於自己維運的伺服器主機，適合內部具有相關資訊人力配置的企業或用戶，請問是下列哪一種資料中心類型的應用？
(A)主機代管資料中心 (B)超大型資料中心
(C)企業資料中心 (D)電信資料中心。

( ) 12. 物聯網「感知層」技術中，主要可分為感測技術與辨識技術，下列何者不屬於感測技術？
(A)亮度 (B)RFID (C)壓力 (D)溫度。

( ) 13. 透過教育雲推動一系列的遠距教學可能成為未來的主流，若我們透過 Google Meeting 的系統進行遠距同步教學，請問是屬於哪一種雲端服務？
(A)SaaS (B)PaaS (C) IaaS (D)DaaS。

( ) 14. 下列何者不是「工業 4.0」的特性？
(A)柔性生產
(B)大量運用物聯網技術
(C)供應鏈上游利用大數據精密地規劃與創造下游需求
(D)以智慧整合感控系統驅動生產。

( ) 15. 臺灣將工業 4.0 奉為產業轉型的國家政策。工業 4.0 是產品開發生產銷售與服務，整個系統的生命週期管理與服務。在下列工業 4.0 的描述中，何者有誤？
(A)做法是分散式、協同式
(B)強調生態系，跟供應商、客戶也要資料共享
(C)上中下游所有機器透過網路相互連線，自動對話溝通
(D)只強調使用電子裝置及資訊技術來校除人為影響以增進工業製造的精準化、自動化。

( ) 16. 何者與雲端運算架構的關係較小？
(A) IaaS (B)PaaS (C)SaaS (D)HaaS。

( ) 17. 下列哪一個名詞最適合描述將所有物品通過無線通訊、感測技術及互聯網連接起來，實現智慧化的生活環境？
(A)大數據 (B)行動運算 (C)物聯網 (D)雲端運算。

( ) 18. 下列雲端科技所提供服務之敘述，何者錯誤？
(A) OneDrive 是一種雲端儲存服務
(B)所謂公有雲是企業或個人可通過互聯網使用第三方所提供的雲端服務
(C)軟體即服務，例如：Google App Engine
(D)物聯網把所有物品通過射頻識別等訊息感測設備與互聯網連接起來，實現智能化識別和管理。

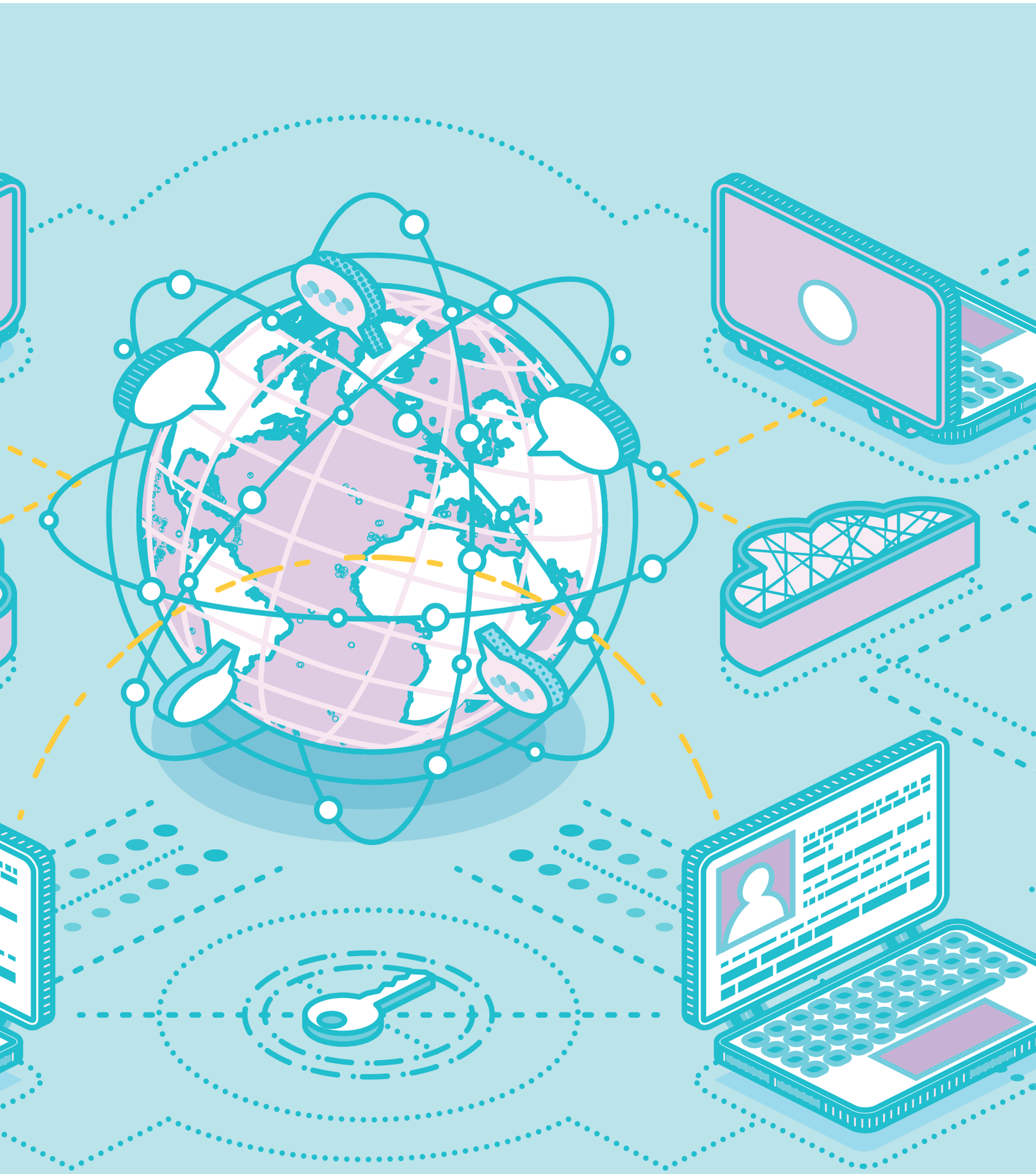
( ) 19. 下列何者為人工智慧技術結合物聯網應用的縮寫呢？
(A) IcaT (B)AIoT (C) IoTA (D) A IoC。

主題四

資料表示、處理及分析

chapter 09
巨量資料的概念與應用　224

chapter 10
資料探勘與機器學習的基本概念　242



chapter

09 巨量資料的概念與應用

隨著軟硬體與各項通訊技術的演進，我們能儲存與處理的資料量與資料型態也隨之增加。各領域對於數據的分析已不限於傳統的資料處理技術，更想透過這些大量、多元、瞬息萬變的資料，進行追蹤、觀察特定領域的潛在趨勢。例如：在瀏覽器中任何一次點擊的動作、對搜尋引擎輸入的關鍵字、通訊軟體所收發的訊息，以及在社群平台發布的文章、照片，或是導航軟體透過手機GPS接收器不斷地接收座標訊號等。

對於政府或企業來說，這些無時無刻不斷傳遞的大量資料，都可能衍生出許多有價值的應用或決策。

9-1 巨量資料的概念

9-1-1 資料處理的概念

當**資料**被蒐集時，它們可能是以文字、數字等形式所呈現的紀錄，並且尚未被賦予任何意義。如下圖9-1-1的空氣品質指標值（AQI）：當我們一開始看到「AQI：120」這筆原始資料時，暫時無法對它做出任何有意義的解讀。但是一旦我們獲得了「AQI介於101到150之間，代表空氣品質不良」這項參照表準後，就可以賦予原始資料「AQI：120代表空氣品質不良」的意義；基於我們的知識或經驗，我們已經知道「空氣品質不良，可能會誘發人體的過敏反應」，因此我們會進一步做出相對應的決策，像是「戴上口罩」或「避免外出」等行為。

因此，各類被蒐集到的原始資料都必須先經過處理，才能轉化成有意義且有價值的**資訊**；各項資訊整理之後就能形成我們的**知識**；最後運用我們的**智慧**，進行更多的應用與創新（圖9-1-1）。

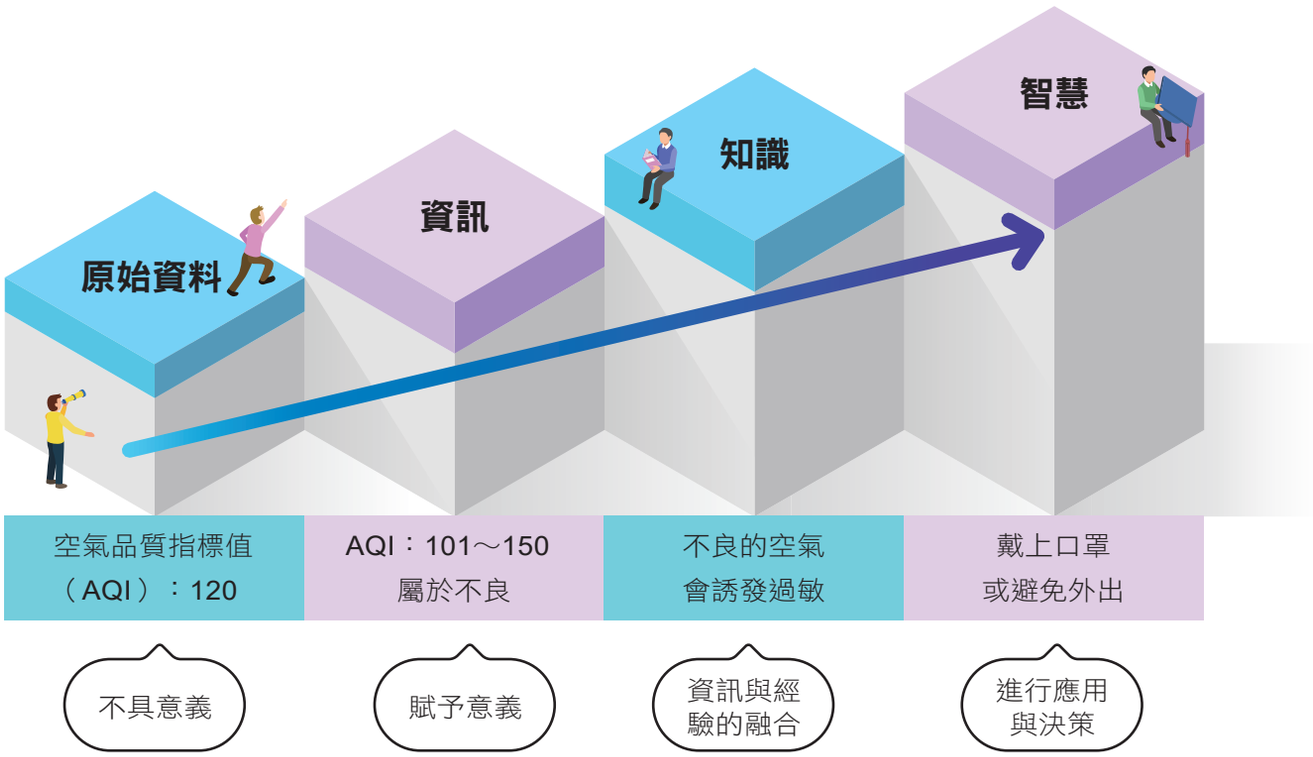


圖9-1-1 從原始資料到智慧的流程示意圖

9-1-2 什麼是巨量資料

巨量資料，含括了來自各種不同管道的大量**結構化**與**非結構化**的資料。其中，結構化的資料是指透過特定規則所儲存的資料，例如：Excel 試算表軟體、SQL 資料庫軟體等，每一個儲存格中都是經過明確定義的資料；而非結構化資料泛指的是所有不屬於結構化的資料，如文字、聲音、圖片、影像等，這些資料的類型和格式相當多元，但是也增加了分析、儲存和搜尋資料的難度。

巨量資料的應用——Google 的搜尋引擎

Google的搜尋引擎是進入網路世界的知名應用之一，當居住在全世界各地區的人將關鍵字輸入到Google的搜尋引擎伺服器系統中、進行搜尋時，其實Google也利用全球使用者不斷輸入而成的巨量資料，了解哪些正是熱門的關鍵詞，哪個網站連結最多人點擊等。如下圖的Google Trends會列出最近的熱門搜尋關鍵字（圖9-1-2）。

圖9-1-2 透過 Google Trends 可以了解目前世界各地的熱門搜尋與累計數量

巨量資料的應用——Google 即時路況

Google Map 的路線規劃功能可以即時計算該路線所需的時間，與目前各路段的行車順暢度，透過不同顏色讓用路人了解最即時的路況；而這些經過各路段的用路人也同時回饋自身的 GPS 座標給 Google Map，使伺服器能透過這些即時產生的巨量資料與演算法，計算出最新的路況資訊（圖9-1-3）。

圖9-1-3 Google Map 可以用顏色區分其規劃路線的即時路況

這些應用程式皆是運用巨量資料進行分析，產生最適切的結果，而所需的資料都是使用者在日常生活中不斷產生的資料。這些資料種類繁多，而且數量非常龐大，從一開始的蒐集到後續的分析，都必須透過有效能的軟硬體搭配，才能產出我們需要的資訊。例如：系統前端結合物聯網感測器蒐集資料後，傳送至系統後端應用層伺服器進行運算分析並產出資訊，再搭配人工智慧等技術進行資訊分析，協助人類進行決策判斷。實際運用則像是 Google Map 透過里程、路況等因素，分析並產出最短時間的最佳路線後，我們也可以再透過自身的經驗，如道路是否正在施工，整體安全性等考量，選擇另一條較適合自己的路線。

巨量資料的特性

巨量資料除了廣義地指資料量龐大外，還擁有下列四種特性，我們將之稱為巨量資料的4V（圖9-1-4）。

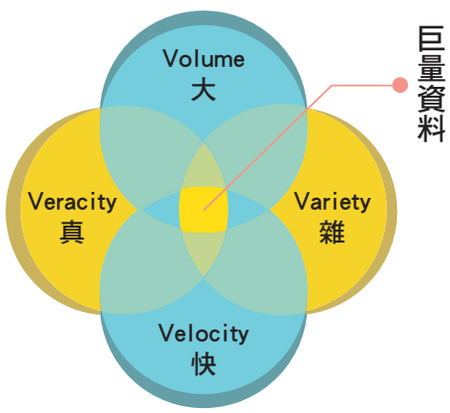


圖9-1-4 巨量資料的4V特性

1 大量性（Volume）

資料龐「大」。對於如此巨大的資料量，必須使用有別於以往一般資料庫的方式，進行資料的儲存、處理和分析。

2 多樣性（Variety）

種類繁「雜」。巨量資料的資料種類繁多，從以往的文字、數據、圖片等形式，直到現今產生的資料型態，包含網頁、聲音、動態影像、溫溼度等各種型態。由於樣式多元且複雜，且多數屬於非結構化資料，因此巨量資料的儲存和處理也與傳統數據的技術不同。

3 快速性（Velocity）

變化飛「快」。巨量資料與傳統數據最大的不同點在於資料是每分每秒不斷新增的，以用戶突破20億的Facebook社群為例，如果每位用戶每天按一個讚，就會有20億筆資料。每個人隨時隨地都在創造新的數據，這些數據生成的速度已遠超乎我們的想像。

有部分學者解釋Velocity為時效性，指的是若這些資料沒有在一定的時間內完成處理和分析，當新的資料持續產生，則這些舊有資料處理後的結果將會是無意義的資訊。就像颱風已經掠過臺灣，再藉由分析的結果發布颱風警報將成為不具任何意義的預警行為。

4 真實性（Veracity）

「真」偽存疑。在3V成為巨量資料的主要定義後，隨著儲存資料的成本與取得成本下降，巨量資料發展出第四個特性——真實性。除了資料量外，也必須確認資料的真實性，過濾造假的數據與錯誤數值後，分析出來的結果將更為準確。例如：某個體溫感測器故障，測出人體體溫高達攝氏88度，我們就必須將這筆數值過濾掉，避免影響資料整體的分析和最終結果。

以空氣盒子網站來舉例，若有某筆數據相較於周遭均值呈現異常，如圖9-1-5中，被標為3332的褐色點測站，這時，我們就該合理懷疑感測器是否故障或是周遭環境正遭到強烈的汙染。

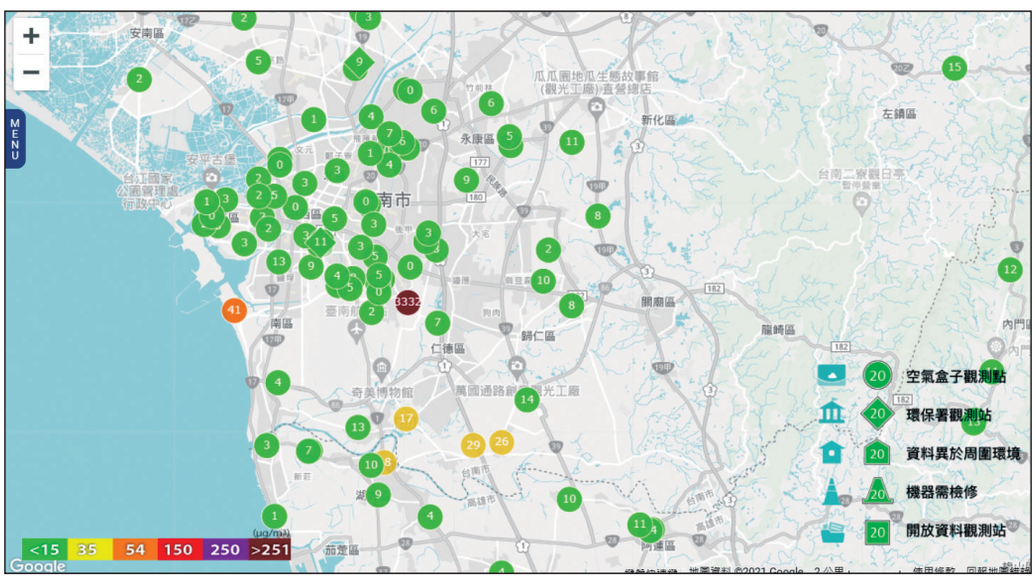


圖9-1-5 空氣盒子網頁數值異常處

目前也有學者提出第5個V——**易變性**（Variability），指的是這些資料內容會不斷改變。例如：Siri在執行情緒或語意分析時，由於使用者不同的語氣，對於相同的問句和詞彙採用完全不同的解釋。為了進行適切的語意分析，演算法必須綜合使用者的語氣、上下文及詞義進行分析。

科技即戰力

1. 請同學到Google Trend探索看看：目前最新的熱門關鍵字前五名分別是什麼？
2. Google Map的「規劃路線」功能可以告訴使用者目前的路況是「壅塞」或「順暢」，請問這項功能是透過什麼機制或演算法來達成的呢？假如某一路段的路邊有很多汽車駕駛臨時停車，會導致該路段在Google Map上顯示為「壅塞」嗎？

9-2 巨量資料的蒐集與取得

目前巨量資料的蒐集，常見的方式可透過網路爬蟲（Web Crawler）、物聯網技術等。而巨量資料的取得，較常見的方式**可透過開放資料（Open Data）平台取用已被蒐集的巨量資料**。以下將分別對網路爬蟲、物聯網技術、開放資料進行說明。

網路爬蟲（Web Crawler）

一種協助我們進行網路資料蒐集的應用，藉由網路爬蟲的程式，可以大量、自動、反覆且即時性地，將目標資料從指定網站中擷取後，再加入自己的資料庫中。常見的應用如蒐集網路中特定商品的售價，許多比價的網站即是運用網路爬蟲的技術，蒐集網路上各大購物平台的商品資料與價格，並將其彙整後呈現給消費者，節省消費者在不同購物平台比價的時間（圖9-2-1）。

圖9-2-1 使用比價網站搜尋商品

運用物聯網技術蒐集資料

我們可以運用物聯網中的各類型感測器，有目的性地蒐集目標資料。經由架設大量的感測器以蒐集即時的資料，再將這些資料進行後續的分析和應用，例如：溫溼度、太陽能發電量、雨量、地震震度感測、道路車流量、河川逕流量、監視器畫面等。

開放資料（Open Data）

以開放共享為主要理念，開放給社會大眾自由使用的官方認證資料。

但是須注意，雖然有些資料可免費取用，並且開放學術或是非營利用途，但另外有些資料卻是屬於需要付費或須徵得資料擁有者的同意、授權後方可使用，在取用資料前仍須確認資料的相關取用資訊，避免因而觸法。如Meta、Google等公司所蒐集的相關瀏覽數據可以透過付費方式取得。

開放資料是屬於巨量資料中的一部分，而政府自己也有公開的資料開放平台，當中有部分若涉及個人資料，通常必須經由**去識別化**（例如：去除可辨識個人資料的部分），才可以開放供大眾使用。

我國的「政府資料開放平臺」選定、整理有價值的資料，並開放這些關於民生、施政、研究、統計等資料給社會大眾，這些資料除了不受著作權的限制外，也建立了資料更新與格式制定的機制，方便使用者取用這些官方所統計的資料。這些具有結構化且多元格式的資料，可以讓政府或民間的系統開發人員更有效率地運用。例如：我們以關鍵字「大學人數」搜尋各大專院校目前的在學人數（圖9-2-2），並了解該校是否受到少子化衝擊，而面臨招生人數不足的危機。

除了「政府資料開放平臺」外，也可以透過一般的搜尋引擎下達關鍵字，搜尋到其他政府機關或是民間機構所統計的數據，例如：各地方政府的統計資料、世界衛生組織、歐盟（圖9-2-3）、世界經濟貿易合作組織等資料開放平台。

圖9-2-2 透過政府資料開放平臺搜尋大專院校人數統計資料

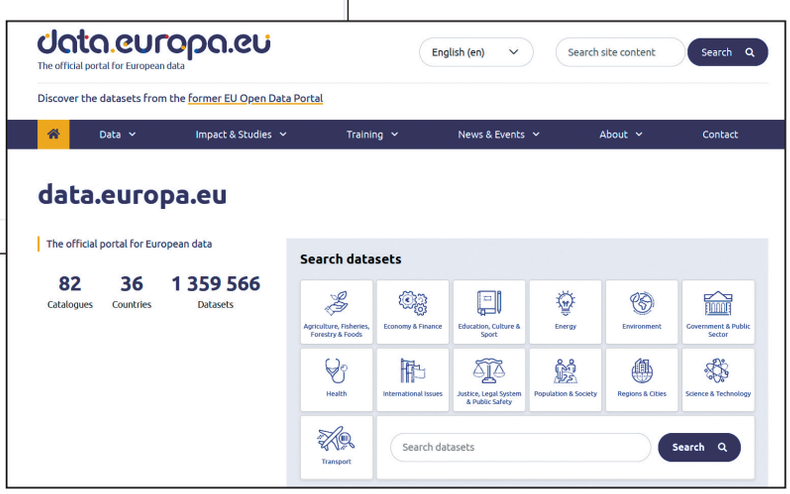


圖9-2-3 歐盟的數據資料中心

IT 補給站

巨量資料所涉及的隱私權問題

許多研究單位或是企業在蒐集巨量資料時，可能會採集到涉及侵害隱私的資訊，嚴重時會引發用戶的個資外洩，進而觸犯法律。所以，如何建立完善的巨量資料應用及管理、保護機制以防止隱私洩漏，是企業以及研究單位需要注意的一大重點。

目前，隨著人們對於資訊安全問題的重視，許多國家已經制定了相關法規，保護被蒐集者的隱私，例如：Google、Instagram等公司有制定相關的「隱私權政策」（圖9-2-4），可以讓用戶清楚了解到自身的資料會被企業及研究者如何運用，自己決定是否同意蒐集資料行為。

在2014年5月13日歐盟法院就「**被遺忘權**」（Case of Right to be Forgotten）一案作出裁定，判決Google應根據使用者請求刪除不完整的、無關緊要的、不相關的資料以保證資料不出現在搜尋結果中。這個判決的裁定，也正式宣告了巨量資料的蒐集與使用也必須合乎個人的隱私規範。

圖9-2-4 Google隱私權政策

在2018年5月25日正式執行的歐盟GDPR（General Data Protection Regulation，一般資料保護法規），這項規範以「被遺忘權」為基礎，主要目標為取回個人對於個人資料的控制，根據GDPR官網描述，這條法規是「保護以及加強歐盟成員國人民的資料隱私，以及重塑整個地區內的組織處理資料隱私的方法。」雖然法規在於保護歐洲經濟區的使用者，但因為網路無遠弗屆的特性，讓資料本身不受地域性的限制，因此全球的網路服務公司皆需要對其巨量資料搜集政策做出更新，並且在向使用者請求資料使用權時，須提供易於理解且明確的說明（圖9-2-5），同時也要讓使用者易於撤回同意。

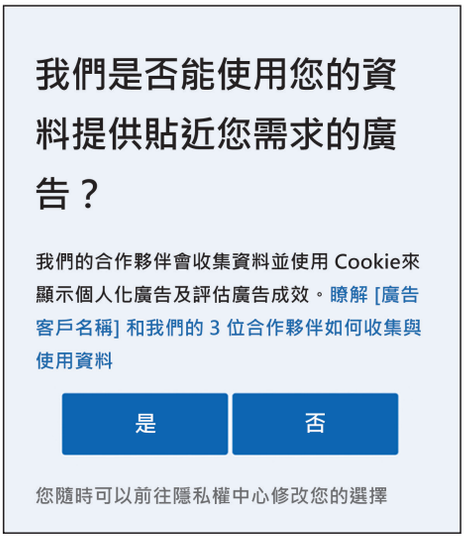


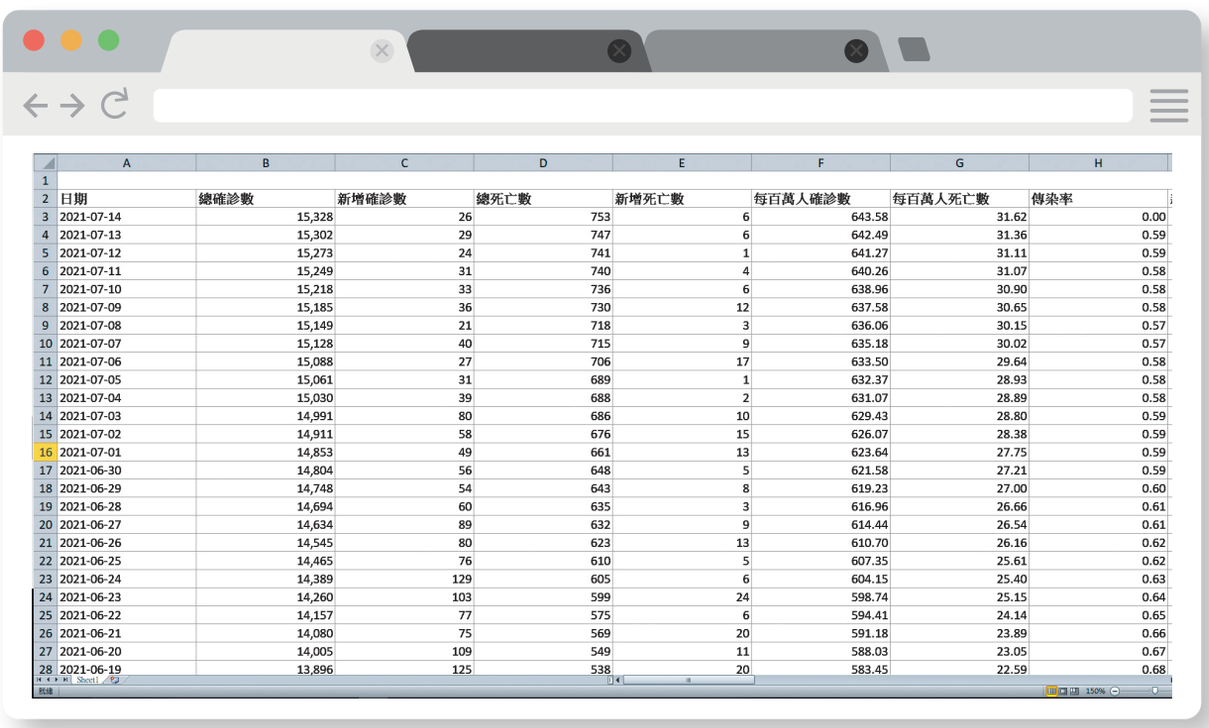
圖9-2-5 網站詢問是否同意授權搜集資料的畫面

科技即戰力

1.請同學試著到政府資料開放平臺（https://data.gov.tw/）搜尋「大學、技專校院各系所註冊率」，查閱近三年的資料。
2.請問你在政府資料開放平臺上最感興趣的三筆資料分別是什麼？

9-3 巨量資料的處理與分析

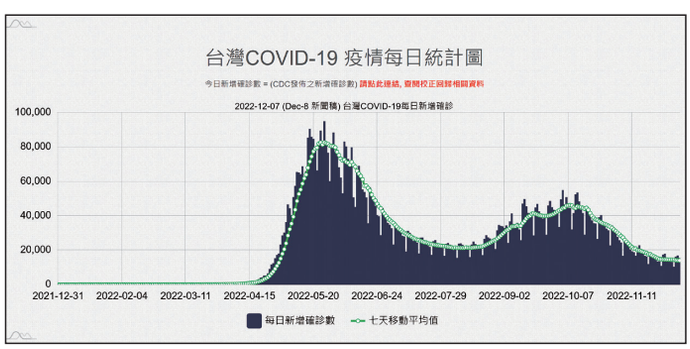
這些所蒐集到的巨量資料，究竟是如何透過電腦進行下一步的處理與分析呢？我們以衛福部疾管署統計的COVID-19資料作為例子，將每一天的確診與死亡案例資料運用試算表進行記錄（圖9-3-1），接下來可以透過後端資料的處理過程，算出總確診數、總死亡數、每百萬人確診數，及目前的傳染率等資訊，進而得出疫情的趨勢分析。



	A	B	C	D	E	F	G	H
1								
2	日期	總確診數	新增確診數	總死亡數	新增死亡數	每百萬人確診數	每百萬人死亡數	傳染率
3	2021-07-14	15,328	26	753	6	643.58	31.62	0.00
4	2021-07-13	15,302	29	747	6	642.49	31.36	0.59
5	2021-07-12	15,273	24	741	1	641.27	31.11	0.59
6	2021-07-11	15,249	31	740	4	640.26	31.07	0.58
7	2021-07-10	15,218	33	736	6	638.96	30.90	0.58
8	2021-07-09	15,185	36	730	12	637.58	30.65	0.58
9	2021-07-08	15,149	21	718	3	636.06	30.15	0.57
10	2021-07-07	15,128	40	715	9	635.18	30.02	0.57
11	2021-07-06	15,088	27	706	17	633.50	29.64	0.58
12	2021-07-05	15,061	31	689	1	632.37	28.93	0.58
13	2021-07-04	15,030	39	688	2	631.07	28.89	0.58
14	2021-07-03	14,991	80	686	10	629.43	28.80	0.59
15	2021-07-02	14,911	58	676	15	626.07	28.38	0.59
16	2021-07-01	14,853	49	661	13	623.64	27.75	0.59
17	2021-06-30	14,804	56	648	5	621.58	27.21	0.59
18	2021-06-29	14,748	54	643	8	619.23	27.00	0.60
19	2021-06-28	14,694	60	635	3	616.96	26.66	0.61
20	2021-06-27	14,634	89	632	9	614.44	26.54	0.61
21	2021-06-26	14,545	80	623	13	610.70	26.16	0.62
22	2021-06-25	14,465	76	610	5	607.35	25.61	0.62
23	2021-06-24	14,389	129	605	6	604.15	25.40	0.63
24	2021-06-23	14,260	103	599	24	598.74	25.15	0.64
25	2021-06-22	14,157	77	575	6	594.41	24.14	0.65
26	2021-06-21	14,080	75	569	20	591.18	23.89	0.66
27	2021-06-20	14,005	109	549	11	588.03	23.05	0.67
28	2021-06-19	13,896	125	538	20	583.45	22.59	0.68

圖9-3-1 利用試算表記錄資料

我們也可以將這些表格資料以**視覺化圖形**的方式呈現（圖9-3-2），運用視覺化圖形的方式進行大量資料的分析將會使我們更快了解這些數據所呈現的意義，也可供決策者制定相關政策與措施。



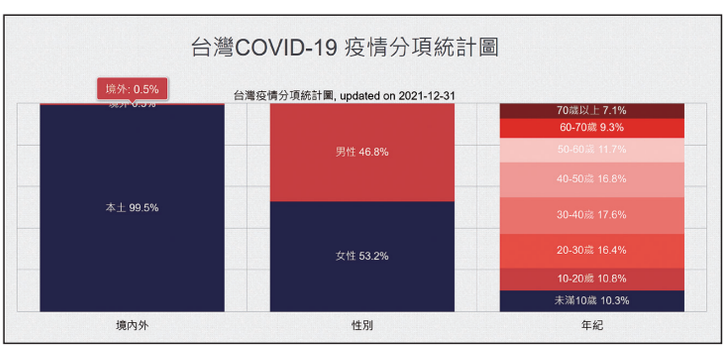


圖9-3-2 利用視覺化圖表呈現資料

其中，巨量資料的分析方式與傳統資料又有何差異呢？讓我們看看表9-3-1來比較兩者的差異。

表 9-3-1 傳統資料分析與巨量資料分析之差異比較

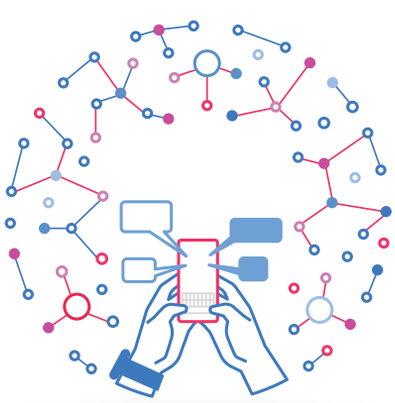
特性 \ 分析方式	傳統資料分析	巨量資料分析
資料範圍	能蒐集的資料有較多的侷限性，如傳統結帳機臺的交易資料，僅有商品、金額、購買時間等。	蒐集所有能分析的資料，且資料來源多元化。除了有目的性地蒐集資料外，還包含許多間接或自動產生的資料，如透過商店監視器畫面，統計當日不同走道所經過的人數。
資料型態	有明確定義格式的結構化資料，如在資料庫中的文字、數字都有具體定義和規範。	含括結構化與非結構化的資料，如數字、影像、聲音等，不一定為具有明確定義的格式資料。
資料分析	藉由各種統計工具，將資料分析後，驗證假設的成立與否，並提供相關資訊以進行決策。	透過機器學習與資料探勘的技術進行分析，可在未知的規則中，尋找有意義的資訊。
資料應用的範例	企業透過資料庫檢視商品銷售數量，並進行統計，決定增加或減少哪種商品進貨的數量。	企業透過數據分析找出潛在的獲利模式。如下一章將介紹的尿布與啤酒的故事。
資料的量	常見以MB、GB為儲存單位。一份文件檔案（如.ods、.xlsx、.odt等）通常以MB為儲存單位。1024KB＝1MB，1024MB＝1GB。	常見以TB、PB為儲存單位。一顆硬碟（HDD）的容量通常以TB來計算，而1024GB＝1TB，1024TB＝1PB。

科技即戰力

1.請同學到「用數據看台灣」網站（https://www.taiwanstat.com/）選擇三項資料（如即時用電、水情等），並和同學討論、分析這些視覺化資料所代表的意義。

我選擇的資料	我的分析

2.請到「SOCIAL BLADE」網站（https://socialblade.com/）尋找Facebook、Instagram追蹤人數前五名的帳號。



9-4 巨量資料的應用

巨量資料的應用可以視同將蒐集到的「巨量資料」透過「分析演算法」產出有價值的「智慧應用」。

這些應用含括許多產業，例如：金融業分析了解客戶的需求，找出潛在客戶，創造更多企業獲利；在醫療業可以用於預防疾病的發生或協助醫師進行診斷；交通運輸業者可以藉此分析各項資源和路線的規劃；科學研究者則可以用其協助預防天然災害等。

而目前巨量資料較為常見的應用有下列四種模式：

圖9-4-1 巨量資料的應用範圍非常廣泛

分析社群媒體資料，掌握社會的喜好與輿論的趨勢

透過Facebook、Instagram分析按讚、愛心、推文、分享、帳號的追蹤數量等，客觀且全面地主動了解社群的喜好及輿論趨勢，這在各個領域都有實際應用的案例。例如：粉絲團的經營者可以了解其群眾支持度和網路聲望的消長，也可以掌握目前熱門的話題或對某些事件的輿情分析等（圖9-4-2）。

2021 FB 粉專排行　勾選粉專　依平均Like排序　時間

粉專名稱	類別	粉絲人數	發文數	最高Like	平均Like	平均Like5	最多留言	平均留言	最多分享	平均分享
國立清華大學Nat...	大專院校	28787	88	3198	700	2193	394	33	1074	51
國立台東大學生命科學系	大專院校	461	8	4465	580	926	27	5	92	13
海軍軍官學校	大專院校	23279	29	2963	543	1220	68	7	124	12
The Ball這...	大專院校	52176	104	24252	504	9478	1538	47	4390	153
The Educa...	大專院校	33982	57	15307	498	5591	35	1	17	2
東海大學 Tung...	大專院校	73616	132	29074	491	7372	209	8	979	16
空軍軍官學校	大專院校	18791	29	805	388	668	33	4	13	3
國立交通大學 Na...	大專院校	13876	12	1199	382	690	34	14	316	40
國立臺灣大學 Na...	大專院校	79867	158	2413	381	2110	195	7	954	30
台大迷因 NTU ...	大專院校	4358	41	1348	352	962	159	16	587	57
馬偕醫學院招生專區	大專院校	863	15	1900	335	989	47	5	121	18
康寧大學	大專院校	2013	16	1713	335	937	8	3	11	4

圖9-4-2 2021 Facebook大專院校粉絲專頁排行

記錄並分析用戶與網站的瀏覽紀錄，以進行個人化的精準行銷

透過「哪些商品最多人點擊」、「最多人購買」、「評價最高」、「商品頁面的停留時間」等巨量資料的蒐集，進而推薦顧客合適的產品，達成個人化銷售，成功提升銷售業績（圖9-4-3）。

圖9-4-3 購物網站以瀏覽行為進行個人化銷售

運用可攜式裝置定位資料（GPS），提供適地性服務（Location Base Service，LBS）

APP結合手機定位資料，可即時依據其所在地點，直接推薦合適的服務或商品給使用者。也可以記錄使用者個體行動的路線及駐足的地點與時間，供當地政府或相關業者分析並規劃觀光景點、大眾運輸、商業區等應用（圖9-4-4）。

分析物聯網資料，調整設備的運作

透過物聯網感測器的資料回饋，隨時調整設備的運作狀態。如藉由路口車輛之感測裝置，動態調整紅綠燈的時間；透過亮度感測器回傳亮度資料，讓系統自動調整太陽能板接收日照的角度，以增加發電率（圖9-4-5）等。

圖9-4-5 太陽能追日系統

圖9-4-4 LBS應用的示意圖

科技即戰力

請以巨量資料4V的特性，判斷下列選項哪些屬於巨量資料？

A.線上遊戲產生的大量用戶遊戲操作行為
B.公司本月份的財務報表
C.全班同學的個人基本資料
D.智慧交通的 IoT 裝置產生的大量資料
E.銀行中大量的客戶信用卡交易資訊
F.50位受訪者的問卷調查資料
G.社群媒體網站產生的大量用戶互動、訊息、照片和影片等

課堂檢測

( ) 1. 下列哪一個選項較符合巨量資料中「多樣」的特性？
(A)某校的線上合作社平台的商品種類繁多
(B)皆來自於非結構化的資料，如影像、聲音等
(C)蒐集了各類型的資料，如文字、聲音、影像、溫溼度等
(D)資料庫所儲存的資料，如電話號碼、生日、地址等。

( ) 2. 雲潔想在升高二選擇類組前，回顧自己前幾次的段考成績。請問這些段考成績較符合下列哪一項性質？
(A)資訊 (B)資料 (C)智慧 (D)知識。

( ) 3. 下列敘述何者較符合巨量資料「真實性」的特性？
(A)校園中各教室不斷產生的電力監控數據
(B)校園中持續累計的缺曠課紀錄
(C)校門口的車牌辨識系統不斷產生的影像資料
(D)分析全臺灣校園電力監控系統的數據，並排除停電時間。

( ) 4. 下列哪一個選項較符合巨量資料中「快速」的特性？
(A)光纖傳輸資料的速度 (B)老師登錄某一班級的成績
(C)CPU運作的速度 (D)Instagram每小時上傳的照片量。

( ) 5. 下列何者為巨量資料的特徵？
(A)巨大資料的資料集
(B)資料大於1TB的資料集
(C)資料超出傳統資料庫的抓取、儲存、管理和分析能力的資料集
(D)資料超出人類的抓取、儲存、管理和分析能力的資料集。

( ) 6. 下列哪一個選項較符合巨量資料中「多樣性」的特性？
(A)校園中各教室不斷產生的電力監控數據
(B)監控全臺灣高中生在校使用平板電腦學習的時間
(C)量測全校體溫後，去除攝氏55度的資料
(D)校門口的車牌辨識系統不斷產生的影像資料和進出時間資料。

( ) 7. 下列何者不屬於巨量資料的應用？
(A)透過路口車輛監測數據即時調整號誌時間
(B)推薦購物節受點擊次數最多的商品
(C)透過粉絲團追蹤人數推測候選人的當選機率
(D)分析自己兩次段考成績是否進步。

( ) 8. 小華的舅舅說他在某金控銀行擔任巨量資料分析師，請問以下何者最可能符合其工作內容的描述？
(A)分析交易數據，整理數據分析報告
(B)於銀行櫃檯辦理開戶、儲蓄等業務
(C)負責人力招募及考核內容
(D)管理粉絲團並提供消費者服務諮詢。

( ) 9. 資料視覺化較不需要考量下列哪個要點？
(A)資料的正確性 (B)讀者的閱讀動機
(C)傳遞資訊的效率 (D)資料的獲利性。

( ) 10. 下列何者可能為巨量資料所帶來的問題？
(A)造成全球暖化加劇 (B)造成知識獨大
(C)蒐集資料造成侵權而觸犯法律 (D)侵害工作權。

( ) 11. 網路爬蟲可以應用在生活中的哪些部分？
(A)比價物品 (B)搜尋引擎 (C)股票數據 (D)以上皆是。

( ) 12. 下列何者不屬於資料視覺化的應用？
(A)長條圖 (B)圓餅圖 (C)折線圖 (D)透視圖。

( ) 13. 網路爬蟲不具有以下何種特性？
(A)大量的 (B)迅速的 (C)延遲的 (D)重複的。

( ) 14. 開放資料不具有以下何種特性？
(A)獨占的 (B)共享的 (C)開放的 (D)自由的。

( ) 15. 在政府開放平台中，下列何種類型的資料無法查詢？
(A)婚姻 (B)車票 (C)教育 (D)職業。

( ) 16. 太陽能追日系統能依據即時的發電量調整太陽能板受日照的角度，請問這是屬於巨量資料的何種特性？
(A)多樣 (B)大量 (C)快速 (D)真實。

( ) 17. 2014年，歐盟法院就「被遺忘權」一案作出裁定，也宣告了巨量資料的蒐集與使用，必須合乎下列哪一項個人的權利？
(A)隱私權 (B)專利權
(C)著作權 (D)自由平等權。

( ) 18. 下列哪一個資料量較可能屬於巨量資料的應用？
(A)某校每日的用電量統計
(B)某班級每日的出缺席報表
(C)Google Map的即時路況
(D)超商每日商品的銷售數量。

( ) 19. 小方在生涯探索課程中搜尋未來適合自己的工作，對於資訊方面有興趣的他在求職網站搜尋時發現，某公司徵求程式設計師，需要能開發「網路爬蟲」程式。請問這是屬於什麼樣的程式？
(A)架構資訊傳輸的路徑 (B)自動蒐集大量的網路資料
(C)自動更改程式錯誤 (D)一種惡意程式，能到處散播病毒侵入電腦。

( ) 20. 若想要分析全臺灣用電的數據，評估未來的用電量，但卻沒排除停電的時段，則可能違反了巨量資料的哪一特性？
(A)快速 (B)大量 (C)真實 (D)混雜。

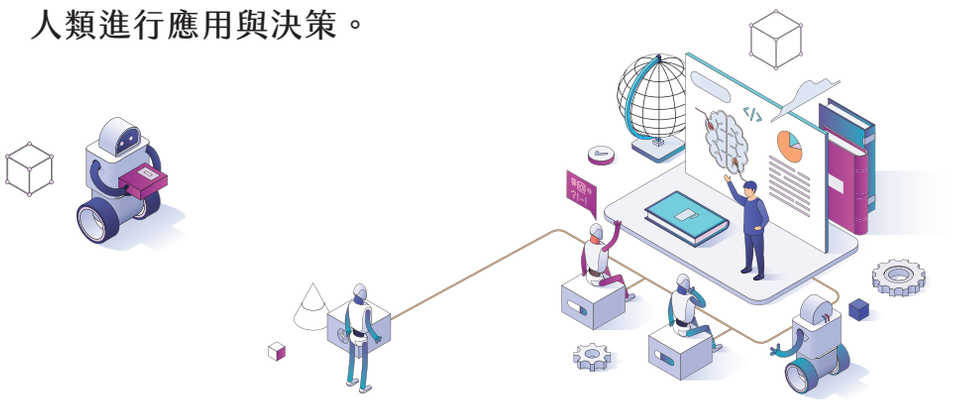
chapter 10

資料探勘與機器學習的基本概念

在上個單元中，我們了解到了巨量資料的蒐集、處理和應用，接下來在這個章節，我們將介紹巨量資料的分析技術——**資料探勘**（Data Mining）與**機器學習**的基本概念與應用。

在先前系統平台章節也曾介紹過，機器學習是屬於人工智慧演進中的一種分析技術，透過建立機器學習的模型做出預測或分類。資料探勘定義為「從資料中提取出隱含的過去未知的有價值的潛在資訊」，透過結構化或非結構化的巨量資料，結合機器學習的模型作為資料探勘的工具，挖掘有價值的潛在資訊，例如：資料的分組（透過聚類分析）、資料的異常紀錄（透過異常檢測）和資料之間的關係等（透過關聯分析）。

資料探勘和機器學習最大的不同在於「目的」：機器學習的目的是讓機器能夠辨識各種情境，並建立準確的模型工具；而資料探勘著重的則是運用機器學習的模型，將分析的結果供人類進行應用與決策。



10-1 資料探勘

資料探勘是指在巨量資料中挖掘出有意義的資訊，是一種巨量資料的分析技術，由於主要是從資料集內萃取資訊，故又稱為「**資料採礦**」。資料探勘同時也是一門跨領域的科學，綜合了人工智慧、資料庫管理、管理科學、機器學習、視覺化資訊、統計學等領域的知識所形成的應用（圖10-1-1）。

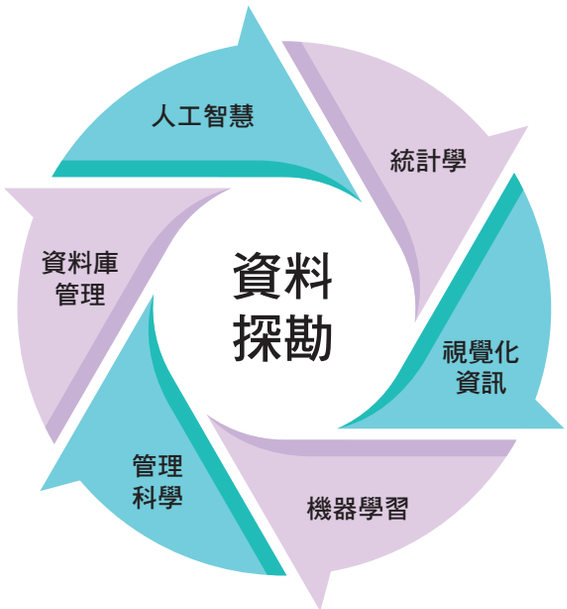


圖10-1-1 資料探勘是門跨領域的科學

關聯分析（Apriori Algorithm）是資料探勘的分析方法之一，有個知名的案例是「買尿布就會買啤酒」的故事：

美國沃爾瑪超市發現「每周五晚上只要購買尿布的顧客，連帶著也會購買啤酒」的消費行為，於是將這兩樣商品擺放在一起，結果增加了公司的營業額。透過資料探勘裡的關聯分析，在資料庫中搜尋繁雜的消費者購買資料，就能使得原本一群數量龐大且尚未發現任何意義的交易資料，成為對企業有價值的商業資訊。

不僅如此，尚有許多資料探勘技術應用於資訊科學的例子，如下圖 10-1-2是信用卡公司以決策樹進行異常交易判定的模擬。藍色橢圓形為決策因子，以分支（邊）作為選擇，綠色和紅色矩形分別為正常交易與停止交易的決策結果的分類（樹葉節點）。當客戶採用行動支付或是刷信用卡時，系統會透過決策樹的分析進行異常檢測，即時判斷這筆刷卡行為是否屬於異常交易，倘若判定結果為「停止交易」，則該筆交易可能會被信用卡公司立即中止。

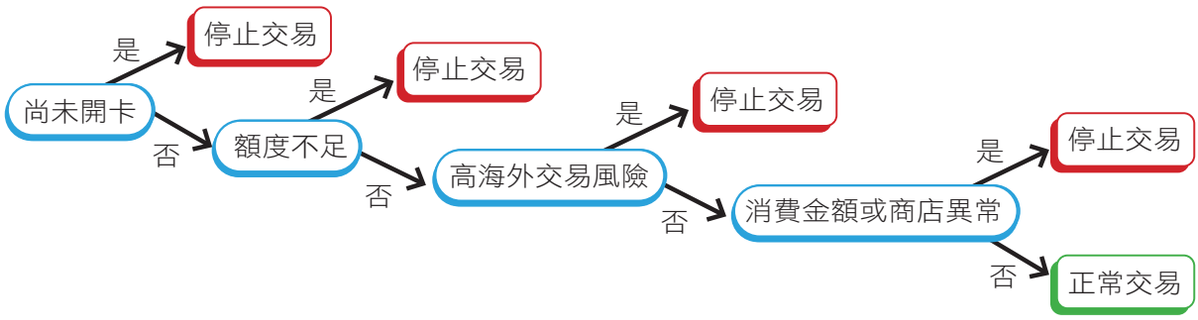


圖10-1-2 信用卡公司運用資料探勘技術進行每筆行動支付的異常交易判定

此外，也可探討不同族群的使用者與行動支付行為、網路購物行為以及服務推薦之關聯。透過使用者、行動支付廠商與品牌商三方面的交互關係，了解現今使用行動支付的使用者是否有特定族群的偏好或潛在需求（透過**聚類分析**），從中找尋是否有可行的行動支付行銷模式，以提供行動支付廠商與品牌商作為行銷策略上的參考。

如上述異常交易的判定、分析行動支付行為、推論可行的行銷模式，這些資料探勘的相關例子，都是為了找出可進行分析、判斷問題的模型，透過挖掘大量的資料，進而獲得有價值的資訊，未來若是預測相關、類似的事件時，即可做為參考。

以下將介紹資料探勘的流程及分析的方法，提升對於資料探勘技術的基礎概念。

10-1-1 資料探勘的流程——以運用關聯分析為例

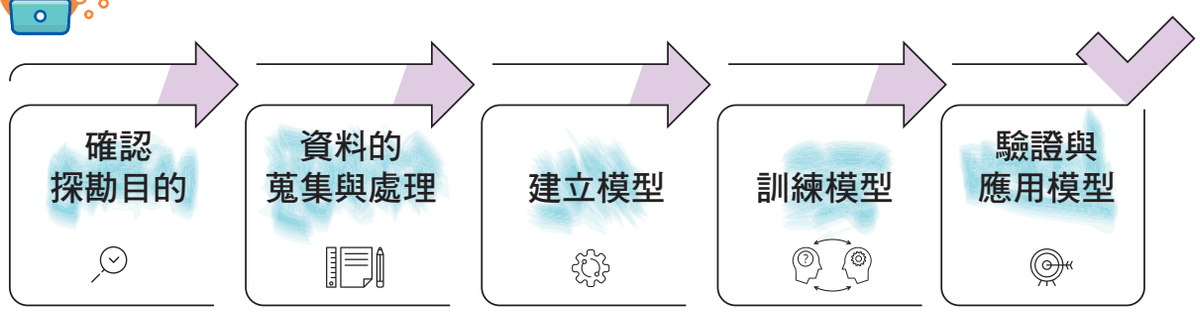


圖10-1-3 資料探勘的流程

01

確認探勘目的

確認探勘的主要目標，並評估所需的人力、資源、時間等。

例如：目標訂為在顧客的購物車中，探索商品購買的關聯性。

02

資料的蒐集與處理

運用巨量資料蒐集的方式（如外部開放性資料、內部日誌檔、網路爬蟲、物聯網感測器等）進行資料的蒐集後，再將資料進行異常過濾、初步處理、資料優化、結構化等處理。

例如：超市將結帳後的每一筆結帳單的明細都加入資料庫中，如購買商品的種類、金額、結帳日期等。

03

建立模型

依據目的和資料選擇適當的資料探勘模型，並運用演算法和統計學等建構初步的模型。

例如：選擇運用關聯分析探討帳單中「若購買A商品也會同時購買B商品」的支持度（Support）與信心水準（Confidence），來挖掘數據間關聯的強弱。

補給站

支持度

支持度是指在所有樣本空間中，出現某項目的比例。例如：在 1,000 筆交易紀錄裡面，有 50 筆紀錄包含「啤酒」這項商品，則其支持度記為 50 / 1000 ＝ 5%。

信心水準

例如在購買 A 商品的訂單中（樣本空間），同時也選擇 B 商品的訂單（事件）比例。數學中常以條件機率 P(B｜A)表示。

我們以前面「尿布與啤酒」為例，若交易數據共有 100 筆，同時出現尿布和啤酒的訂單有 20 筆，那它的支持度即為 20/100＝20%；若交易數據中出現 50 筆尿布，則在購買尿布的前提下，同時又購買啤酒的信心水準為 20/50＝40%。

04

訓練模型

利用蒐集的資料輸入模型後，持續不斷地調整模型參數，使其最佳化。

例如：超市結帳的交易數據持續不斷地增加，所得到的數值也會因為輸入的資料而持續變動，因此超市經理仍需要再尋找更多的關聯規則才能符合市場的動態需求。

05

驗證與應用模型

以新產生的數據資料輸入模型後，驗證模型的正確性，並嘗試解釋探勘結果，再導入實務應用中。

例如：超市得到的兩項數據——支持度和信心水準必須經由專家來解釋，並實際在商場中將兩項商品擺設在附近，才能驗證其推論是否有效。

並不是每一種資料探勘的方法都和上述的流程相同，每種不同的探勘分析方式都有其特定的步驟。如分類分析（Classification Analysis）、分群分析（Cluster Analysis）、關聯分析等，每種分析方式又有不同的演算法，因此必須選擇合適的分析工具才能得到正確的結果。

10-1-2 常見的資料分析方法

分類分析

分類分析法經常用來區分資料為何種類型，此方法會利用欲處理的問題，找出相關已知的資料類型，接著再對這些類型的資料作分析。例如：銀行透過貸款人的信用紀錄以及收入情形，分析是否核准貸款；商家依照會員的消費金額分析消費力，並將所有會員依序分為高消費群與低消費群。

以下將介紹如何利用決策樹（Decision Tree）、K最鄰近分類法（K-Nearest Neighbor，KNN）以了解分類分析在實務上應用的方法。

> 「監督式學習」、「非監督式學習」等機器學習的方法，將在本章10-2節進行詳細說明。

1.決策樹分析法

決策樹是一種常見的資料探勘技術，主要是使用樹狀分枝的概念來作為未來的決策模式，是一種**監督式學習**的分析方法。決策樹（模型）的組成（如下表10-1-1）有三大要素：

表 10-1-1 決策樹的結構

決策樹內部節點（非末端樹葉節點）	可能的屬性值	最終決策的分類
決策因子	決策樹的邊	決策樹的樹葉節點
用來進行分類的過程中所經過的決策因子（特徵），以樹狀結構當中來說，其為內部結點（非終端的樹葉節點）。如下圖10-1-4的抗原快篩、核酸檢測。	決策因子所產生的屬性值。如下圖10-1-4的陰性或陽性。	作為最終決策的分類。如下圖10-1-4的真陰性、真陽性、偽陰性、偽陽性。

我們以表10-1-2作為例子，建立一棵「是否確診特殊傳染疾病的決策樹模型」。

圖10-1-4是一棵「是否確診特殊傳染疾病的決策樹模型」，從樹根開始，依決策因子的答案依序往下走，最終決策的分類即在最後到達的樹葉節點（方框），若患

表 10-1-2 範例：特殊傳染疾病分類表

抗原快篩	核酸檢測PCR	類型
陰性	陰性	真陰性
陰性	陽性	偽陰性
陽性	陰性	偽陽性
陽性	陽性	真陽性

者最後的分類為偽陰性或真陽性，則歸類為確診，而分類為偽陽性或真陰性者，則歸類為非確診。

我們發現關鍵的決策因子是核酸檢測，若為陽性，則歸類為確診，因此從核酸檢測即可判定是否為確診。

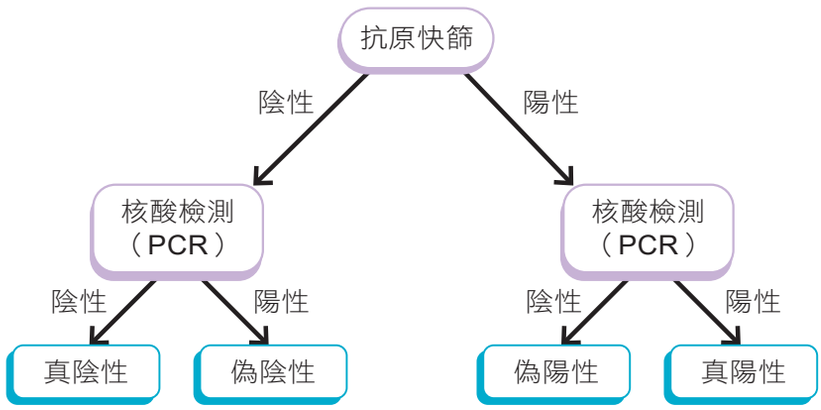


圖10-1-4 分類患者「是否確診特殊傳染疾病」的決策樹模型

決策樹的優點是可用簡單規則將欲分析的資料進行分類，而且無須經過龐大的運算，所以其在資料探勘領域中，是被廣泛運用的一種分類方法。

科技即戰力

若下表為某人周末是否去市立游泳池游泳的資料，請使用決策樹，預測未來各種條件、情況下，這個人是否會去泳池游泳。

水質	清	普	清	濁	普	濁	清	清	普	濁
人潮	多	少	多	多	少	多	少	多	多	少
價格	高	低	低	高	高	低	高	低	低	高
是否游泳	是	是	是	否	是	否	是	是	是	否

2.K最鄰近分類法

K最鄰近分類法屬於機器學習中的**監督式學習**方法，具有準確度高、不易受到少數異常資料的影響、適合用在非線性的分類問題等優點。準備一個已分類的資料集，將欲分類的新資料設為中心（圓心），並且以預先指定的K值，向外擴展直到取得K個最鄰近的資料（已知分類的訓練樣本）進行分類的數量比較，最終將新樣本資料歸類為出現最多次的分類標籤，即相似類別的資料點彼此更接近。

而該如何選擇合適的K值呢？首先我們會盡量取奇數作為K值，避免有平手的情況發生；再者，將樣本中的所有資料分成不同的子集合（訓練集和測試集），對訓練的模型進行交叉驗證（Cross Validation），以求能正確分類的K之平均值。舉例來說，假設有100筆樣本資料，第一次先使用前90筆進行模型訓練，最後10筆進行測試，依序再用第80到90筆進行測試，反覆地以不同資料集測試。

現在我們想運用K最鄰近分類法將新加入的資料進行分類，在K最鄰近分類法中的「K」可以是任何數字，藉由計算圖10-1-5K點中的分類（荔枝、芒果）的數量，將最終新加入的資料點歸類為計數較高的類別。

當K= 3 時，新資料將被歸類在芒果（荔枝：芒果 ＝ 1：2）；當K= 5 時，新資料將被歸類在荔枝（荔枝：芒果＝ 3：2）。可見K的值是影響新資料分類的關鍵值。當K的值太小，結果容易受到少部分的鄰近值影響；當K的值太大，則可能永遠以樣本數較多的資料為新資料的分類，失去取K值的意義。並非所有資料都適合用K最鄰近分類法進行分類，當樣本資料的分類沒有明顯的特徵值區分開來，則容易導致新資料的分類誤判，例如：櫻桃、葡萄的酸度、甜度、半徑大小皆相近，因此不易進行分類。

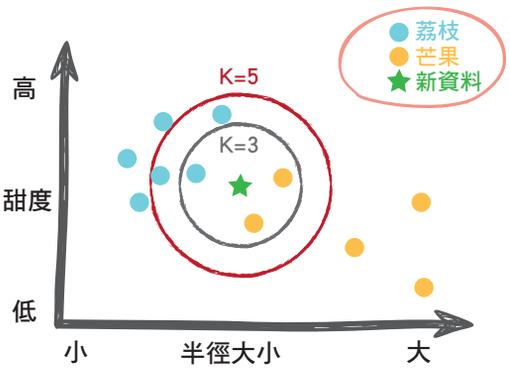


圖10-1-5 荔枝／芒果的甜度與大小分布圖

分群分析

分群分析同樣也是進行資料分類，但比起分類分析，分群分析並沒有事先明確定義好的類別或是特徵，是一種**非監督式學習**的分析方法。分群分析是將大量的數據區分出特徵相似的**群集**（Cluster）。**K-means分群分析法**就是透過這個概念將資料進行分群，將相似的數據歸類為一起，每一筆數據只能有一個分組，每一組稱為群集，這樣的方法常被用在客戶分群、特徵抽象化、非結構化資料分析等應用。

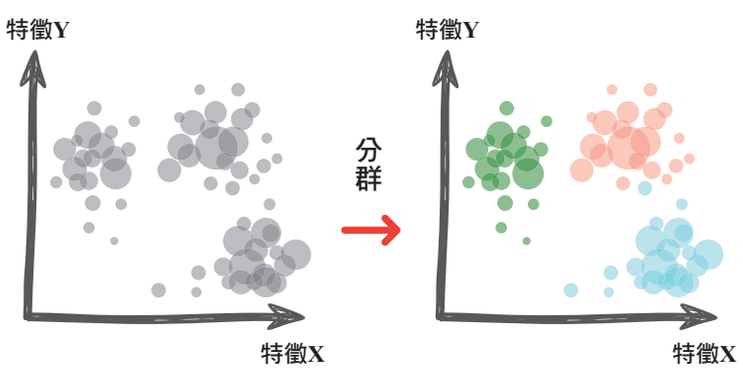


圖10-1-6 分群分析示意圖

K平均演算法（K-means）

演算法從使用者輸入K（我們想要的群集數量）開始將資料分成K群，其中先隨機選取K個資料點作為該群集的群集核心，接著會進行兩步驟：

STEP 01 **將每筆資料歸類到離自己距離最短的群集核心（以直線距離）。**

STEP 02 **依據該次分群結果，修正該群的群集核心位置（以該群集點的平均值）。**

進行完兩步驟後，因為群集核心（K個中心點）的位置將個別更新到新的位置，此時須反覆進行上述兩步驟的運算，直到核心位置不再更動為止。

上述演算法進行兩步驟運算，其每一次更新，都會讓每個群集的所有資料點到群集核心的直線距離越來越小，也代表演算法正在反覆運算的過程中收斂，藉由群集核心的調整，逐漸讓同一群集內的同質性（相似性）越高。

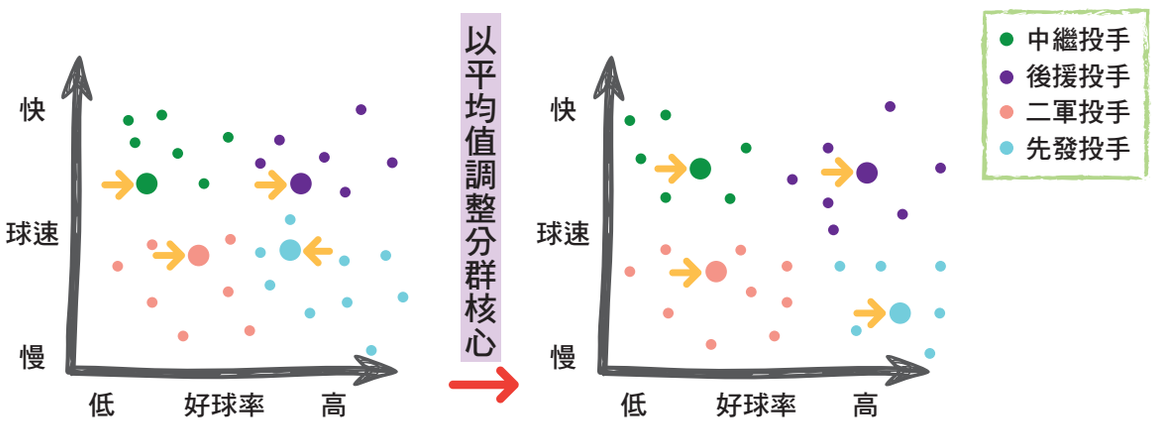


圖10-1-7 以K平均演算法分析不同投手的場上功能性

以圖10-1-7來說，若我們先將多位球員以球速和好球率（控球）作為特徵值，製作成「球速—好球率的散布圖」，並以K平均演算法的方式將球員們分為四個群集，而以K平均演算法完成分群後，球隊會希望找到的是同時兼具控球和球速的後援投手，且這些投手屬於即戰力選手。

首先，在散布圖左下方的點（粉紅色群集）屬於非即戰力的練習型二軍投手，故不考慮網羅；左上方和左下方的球員分別屬於中繼和先發型投手，也不在考慮範圍內；因此在考量到須網羅「可以直接投入比賽的後援投手」的前提下，球隊可以優先網羅分群在紫色的球員人選（圖10-1-8）。

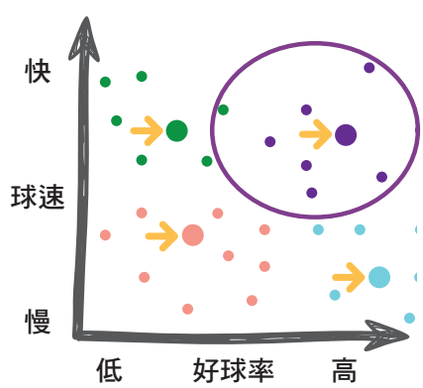


圖10-1-8 可以直接投入比賽的後援投手

線性迴歸分析（Linear Regression Analysis）

運用過去歷史的資料（點），建立線性迴歸方程式的模型預測未來的趨勢，希望透過這些點，找出最符合這些點的直線，常用在預測匯率、股市等價格，是一種**監督式學習**的分析方法。

假設僅考慮某商店的霜淇淋銷售量與氣溫之間的相關度，以下為店內的霜淇淋每日銷售量與氣溫的表格及線性關係，座標為直線且可以代表資料點的整體變化趨勢，進而用以推估其他鍵入的資料。經過數筆資料的鍵入，選擇趨勢線，即可知道此趨勢線方程式為 Y = 6.8986X - 8.6792 ，未來該商店只要知道 X 軸的當日氣溫，即可預測出 Y 軸的當日霜淇淋銷售量。

表 10-1-3 氣溫與霜淇淋銷售量的資料紀錄

氣溫（℃）	霜淇淋銷售量（支／日）
35	230
31	205
29	200
26	165
23	150

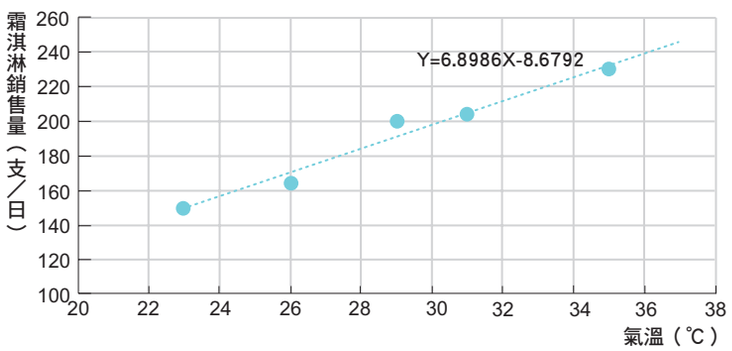


圖10-1-9 以線性迴歸進行分析及預測

科技即戰力

下表是近年汽油價格（元／公升）以及某城市捷運的總運量（百萬人次）的資料，請完成下列問題：（※可以使用 Microsoft Excel、LibreOffice Calc 等試算表軟體求解）

1.請畫出汽油價格與總運量的散布圖，並找出一條最接近所有資料的直線方程式。
2.請預測汽油價格愈高，旅客會不會更多？
3.若汽油每公升 30 元，請預估旅客總運量為多少？

年份	汽油價格	旅客總運量
2008	30.22	29.47
2009	26.79	43.34
2010	29.79	46.01
2011	31.90	49.64
2012	33.87	56.47
2013	34.90	60.71

年份	汽油價格	旅客總運量
2014	33.59	61.31
2015	25.31	60.20
2016	23.71	63.10
2017	25.97	63.77
2018	29.37	64.72
2019	28.53	65.39

關聯分析

利用大量的交易資料進行分析，找出資料項目之間存在的**關聯性**以及**規律性**，是一種**非監督式學習**的分析方法。關聯分析較常見的應用便是「購物籃分析」，如前面提過的尿布與啤酒的故事；又如某電信行彙整營業報告所得到的統整數據，數據顯示購買手機充電線的客人通常也會同時購買電源轉接器，經過此資訊的分析結果，電信行便可依此進行合購促銷或是置入性行銷等銷售策略，以提升產品銷售額。

10-2 機器學習的概念與應用

人類自幼學習辨別動物的過程，往往是來自於生活中所獲取的大量經驗，再將這些經驗歸納後，逐漸地可以區分不同動物間的特徵（圖10-2-1）。而機器學習也是透過蒐集大量樣本資料，並將其輸入演算法後，找出適合的預測模型。

藉由這種方式，使得機器學習在各種領域都得到了廣泛的應用，包括自然語言處理、購物籃分析和電腦視覺等。

圖10-2-1 如人類幼年時，可能會透過動物園參訪等實際接觸，或是藉由卡通、影片、圖卡等方式，學習辨別不同動物的特徵，並透過這些經驗，逐漸修正自己辨識的規則。

10-2-1 機器學習的概念——以監督式學習為例

假設我們想要購買非全新的 iPhone 手機，所以上網瀏覽二手手機的轉賣平台，從查詢的過程中發現了價格的趨勢：假如某全新手機的價格落在20,000元，則已經使用1年的二手舊機價格是15,000元，已使用 2 年的則是10,000元。從中我們發現一種規律：手機的價格可能取決於手機已被使用了多久。每增加1年，價格下降 5,000元，但標價不太會低於8,000元。

以機器學習的模型來說，上述這類特定的方法稱為「**迴歸**（Regression）」，是一種基於已知的歷史資料，來預測未來的數值（價格）。

在日常生活中，我們總是希望能得到一個固定的通式，來解決相關的問題，但顯然這是一件不太容易做到的事情。以上述的情境為例，我們或許還會發現除了新舊的程度以外，不同的顏色、銷售量、使用痕跡、是否有保固等眾多複雜因素（**特徵**），也會影響二手手機的價格。

因此，面對各種複雜的問題，就需要電腦替我們算出適合的公式（**模型**）。我們提供電腦大量的資料，讓它從所有和價格有關的因素推導公式的參數。所有機器學習的最終目標就是將大量資料輸入，並輸出預測的結果。

在進行機器學習時，通常可以透過三個要點來訓練機器：

1. 大量的樣本資料 → 2. 特徵（Feature） → 3. 學習演算法

將大量的樣本資料輸入至學習演算法，使得機器能夠擷取特徵，推論出合適的模型，並且藉由新資料驗證模型的預測結果。

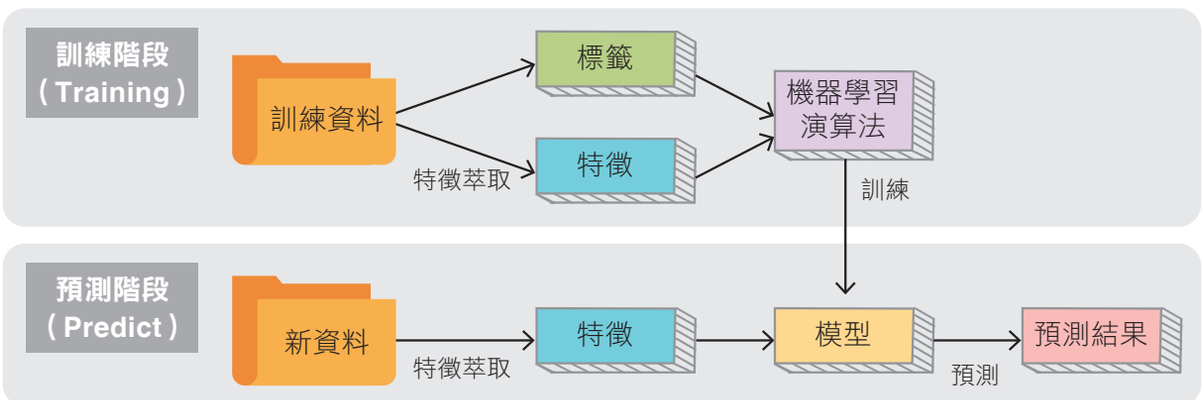
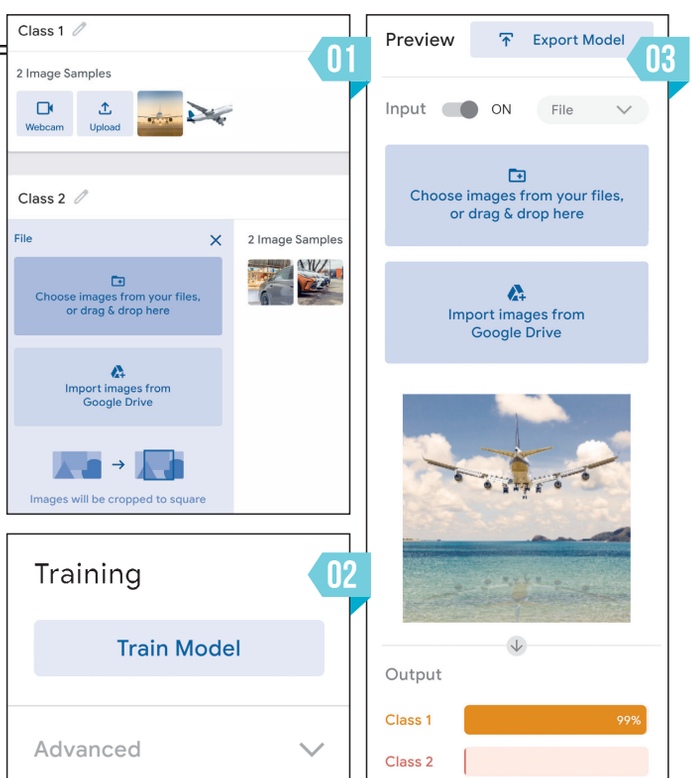


圖10-2-2 機器學習可分為訓練和預測兩階段

科技即戰力



影像辨識大挑戰

「Teachable Machine」是Google提供的線上機器學習平台，可運用快速且簡易的方式訓練出機器學習的模型，讓使用者免除撰寫程式碼的過程，最終能讓使用者的電腦裝置進行影像、聲音、姿勢的辨識。請同學們試著透過網站的Image Project選項（https://teachablemachine.withgoogle.com/train/image），進行模型訓練。

Step 1.輸入影像：可選擇圖片檔案或是透過視訊鏡頭擷取圖片（至少要兩種以上的不同類型才能進行辨識）。

Step 2.訓練模型：只要點擊「Train Model」按鈕，平台就會立即用輸入的影像進行模型訓練。

Step 3.測試：透過視訊鏡頭或是上傳圖片，進行動態圖片的擷取，並且分析、輸出此影像屬於各類別的機率。

10-2-2 機器學習的方法

依據訓練時所輸入的資料是否帶有**標籤**，以及判別過程的不同，可以將機器學習的方法分成常見的三大類型：**監督式學習**、**非監督式學習**及**強化式學習**（Reinforcement Learning），如圖10-2-3。

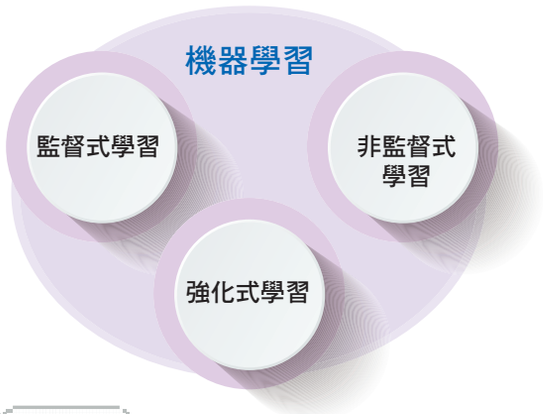


圖10-2-3 機器學習含括了三種學習方式

監督式學習

在訓練的過程中輸入大量有**標籤**的資料，訓練機器學習的模型（圖10-2-4）。其中，告訴機器這張圖片的預測資訊，叫作標籤。輸入的資料可以是「有標籤」的分類資料或是一個連續的值（迴歸分析），例如：將大量有標記鳳梨標籤或橘子標籤的照片輸入學習演算法後，預測模型會將「葉片帶有尖刺且布滿格狀的球體」分類為鳳梨，而「表面橘色且具有點狀毛細孔的球體」為橘子。之後就可讓機器分辨尚無標籤的照片是鳳梨還是橘子。實際上的運用如上一節資料探勘中所提到的分類、迴歸等資料分析方式。

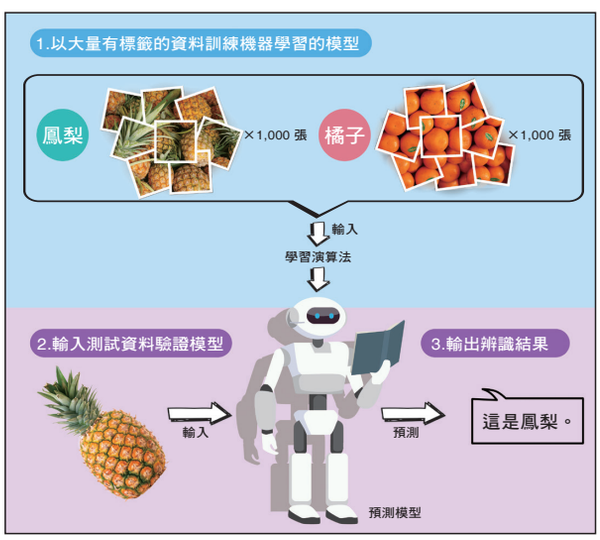


圖10-2-4 監督式學習的訓練過程與驗證模型

非監督式學習

非監督式學習的方式並不需要在訓練前將資料進行標籤，而是以沒有標籤的資料來訓練機器，在學習時，機器會「自動」找出潛在類別的特徵，並且在反覆測試後，將學習後得到的模型，以新的資料輸入驗證，如上一節提到資料探勘中的分群（圖10-2-5）、關聯分析。

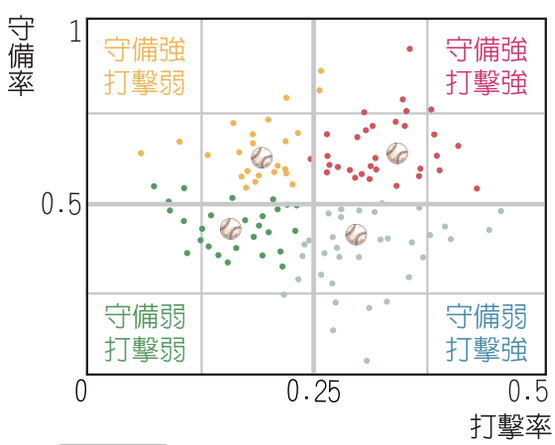


圖10-2-5 非監督式學習的應用：教練運用K-means分群分析法將選手分為四大類

強化式學習

強化式學習強調的是機器隨時透過環境而採取相對應的行動，例如：AlphaGo可以根據對手前一個動作立即調整自己下一步的動作，並且使用獎勵（Reward）和懲罰作為積極和消極行動的信號。根據採取的行動當成輸入的資料，進行逐步學習，並且根據獎勵更新狀態（圖10-2-6）。這個方法直接從互動中進行學習，目標是找到合適的動作模型，以最大化機器累積的總獎勵信號。

強化式學習亦適合用在有時間限制的情境進行探索，例如：導航一開始的路線決定並無法馬上考慮到所有路線的路況，通常僅能先在多條路線中決定某條路線（如路徑最短），但當選擇該條路線後，系統可以根據即時路況或駕駛人臨時改道，隨時更新最佳路徑，逐步修正。這樣的方法在許多領域（如控制理論、統計學、教育學、基因演算法等）都有相關研究與應用。

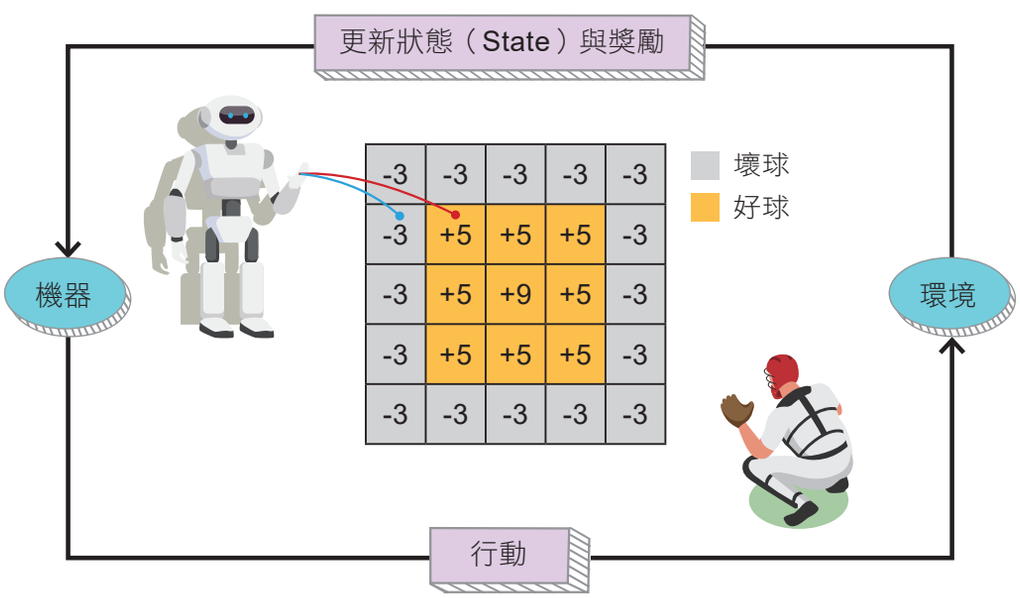


圖10-2-6 強化式學習模型的訓練過程

科技即戰力

ChatGPT——最佳化的語言對話模型

OpenAI公司在2022年12月正式推出了「ChatGPT」互動式對話機器人，引發各界關注。ChatGPT透過和使用者進行對話，分析使用者輸入的文字，給予後續互動相對應的回答。在過程中，ChatGPT也容許自己在回答上的失誤，並且透過強化式學習的回饋機制，讓使用者對於每次對話給予正向或負向的回饋。同時也能夠拒絕不適當的提問，符合我們在chapter 8學到有關於「負責任的AI」中「安全性」的規範。

10-2-3 機器學習的應用

機器學習的唯一目的是運用輸入的資料來預測其他資料的結果，能夠從巨量資料中自動分析並獲得規則，對新的資料進行預測，而機器學習的模型也可以視為資料探勘的工具，兩者的關係是相輔相成的應用。

以下為機器學習的應用實例：

自然語言處理（Natural Language Processing，NLP）

自然語言是人們溝通時自然地發展出來的語言。「自然語言處理」的目標，就在於讓電腦理解或是運用人類語言，如Apple iPhone 的 Siri 語音辨識、Google 支援各國語言的即時翻譯（圖10-2-7）或是互動對話機器人ChatGPT 等，都是自然語言處理的應用案例（圖10-2-8）。

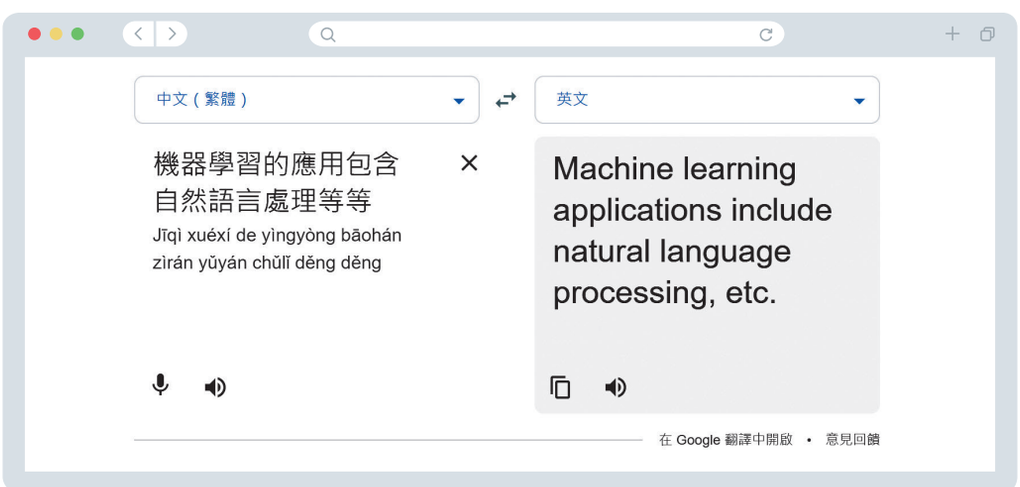


圖10-2-7 Google各國語言即時翻譯

1.請先到OpenAI網站（https://chat.openai.com/auth/login）進行註冊，目前這項服務仍是免費的。
2.請嘗試和ChatGPT進行連續對話，並和同學討論是否滿意這次的回覆。
3.請嘗試讓ChatGPT替你編寫、完成一段C++程式。
▲在每個回應後皆能讓使用者給予回饋。

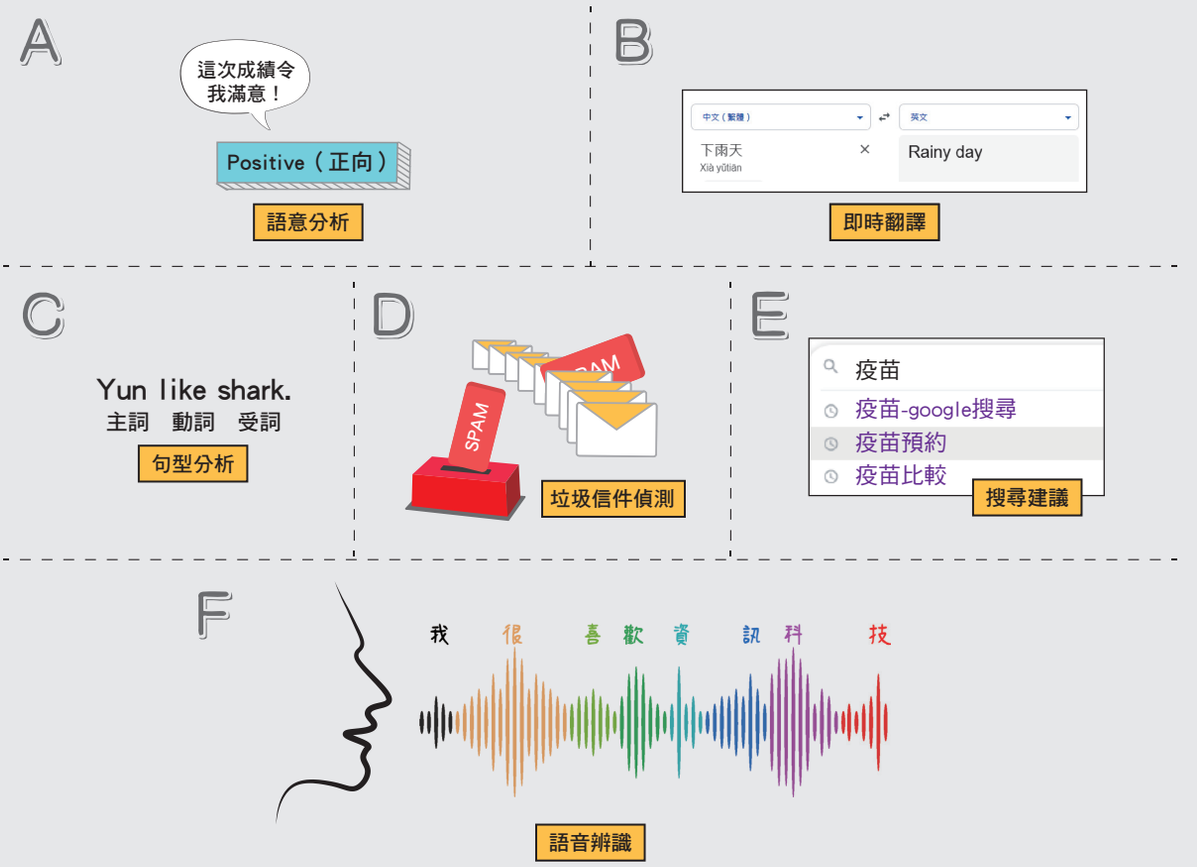


圖10-2-8 自然語言處理常見的應用

電腦視覺（Computer Vision）

電腦視覺的目標是使用蒐集到的「圖像資料」來理解世界上的事物，利用電腦、攝影機等設備模擬人類的肉眼進行偵測、量測、追蹤並分析這些資料訊息，再運用電腦進行檢測和識別，然後提供適當的輸出。這就像是將人類的智慧和本能傳授給電腦。電腦必須解釋它所看到的，然後進行適當的分析或採取相應的行動。

圖10-2-9 7-11 X-store無人商店

一般常見的如臉部辨識、車牌辨識、無人商店（圖10-2-9）和自動駕駛車等，都是電腦視覺的應用場景與研究領域。

推薦系統（Recommender Systems）

將使用者過去的購買／使用紀錄輸入系統，並藉由推薦系統分析出使用者可能想要購買的商品。如Amazon精準地將商品推薦給他們的顧客，藉此提高訂單成交率；Spotify、Netflix及YouTube影音平台總是能將使用者喜愛的節目內容推薦到使用者眼前，也會針對每位使用者的偏好，推送不同的節目內容（圖10-2-10）。

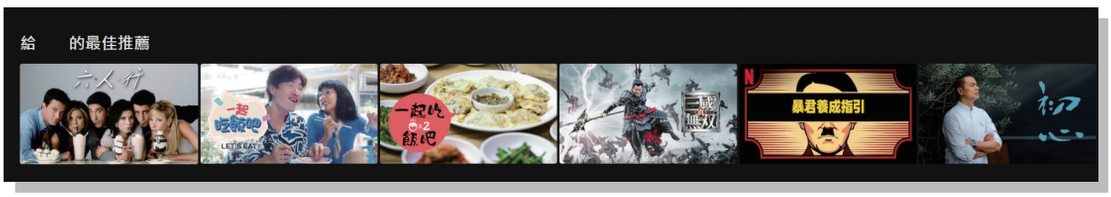


圖10-2-10 影音平台依據使用者觀看紀錄推薦相關影片

醫療診斷

除了基本的生理檢查外，醫療上也應用了機器學習進行各項診斷，其主要是進行病灶的偵測或證實疾病的存在，並對其進行準確的辨識。透過分析患者的各項檢查數據和影像，包含醫學上所得到的診斷、測試、患者的症狀、影響程度等，以提升醫學的診斷準確性，使醫療人員能夠根據相關檢測結果，聚焦並鎖定患者可能罹患疾病的狀況，藉此降低誤診率，增進醫學診療的整體效能與品質。

IT補給站

科幻電影「關鍵報告」主要描述在未來的世界中，科技已完全改變警方辦案的方式，預防犯罪系統能偵測到犯罪企圖，預測某些重大的刑案即將發生，而警方也能在案發前就抵達現場，因此大幅降低了城市的犯罪率。

電影中可以看到警方運用影像辨識與資料探勘等技術，發現可疑人物即將要進行的犯罪行為；日本的便利商店運用人工智慧技術，預測疑似竊盜者即將犯罪的機率，運用監視器鏡頭搭配人工智慧軟體偵測店內顧客的肢體動作與面部表情，如坐立不安、神情焦慮或行跡可疑等等，進而找出可能準備犯罪的人物，並預先告知店員留意。

這項技術同時也可以運用在工程現場增加勞工安全（圖10-2-11），或是防範在月臺上蓄意跳下軌道的人員等。

但由於人權意識的高漲，運用科技預測的技術確實提高警方辦案的效率，卻也帶來不少爭議，像是針對執法部門監視、蒐集到的人們的各類活動畫面，是否有侵犯人權或內部人士可能進行非法濫用等等的隱憂，也引發外界質疑的聲浪。

然而，隨著技術日漸成熟，人們更需要思考如何在避免技術遭濫用而導致劇烈社會動盪的同時，使其成為能夠造福人類的科技。

圖10-2-11 影像辨識等技術應用於工程現場

課堂檢測

( ) 1. 影音平台依據許多用戶的觀看紀錄之相似性來推薦影片，主要是運用何種技術？
(A)資料備份 (B)資料探勘 (C)資料儲存 (D)資料處理。

( ) 2. 若某百貨公司想進行週年慶行銷，已先設定出三種不同的顧客類別，並希望讓系統透過每位顧客的來店次數、消費金額等資料，自動將顧客歸類到三大類別所屬的群集之中，試問該公司可運用下列哪一項技術作為分析方法？
(A)分群分析 (B)關聯分析 (C)迴歸分析 (D)分類分析。

( ) 3. 某大學教授若想運用學生的在校成績，預測未來就業時的年收入，其可能採用何種分析方法？
(A)關聯分析 (B)迴歸分析 (C)分群分析 (D)分類分析。

( ) 4. 許多購物網站會在我們購買甲商品後，接著推薦我們購買與甲商品相關性高的乙商品。請問這種推薦系統最可能使用到的為以下哪一種資料探勘的技術？
(A)群聚分析 (B)分類分析 (C)喜好分析 (D)購物籃分析。

( ) 5. 提米常常使用手機中的人工智慧語音助理功能。語音助理嘗試替使用者解決問題後，在關閉APP前皆會詢問使用者是否滿意這次的回答，試問這種行為較屬於何種機器學習方式？
(A)監督式學習 (B)半監督式學習 (C)非監督式學習 (D)強化式學習。

( ) 6. 現在許多停車場都採用車牌辨識系統，原理是利用蒐集大量的車牌，訓練電腦學會辨識車牌號碼。請問此種車牌辨識系統最有可能是應用以下的何種技術？
(A)物聯網 (B)自然語言 (C)機器學習 (D)語音辨識。

( ) 7. 嘟嘟是連鎖咖啡店店長，他想要分析訂單中「買A咖啡的顧客是否也會同時購買B蛋糕」的消費行為，看能不能將這兩項商品結合成套餐，提升銷售業績。請問他最有可能應用以下何種分析技術？
(A)分類分析 (B)迴歸分析 (C)分群分析 (D)關聯分析。

( ) 8. 下列何種技術，較不可能是利用機器學習技術所建立的？
(A) GPS系統 (B)臉部辨識系統 (C)購物推薦系統 (D)垃圾簡訊過濾系統。

( ) 9. 下列何種機器學習的技術適合運用在道路導航技術中，可隨時根據使用者的GPS座標，進行導航路線修正？
(A)監督式學習 (B)強化式學習 (C)非監督式學習 (D)以上皆非。

( ) 10. 小美想要藉由輸入大量的語音，並逐一為各語音檔案標註其內容為何，開發出語音辨識系統，作為期末的專題發表。請問小美最有可能是使用哪一項機器學習的技術，進行此語音辨識系統的開發？
(A)非強化式學習 (B)強化式學習 (C)非監督式學習 (D)監督式學習。

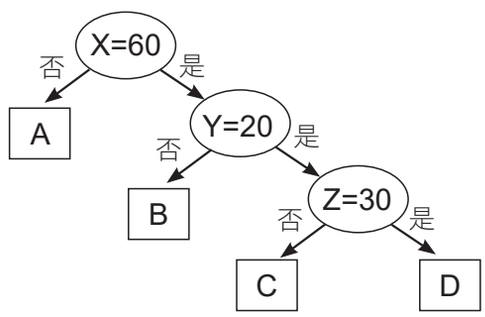
( ) 11. 喵喵將許多水果的圖片直接輸入到機器學習系統，但每張圖片並無標註水果名稱，試問下列何種機器學習方式較適合替喵喵區分不同水果的類別？
(A)非監督式學習 (B)非強化式學習 (C)強化式學習 (D)監督式學習。

( ) 12. 下列何種分類演算法，是採用類似多數決的方式決定其分類類別？
(A)決策樹分析 (B)迴歸分析 (C)分群分析 (D)K最鄰近分類法。

( ) 13. 某職棒球隊添購了一臺具有機器學習能力的自動投球機，機器在每次投球後，皆會藉由「是否通過好球帶」給予加分或減分的回饋機制，進而修正下一次投球的進壘角度，以提升其好球率，試問這是屬於下列何種機器學習模式？
(A)強化式學習 (B)非強化式學習 (C)非監督式學習 (D)監督式學習。

( ) 14. 下列何者並非應用資訊科學的例子？
(A)系統智慧判斷信用卡刷卡是否為盜刷
(B)商店店員親自結算每日收銀金額是否短少
(C)超市藉由關聯分析將尿布和啤酒進行綑綁銷售
(D)微軟紐約研究院的經濟學家利用資料探勘的模式成功預測大部分的奧斯卡獎項。

( ) 15. 下圖為一決策樹，若是條件為X = 60，Y = 20，Z = 30，則決策的結果為下列何者？
(A)A (B)B (C)C (D)D。



( ) 16. 酒廠先蒐集許多酒類的樣本，檢驗其化學特徵並整理，接著透過分類決策樹模型進行預測及判斷酒的品質和所屬等級，此技術稱為下列何者？
(A)資料盤查 (B)資料搜尋 (C)資料探勘 (D)資料加總。

( ) 17. 下列何者的應用中，最不可能是自然語言處理可以做到的？
(A)人臉辨識 (B)搜尋建議更正 (C)語音辨識 (D)機器翻譯。

( ) 18. 監督式學習與非監督式學習，兩者之間最大的差異為以下何者？
(A)預測之結果不同 (B)一個有人監督，另一個無人監督
(C)訓練的資料量多寡 (D)資料是否有標記的數據。

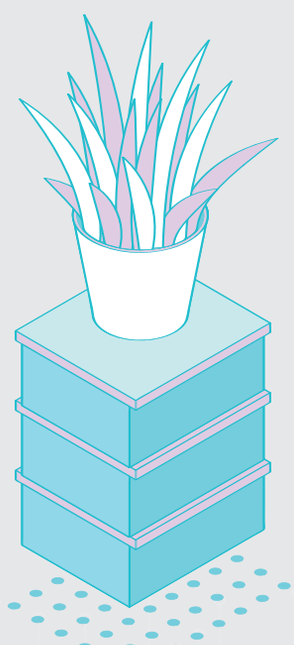
( ) 19. 音樂平台根據閱聽人喜歡的內容進行歌曲推薦，此為何種機器學習的應用？
(A)電腦視覺 (B)醫療診斷 (C)推薦系統 (D)自然語言處理。

( ) 20. 透過分析患者的各式檢查結果數據，包含醫學上所得到的診斷、測試、患者的症狀、影響程度等進行分析，提升醫學的診斷準確性，此為何種機器學習的應用？
(A)藝術創作 (B)電腦視覺 (C)機器控制 (D)醫療診斷。

主題五

資訊科技應用

chapter 11
專案管理與數位協同共創 262



chapter 11

專案管理與數位協同共創

近年來，科技進步與國際化造就工作型態的快速變遷，許多企業經營模式也隨之改變，不少企業從單一國家開始拓展到各個國家進行人才招募，甚至在不同國家紛紛建立起跨國的分公司。

也由於許多工作愈來愈趨向高度專業化，產生了如個人接案、工作室、派遣、顧問等新興工作型態，因此無論是公司內部的團隊，或是將內部工作外包給外部專業團隊執行，在管理上也都需要運用到「**專案管理**」的技術。

在校園中，也可以運用專案的方式完成許多活動，例如：小論文、科展、社團成果分享、期末分組報告、各類競賽等，都可以運用科技工具協助我們產出最終結果。在這個章節我們將依序介紹專案管理的概念、**協同共創**的工具、**版本控制**，以及專題實作範例等數位協同共創的概念與工具使用。

11-1 專案管理的概念與應用

「**專案**」是為了達成特定的目標、計畫、項目所進行的任務，因此專案會明訂確切的起迄時間，在這個時間內完成一連串的工作計畫，以達成最終的特定目標。

而「**專案管理**」是運用科學化的方式執行專案，以期能更有效率地運用有限的資源和時間，將目標如期、如質、如預算地達成，並充分滿足任務需求。

11-1-1 專案管理的流程

專案管理會先將特定目標細分成可以執行的最小的單位——**程序**，每個程序會有自己的工作管控，從開始的資源輸入到最終的結果產出。有些專案管理的目標較著重在抽象的理想和願景，而程序通常則比較具體化且貼近現實執行面，通常是可以量化的數字，例如：想要提升學校升學形象（目標），國立大學錄取率達到80%以上（程序）、全校學測4科平均50級分以上（程序）等。

而專案管理的執行項目通常包含下列五大程序：

表 11-1-1 五大程序

項目次序	程序	說明	以開發軟體的系統為例
1	起始階段	定義專案所要解決的問題與範圍，並確認與專案相關之人員，如各項工作負責人、顧客等。	進行系統開發的需求分析與計畫（Requirements Analysis and Planning）
2	規劃階段	規劃工作項目、負責人員、預算分配、時程安排、所在位置、資源供應。	系統的規劃與設計階段，規劃演算法的虛擬碼與流程圖，並將工作細項與進度分配給工程師。
3	執行階段	整合規劃階段的資源，投入專案的執行。	系統實作階段，工程師們將規劃階段的演算法，以程式設計進行開發實作。
4	監控階段	定期檢視各階段進度，掌控專案進行以達成目標。	監控程式開發的進度，並將開發完成的程式，進行系統整合與測試，若測試過程中發現問題，則再回到上述階段，重新修正規劃或實作。
5	結束階段	檢視專案的成效，並讓產生的產品、服務或結果被接受，最終結束專案流程。	將系統進行移交與後續系統的維護。

為了提高專案執行的效率，可以運用專案管理的共同協作平台，將專案所有資訊集中管理和公告，包含文件、簡報、時間表以及其他資源等。此外，也可以選擇合適的通訊工具，如視訊會議軟體Meet或是即時通訊軟體Line等，讓成員得以不受空間的限制進行遠距離溝通，協調專案內容。

專案管理的特性

系統化地管理每項工作

管理者可以更全面性地檢視專案的進行，增加對專案的控制並且提升專案成功的機會。

明確的工作排程

將工作具體化，並將其進行排程。

有效運用資源

如人力、物力、預算的分配等。

面對突發狀況應變力強

對資源掌控度高，可即時調配資源。

11-1-2 專案管理的工具

利用Trello進行專案管理

Trello是一款能進行專案管理、時程規劃、任務分配的線上整合平台，具有容易操作、介面親和力高等特性，除了能提供管理團隊進行專案排程規劃與人力分配，也非常適合一般使用者進行工作進度的安排。透過簡易的視覺化介面，管理各個「看板」代表的專案任務，將工作流程圖形視覺化，幫助所有人完成細部流程執行與整體時程控管。

IT補給站

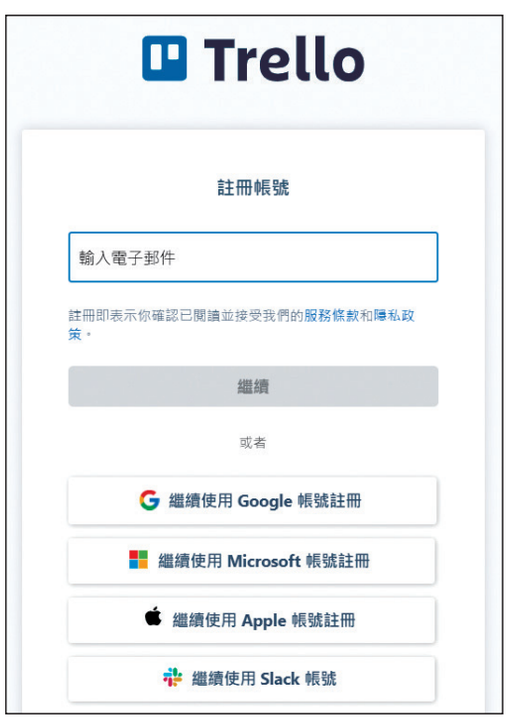
雖然澳洲軟體公司Atlassian已收購專案管理軟體公司Trello，但目前大多數的主要功能仍維持免費提供給一般個人用戶，若有需要更進一步的使用，則可以選擇付費加值功能。

科技即戰力

假設我們要跟朋友一起組隊參加「全國高級中等學校小論文寫作比賽」，必須依照比賽規定的格式（六大段落：「壹、前言」、「貳、文獻探討」、「參、研究方法」、「肆、研究分析與結果」、「伍、研究結論與建議」、「陸、參考文獻」）在期限內完成小論文。這時我們就可以透過 Trello 進行「小論文專案」的專案管理實作，像是擬定主題、分派各項任務、管理團隊人員等，有效率地達成「按時完成小論文」的最終目標。以下步驟將逐步引導大家完成這個專案。（建議三人一組）

Step 1

首先，進行註冊。建議可以使用 Google 帳號註冊，以利後續資源的整合。



Step 2

以團隊名稱（小論文製作團隊）命名工作區，並且可以在一開始就加入團隊合作夥伴。

Step 3

進入 Trello 初始工作面板。

Step 4

輸入看板名稱（小論文進度之專案管理），並點選「建立」。

進入「看板」（專案）後，可依自己的需求新增「列表」，如擬定小論文題目、相關文獻探討、擬定／執行研究方法、數據蒐集與整理、研究分析與結果等（圖11-1-1）。

圖11-1-1 Trello看板編輯介面

接下來在各列表中新增「卡片」（最小的工作單位——流程），如圖11-1-2。

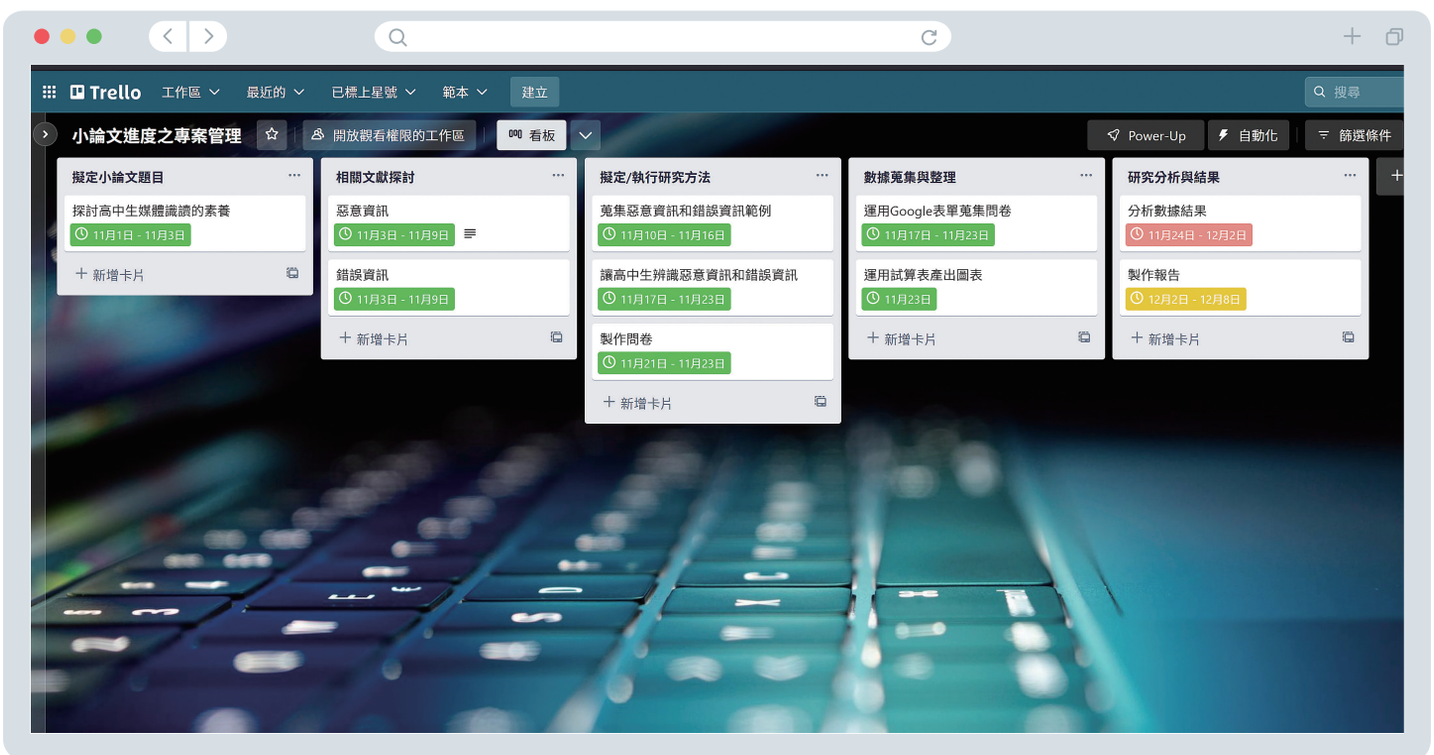


圖11-1-2 在各列表中詳列各項卡片，以便具體執行各項進度

整體來說，Trello可以分成看板（專案）、列表（工作分類）、卡片（可分配且執行的工作流程），只要把握這三個分層架構，就可以運用Trello進行有效率的專案管理工作。

新增了卡片後，可以對每張卡片進行設定調整，也可以隨時拖曳、編排調整列表與卡片的相對位置。

卡片如同流程一樣，是具體可行的，且多數是可以量化評估的工作，我們可以運用卡片進行各項任務的監控、追蹤。

Trello的卡片功能（表11-1-2）就如同功能強大的便利貼，在卡片中可以編輯許多重要流程訊息。例如：新增負責的成員、加上醒目的標籤顏色、擬定該任務的待辦事項與清單、規劃起訖日期與時間、添加所需附件資料等功能（圖11-1-3）。

表 11-1-2 卡片功能

名稱	詳細說明
成員	可以邀請成員加入該卡片任務，當卡片任務有負責成員時，成員就擁有編輯這張卡片的權限，讓該負責人可以即時編輯、記錄自己的進度，也可以讓其他成員了解該任務的負責人是誰、工作進度到哪裡，避免工作重疊或延宕。
標籤	可以善用顏色、背景的設定，區分卡片的屬性及狀態。例如：需要緊急處理的卡片任務，可以使用紅色代表急件；進度超前的卡片任務，可以使用綠色代表。
待辦清單	可以在卡片內新增待辦事項，讓使用者註記須完成的任務有哪些，並勾選各階段的工作任務是否如期完成。只要目前有完成的項目就勾選，而進度條也會同時顯示完成比例。
日期	設定卡片任務的起訖時程，並可以設定提醒通知即將到期、逾期等狀態功能，降低工作任務延宕的可能性。
附件	可以在卡片任務中新增檔案，讓團隊成員即時共享資料，包含裝置內的檔案（不限格式）、雲端硬碟（Google、One Drive等）。

圖11-1-3
編輯卡片各項功能的畫面（以「文獻探討」中的「惡意資訊」卡片為例）

藉由Trello，可以明確地掌控各項工作的期程安排，了解團隊每一位成員目前的工作量、工作進度，哪些工作正處於進度落後，以便立即安排人力支援該項任務；同時，成員們也可以知道目前專案的進展，了解團隊夥伴各自負責的工作任務，進而讓跨部門的夥伴溝通、協調有先後順序的任務，或是進行資源的調配，更可以即時回報自己負責的專案進度。因此善用Trello這類的專案管理工具，有助於提升專案成功的機會。

Trello也非常適合個人使用，例如：個人工作室、接案、短期工作目標的設定、旅遊計畫等，也都能運用本專案管理程式進行進度的管控，讓個人的時間和資源更清楚地被規劃和掌握。

IT 補給站

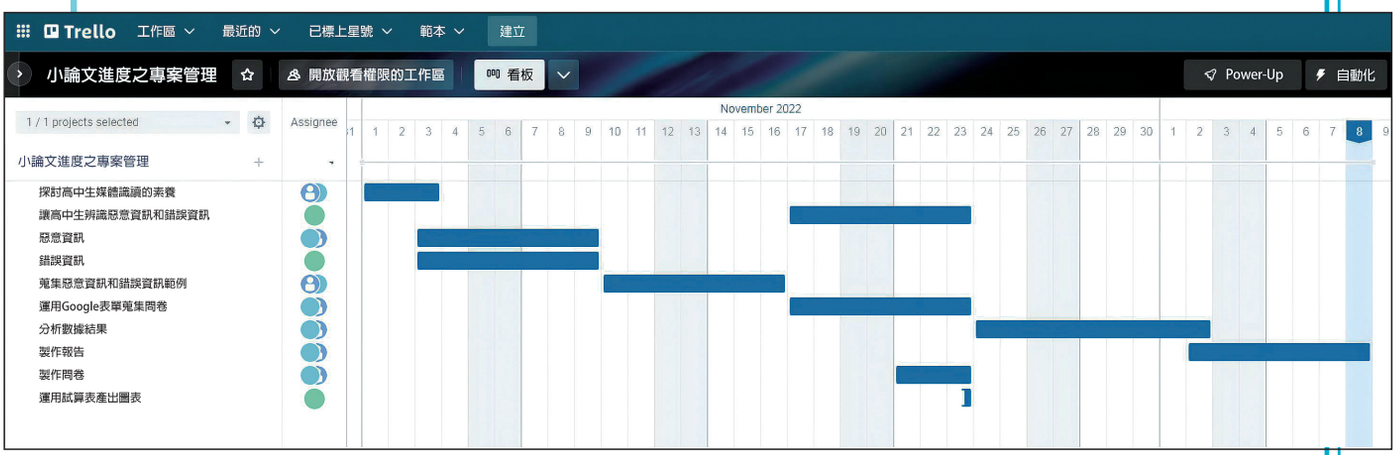
建立追蹤進度的利器：「Good Gantt Chrome Extension」
——可在 Trello 中使用的甘特圖外掛工具

Step 1 安裝外掛程式

在chrome線上應用程式商店新增Good Gantt Chrome Extension擴充功能，並確認授權。

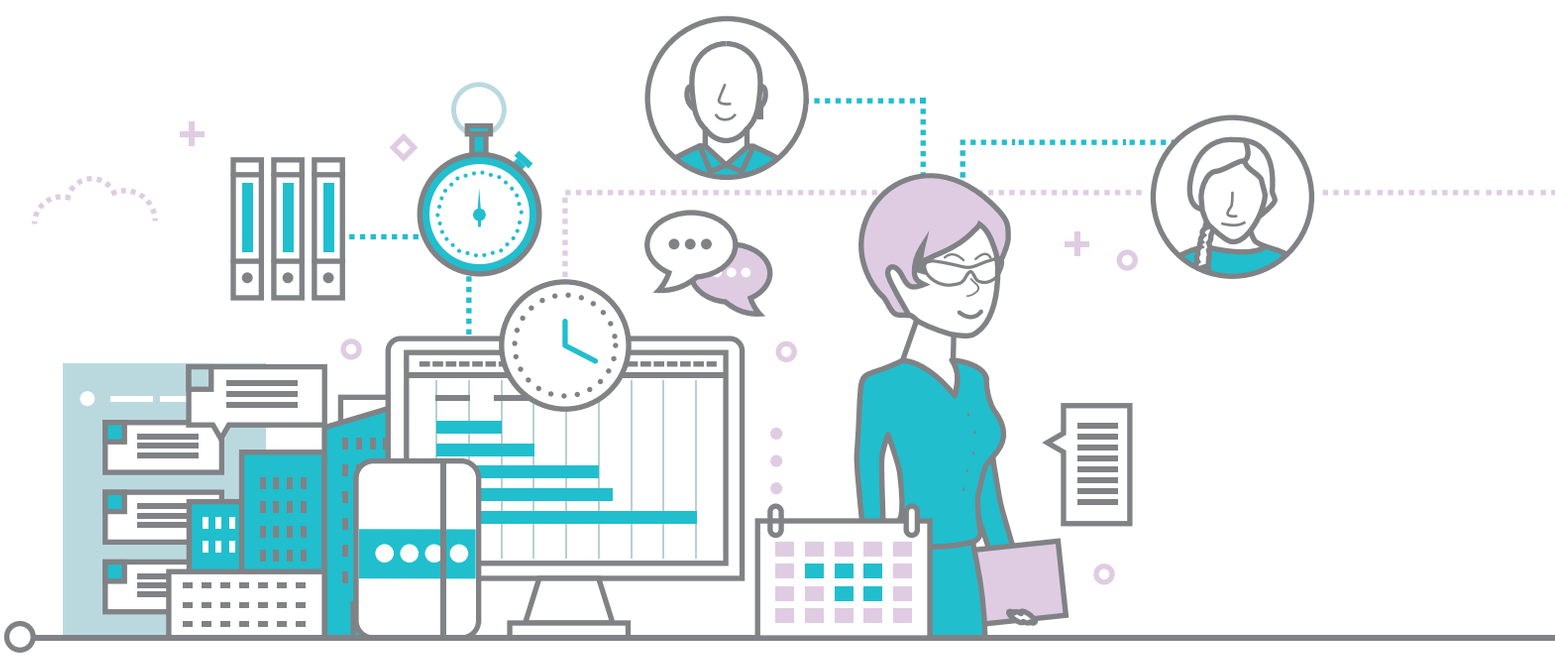
Step 2 打開甘特圖

如果安裝順利就可以看到Gantt的外掛按鈕在看板上，點擊Gantt，就可以進入「GoodGantt」的甘特圖模式，並看到「小論文進度之專案管理」看板上的任務整合在甘特圖上了。而點選每個進度的長條圖，皆可以詳細查閱該卡片的各項功能和資訊。



甘特圖

甘特圖是屬於條狀圖的一種類型，顯示專案、進度，以及其他與時間相關的系統項目其內在關係隨著時間進展的情況。其橫軸代表時間，縱軸代表工作項目，條形則代表工作項目的開始時間和完成時間。



11-2 版本控制的概念

很多使用者進行**版本控制**（Version Control）的方法，是將檔案人工複製到另一個資料夾，但由於缺乏系統性的管理方式，久而久之，這類傳統人工的方式便容易產生問題，例如：資料尋找不易、在複製時不慎覆蓋同檔名的檔案等等。當多人共創時，管理的難度就會更高。在開發過程中，經常會發生下列幾種情形：

- 檔案被別人（或自己）覆蓋或是遭到誤刪
- 想知道成員修改了檔案的哪些內容
- 想要恢復之前修改的版本

而版本控制即是協助使用者進行檔案管理的工具，能夠隨時記錄檔案的變更時間及內容，並且讓使用者日後需要資料時能夠取回特定版本的檔案。

一般來說，版本控制軟體可達成以下功能：

- 多人同時修改同一份檔案時，不會覆蓋或是刪除別人的檔案
- 保留所有修改的歷程，可以很容易找到變更者，以及更改的歷史紀錄
- 隨時復原錯誤，可以將檔案恢復到任何時期的版本

執行專案時，應選用具有版本控制的協同共創工具。例如：使用 Google 雲端硬碟的程式建立的檔案文件，可以多人同時線上編輯，使文件保持最新的狀態；同一份 Google 文件，可以支援 100 個人同時線上共同編輯。

IT 補給站

Android 版本與命名

Android作業系統有預發行的內部版本，分別為原子小金剛（Astro）與機器人班亭（Bender，電視動畫《飛出個未來》的角色）。從2009年5月開始，Android的版本代號改以甜點來命名，且每個代號間的字首以英文字母序接續排列：Cupcake（杯子蛋糕）、Donut（甜甜圈）、Eclair（閃電泡芙）、Froyo（優格冰淇淋）、Gingerbread（薑餅）、Honeycomb（蜂巢）、Ice Cream Sandwich（冰淇淋三明治）、Jelly Bean（雷根糖）、KitKat（奇巧巧克力）、Lollipop（棒棒糖）、Marshmallow（棉花糖）、Nougat（牛軋糖）、Oreo（奧利奧）、Pie（派）。2019年8月23日，Google宣布從Android Q開始不再以甜品命名，且直接稱 Android Q 為Android 10 。

名稱	版本號	發行日期	API 等級
Android 1.0	1.0	2008年9月23日	1
⋮			
Android Lollipop	5.0 – 5.1.1	2014年11月12日	21 – 22
Android Marshmallow	6.0 – 6.0.1	2015年10月5日	23
Android Nougat	7.0 – 7.1.2	2016年8月22日	24 – 25
Android Oreo	8.0 – 8.1	2017年8月21日	26 – 27
Android Pie	9	2018年8月6日	28
Android 10	10	2019年9月3日	29
Android 11	11	2020年2月19日	30

科技即戰力

小論文版本控制實作

讓我們延續11-1節的「小論文進度之專案管理」活動：當我們最後要和組員共同製作一份小論文成果報告時，若不小心覆蓋到其他同學在Google Drive上傳的相同檔名（包含副檔名）的檔案資料，或是想把檔案還原到之前的版本時，都可以透過「管理版本」的功能來下載舊版本、上傳新版本，或是永久保存重要的版本。

小論文版本控制實作1 以Google Drive為例

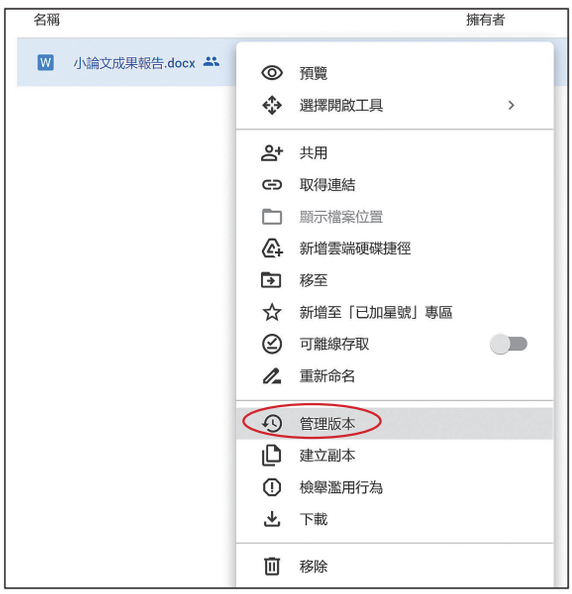


圖11-2-1 在 Google Drive 中，當需要還原小論文成果報告的舊版本時，可對目標檔案點擊右鍵，並在彈出的功能表中點選「管理版本」。

圖11-2-2 在該檔案管理版本的對話視窗中，顯示先前所有編輯過的檔案版本，並點選想要還原的目標版本之右下角處「⋮」選單。

圖11-2-3 選擇下載舊的版本檔案，並依提示重新上傳此檔案，即會覆蓋目前的檔案。如要避免特定檔案版本遭到刪除，也可以勾選「永久保存」，否則舊版本會在已儲存30天後，或已超過 100 個版本後遭到系統自動刪除。

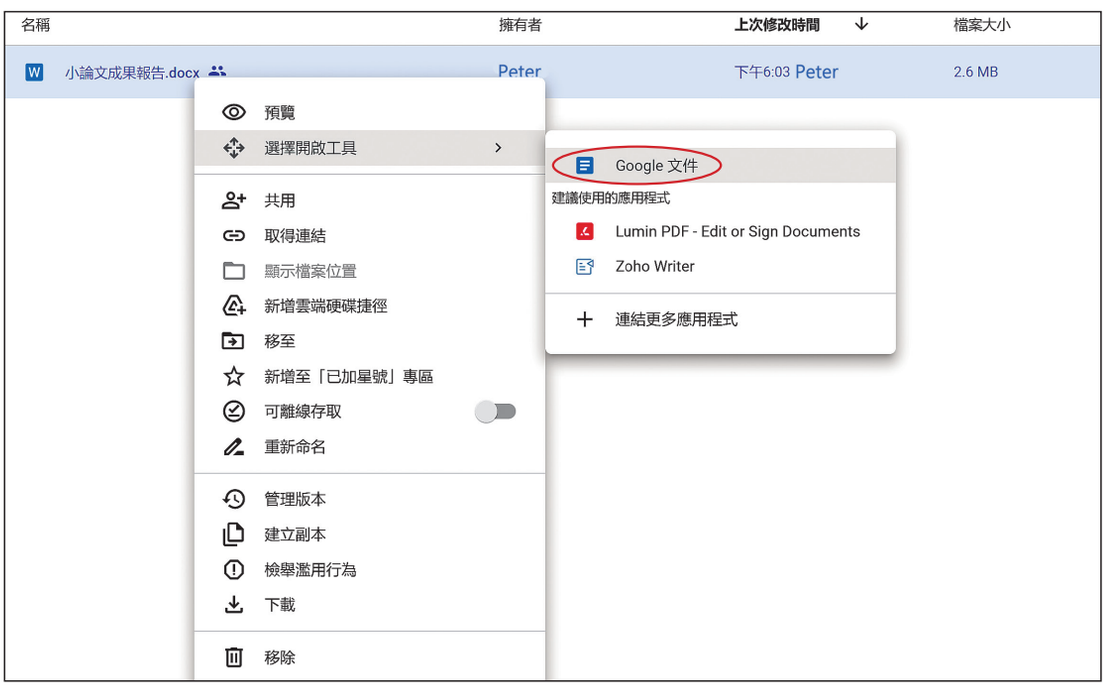
透過版本控管，在團隊的協作上，有以下的好處：

- 透過單一的檔案來源與團隊聯繫，避免版本問題
- 借助版本歷史紀錄，不會錯過文檔中的新增功能或更改內容
- 可以查看雲端硬碟檔案的變更情形，並追蹤對檔案進行變更的使用者

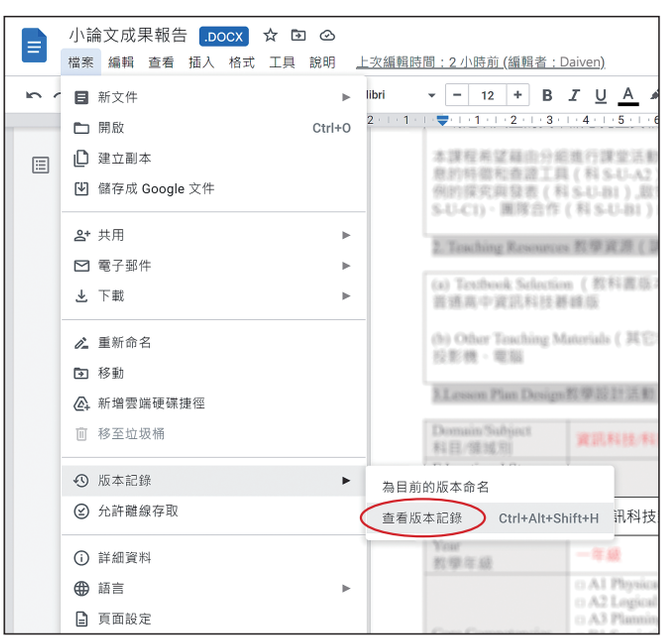
小論文版本控制實作2　以 Google Docs 為例

在Google Docs（Google文件）中編輯文件時，平台會自動儲存檔案與修訂紀錄，方便使用者進行修訂內容的控管、追蹤檔案與協同共創，如Google簡報、文件、試算表、表單、協作平台、Jamboard等，這些雲端應用程式的檔案都可以進行版本還原及追蹤修訂等功能。以下我們以Google文件控管小論文成果報告版本作為示範：

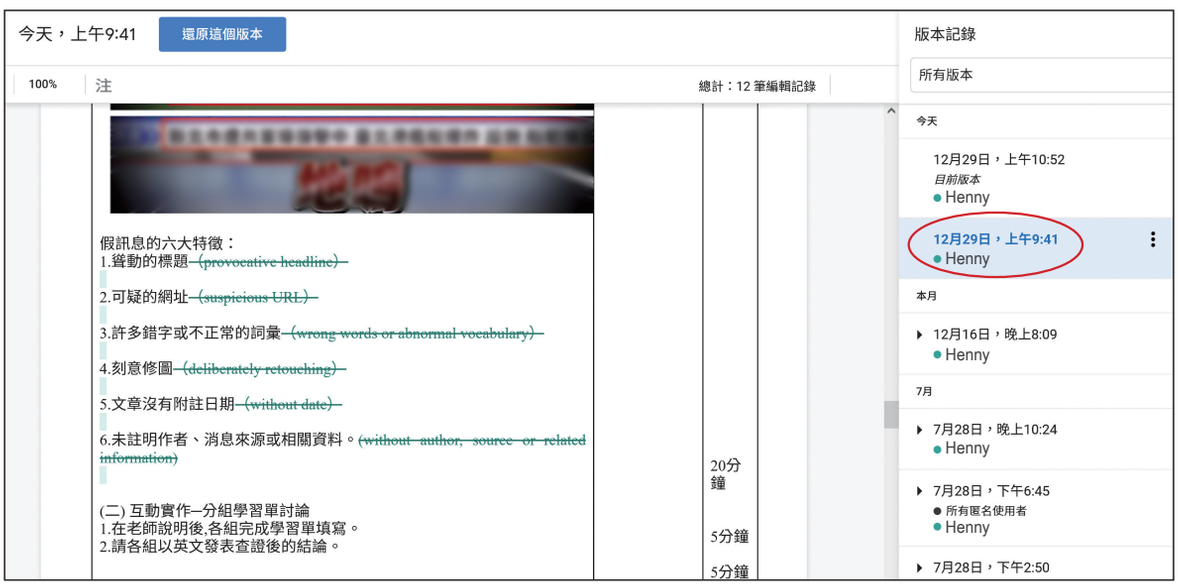
1.首先須打開小論文成果報告檔案。



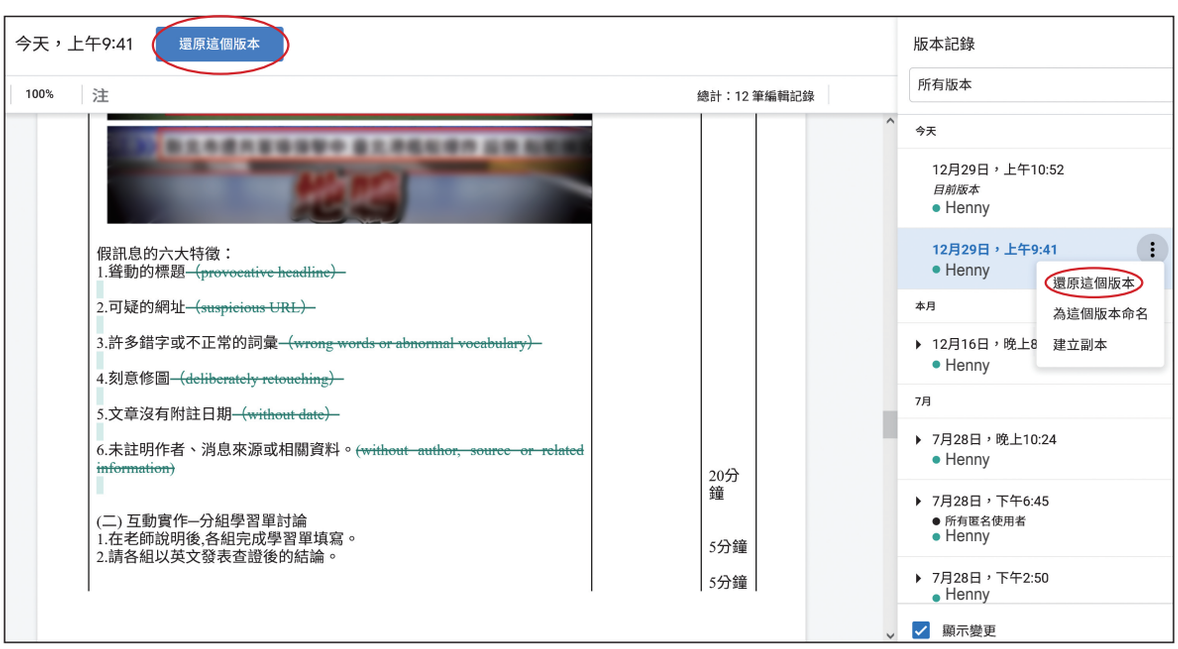
2.接下來點擊左上方的「檔案」選單，並於「版本記錄」中點選「查看版本記錄」。



3.可在畫面右側查看不同版本間的修改人員、修改時間，並透過左側文件預覽畫面上的文字顏色差異或是刪除線，來區隔不同版本的內容。



4.當我們想要還原的時候，可以點選想還原的目標版本，再點選上方的「還原這個版本」按鈕，或是點選右側目標版本的「⋮」選單，再點選「還原這個版本」。同時也可以在選單中替某個版本進行版本命名或是建立副本（再新增為另一檔案儲存）。



11-3 協同共創工具與實作

協同共創的過程中，應善用協同共創的數位工具，增進效率，提升品質。協同共創工具有很多種，如專案管理工具、文件、多媒體、程式設計等。網路上有許多免費的資源以及工具可以下載、安裝或線上使用，選擇合適的工具後，就可以開始進行專案中的各個流程。

11-3-1 協作共創的文件工具

共創的文件管理工具有很多種（表11-3-1），例如：Google、Microsoft 等公司不僅提供各項網路應用程式，讓使用者能與他人即時共同編輯文件，也能讓用戶建立共同資料夾，與他人分享資料。這些工具都具有版本控制功能，可作為協同共創的工具；不同工具各有功能上的特性，使用者可根據自身執行專案時的需求，選擇適合的工具。

表 11-3-1 Google、Microsoft 工具

Google 工具	Microsoft 工具	功能
Google 雲端硬碟	Microsoft OneDrive	提供資料儲存空間，可在線上編輯或分享。
Google 行事曆	Microsoft Outlook	可與他人共同訂定行程、會議時間，控管專案進度。
Google 的辦公室套件	Microsoft 365	可與他人共用文件、簡報、試算表，並一同編輯。
Google 表單	Microsoft Forms	可以協作建立、分析問卷調查，用以蒐集多筆資料。
Google 協作平台	Microsoft SharePoint	可與他人共同在線上設計網站。

IT 補給站

Google 提供的協同共創工具

Google 提供的協同共創工具可分成兩大類別：「協助專案上的溝通」以及「專案開發」。我們將進一步舉例，在一個由多人組成的小組專案當中，如何巧妙地利用這些工具來促進多人合作專案的協同共創，加速團隊效率。

圖11-3-1 Google 提供了專案上的「溝通」與「開發」兩種類別的協同共創工具

在以往的團隊協作中，我們或許會透過紙本的方式發放問卷，並手動把資料鍵入表格內，再生成報表，最後發送給團隊的其他成員。在這些工作流程中，我們切換了不少工具，也相對地耗費了更多的時間。讓我們想想看，如何改善此一流程呢？

圖11-3-2 紙本的問卷方式耗費較多時間

在學習過的工具裡，我們可以使用 Google 表單創建電子化的問卷，透過無紙化的方式，能更快速地蒐集資料；再將這些資料整理、匯入 Google Sheets，並直接在 Google Sheets 上生成報表文件；接著將文件連結放在 Gmail 上、分享給團隊的成員，藉此簡化整體的流程、省下大量的時間。若能將這些協同共創工具在實際專案上巧妙的運用，能讓日常工作變得智慧化，工作效率更高，團隊合作也能更加緊密。

圖11-3-3 運用Google提供的工具改善問卷流程的效率

而這些協作的工具，其實皆有使用到前面章節提到的「人工智慧」與「機器學習」所實作的功能。我們可以在使用這些工具的過程中，同時探索這些人工智慧與機器學習的功能。

Gmail
- 智慧預測撰寫
- 預測回覆內容
- 個性化垃圾郵件過濾
- 語法檢測
- 郵件沙箱

Google日曆
- 預測與會人員
- 就近會議室建議
- 會議狀況報告

雲端硬碟
- 智慧視圖（Priority View）
- 預測打開檔案建議
- 個性化搜尋結果
- 圖像辨識搜尋

Google Meet
- 主動降噪
- 會議字幕
- 會議字幕即時翻譯
- 畫面明暗自動調整
- 會議背景模糊／更換

Google Chat
- 預測回覆內容
- 遺漏訊息提醒
- 聊天機器人

Google Docs
- 多種語音轉文字
- 語法預測及檢查
- Explorer
- 問題自動撰寫（Form）
- 預測檔案分享用戶

圖11-3-4 在協作的工具裡也有人工智慧與機器學習實作的功能。

例如：在Gmail信箱回覆信件時，由人工智慧驅動的新功能——Smart Compose能幫助使用者更快速、輕鬆地從頭開始編寫或擬定郵件草稿。而這項功能是在後臺運行，因此使用者仍然可以用平時的方式編寫郵件，Smart Compose會在使用者輸入內容時，建議完整的句子。當使用者看到合適的建議時，可以點擊「Tab」選項來直接使用。目前這項實驗功能僅限於English(US) 模式下運行。

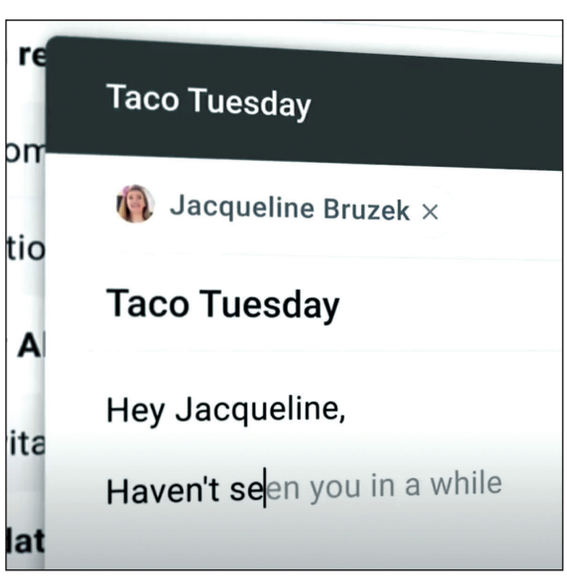


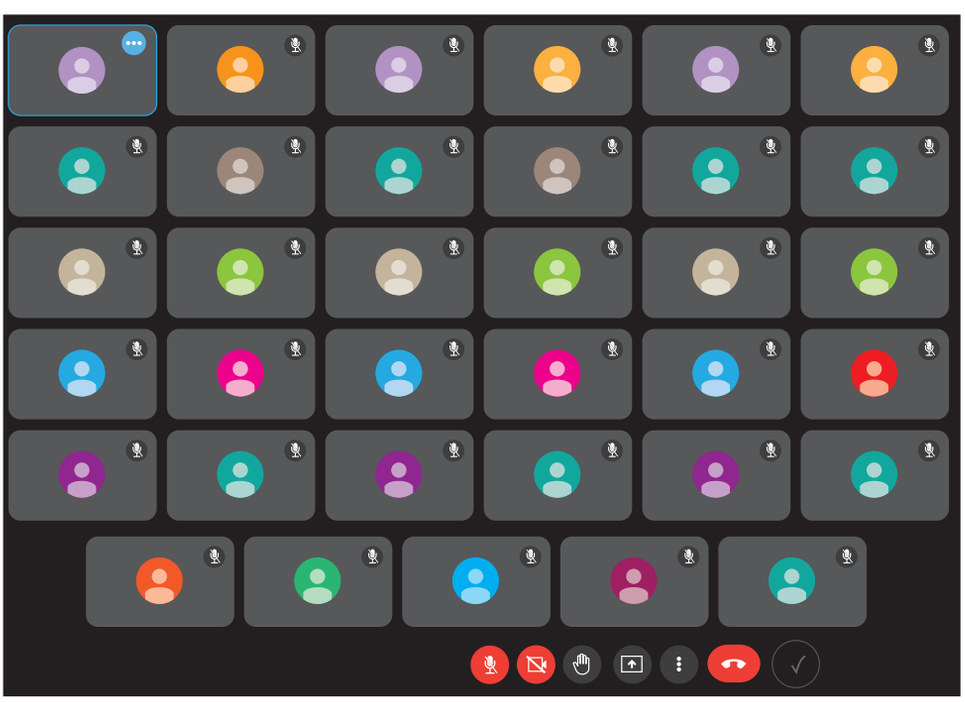
圖11-3-5 Smart Compose運用人工智慧建議合適的句子。

11-3-2 小論文成果報告製作

我們在這一小節將延續11-1、11-2中的小論文成果報告作為範例，以下是小論文專案過程中會使用到的協同共創工具。

	專案管理（時程、人員、流程、地點、資源等規劃）					
協同共創過程	1.擬定小論文題目	2.相關文獻探討	3.擬定／執行研究方法	4.數據蒐集與整理	5.研究分析與結果	6.製作學習歷程檔案
協同共創工具	Google Meet	Google 文件	Google 文件	Google 表單、試算表	Google 文件	Google 協作平台

1 運用 Google Meet 和組員線上討論小論文主題



2 進行小論文的相關文獻探討

文獻探討的功用主要是提供研究主題背後理論參考的依據，透過圖書、期刊以及電子資料等，了解問題的相關概念，並且協助研究者發展研究的架構、命題或假設。為了便於協同研讀與記錄文獻的摘要，可使用Google文件作為協同的工具，方便大家共同合作、撰寫文獻摘要，步驟如下：

STEP 01

進入「https://docs.google.com」，登入後新增一個文件。

STEP 02

撰寫檔案內容。

STEP 03

點選右上角的**共用**按鈕，將檔案分享給欲分享的共同成員，使之能一同編輯此份文件。

若是想要限定共用之對象，共用檔案連結可以控制檔案的共用權限範圍，有下列兩個選項：

限制

只有檔案擁有者指定的共用對象，才擁有使用檔案的權利。

獲得連結的使用者

只要取得檔案的分享連結，便擁有使用檔案的權利，且無須登入Google帳戶即可使用。

STEP 04

選取後可以選擇是否通知邀請對象、傳送訊息，以及設定其權限——檢視者／加註者／編輯者。

檢視者

僅有瀏覽這份檔案的權限。

加註者

可以在檔案上加入自己撰寫的建議和註解。

編輯者

為最高等級的身分，擁有與他人共用、編輯檔案，以及「接受／拒絕」檔案更動建議的權限。

研究方法的擬定與實施

同第2項的文獻探討工具，運用Google文件擬定與實施研究方法。

4 數據蒐集與整理

運用Google表單、試算表等工具，分別蒐集、整理資料（圖11-3-6～11-3-8）。

圖11-3-6 以Google表單進行問卷調查

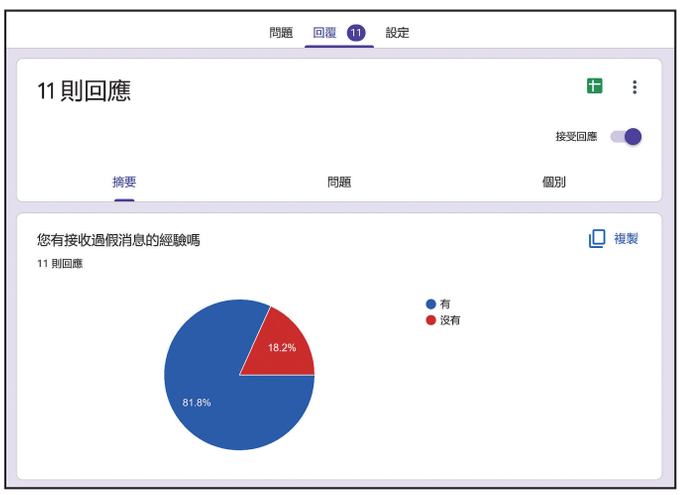
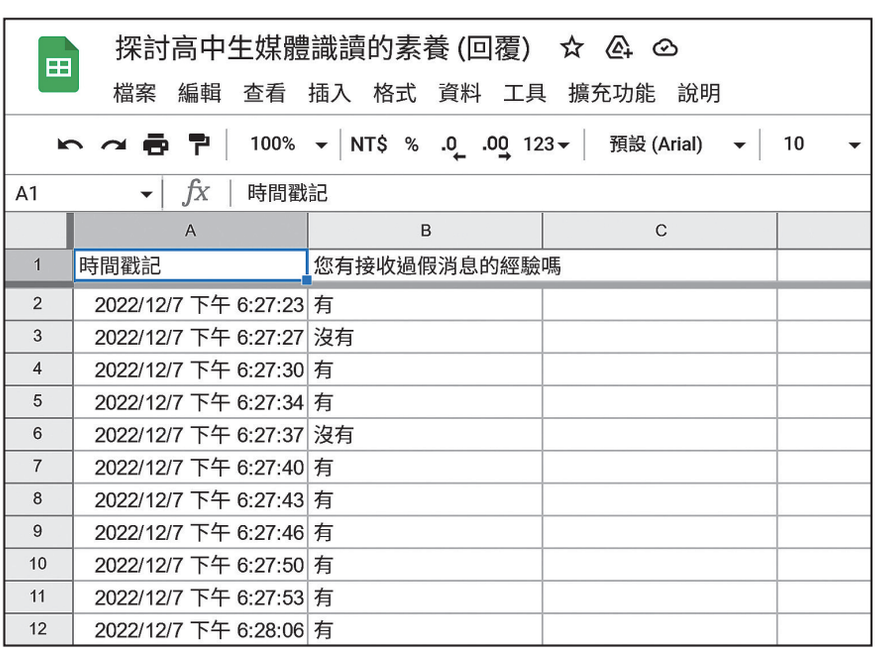


圖11-3-7 Google表單可以分別對每一問卷項目產出合適的資訊圖表



	A	B	C	
1	時間戳記	您有接收過假消息的經驗嗎		
2	2022/12/7 下午 6:27:23	有		
3	2022/12/7 下午 6:27:27	沒有		
4	2022/12/7 下午 6:27:30	有		
5	2022/12/7 下午 6:27:34	有		
6	2022/12/7 下午 6:27:37	沒有		
7	2022/12/7 下午 6:27:40	有		
8	2022/12/7 下午 6:27:43	有		
9	2022/12/7 下午 6:27:46	有		
10	2022/12/7 下午 6:27:50	有		
11	2022/12/7 下午 6:27:53	有		
12	2022/12/7 下午 6:28:06	有		

圖11-3-8 表單也可以轉換成試算表進行分析

5 將步驟4得到的資料進行研究分析與結果產出

同第2項的文獻探討工具，運用Google文件進行分析，並產出結論。

6 將上述1～5項的工作呈現（學習歷程）

若是想將成果以網頁的方式分享給群眾，則可以使用Google協作平台，步驟如下：

STEP 01

在 Google 雲端硬碟新增「Google 協作平台」

STEP 02

進入平台後，設定共用的對象，接下來就能共同編輯平台上的內容。

STEP 03

編輯完成後，可點選右上角發布的按鈕，設定發布的網址後，發布小組共同創作的協作平台。

科技即戰力

請同學進行分組（4～6人），運用本章所學習的專案管理與協同共創實作，製作屬於自己的科技小專題。

課堂檢測

( ) 1. 版本控制可解決下列何種情形？
(A)凱凱覺得修改後版本的研究方向錯誤，想要還原之前的版本
(B)詠心的期末報告檔案遭到誤刪，想要找回檔案
(C)勝敏期末分組報告的共同編輯內容不小心被組員覆蓋，他想要找回之前的版本
(D)以上皆是。

( ) 2. 下列何者不是版本控制的特色？
(A)當發現新內容有誤時，隨時復原錯誤
(B)無法找出每次更改的所有歷史紀錄
(C)可以多人同時修改同一份檔案內容
(D)所有內容修改的歷程皆會被記錄下來。

( ) 3. 聖潔想透過線上平台與同學共同編輯分組報告，下列何者並不具備共同編輯的功能？
(A)Google 文件 (B)Trello 平台
(C)Gmail (D)Microsoft 365。

( ) 4. 班長想創立一個班級共用的行事曆，使同學們可以記錄重要日期及活動。請問他可以選用下列何種工具？
(A)Google 行事曆 (B)Google 表單
(C)Power BI (D) Wufoo 表單。

( ) 5. 下列關於 Google 行事曆的功能敘述，何者為非？
(A)以時間軸記錄並分享每天自己的所在位置
(B)與同學共同訂定並分享寒假營隊行程
(C)與科展組員共同擬定會議時間，並提供視訊會議的連結
(D)老師控管科展小組的專案進度。

( ) 6. 琪琪想要撰寫有關物聯網技術的小論文，指導老師請他先進行相關主題的文獻探討。下列何者並非文獻探討的主要目的？
(A) 了解物聯網近期發展的趨勢與尚未解決的問題
(B) 了解物聯網的研究和不同領域的結合
(C)學習相關研究的研究設計與研究步驟
(D)提供物聯網技術背後理論參考的依據。

( ) 7. 曉華想要研究每年臺灣的用電情形，推估未來用電趨勢，下列何種方法不是蒐集資料的良好方式？
(A)擷取台電網站相關的公告數據
(B)搜尋政府的開放資料網站
(C)整理其他民間機構統計的資料
(D)每天手動記錄台電網站的即時用電數據。

( ) 8. 下列何種狀況最可能使用到甘特圖？
(A)公司財務報表分析 (B)解釋資訊系統架構
(C)研究計畫進度的控制 (D)產品發表會的呈現。

( ) 9. 下列何者較適合使用專案管理的方式進行？
(A)米米想規劃自己明天三餐的內容 (B)佑佑想和同學一起撰寫小論文
(C)宗宗想分析要買哪款智慧型手機 (D)小艾今天想致電向導師請假。

( ) 10. 今天剛到職的婷婷收到軟體開發專案團隊的訊息，通知她所負責的工作內容是撰寫程式。試問下列何者為婷婷所負責的軟體開發階段？
(A)系統實作與開發 (B)系統規劃與設計
(C)系統移交與維護 (D)系統整合與測試。

( ) 11. 甘特圖是屬於條狀圖的一種類型，其條形代表下列何種意義？
(A)工作項目所使用的軟體程式 (B)工作項目有變動的人數
(C)工作項目雲端化的程度 (D)工作項目開始和完成的時間。

( ) 12. 下列關於圖表製作的敘述，何者不正確？
(A)資料更容易理解 (B)沒有現成軟體可製作
(C)能更清楚比較資料 (D)便於觀看資料。

( ) 13. 下列何者並非專案管理的特性？
(A)可以把許多小型工作整合成一份大工作、同時執行，以節省時間和人力
(B)可以更彈性地配置資源，如人力資源等
(C)可以讓管理者更全面地檢視工作進度
(D)對於突發狀況或緊急事件的應變能力佳。

( ) 14. 昀傑擔任建造新半導體廠的專案經理，他想使用 Trello 線上平台進行新工廠建置的專案管理，請問 Trello 無法提供昀傑下列哪一項協助呢？
(A)了解專案的行程是否在時程進度的規劃內
(B)了解團隊成員各自投入在哪些工作上
(C)將整個專案拆分成許多可具體執行的工作
(D)線上即時支付某一工作的款項。

( ) 15. 宜臻想透過線上平台和同學共同完成小論文，以下分別是她想做的事情：（甲）需要與同學一起製作簡報、文件；（乙）需要透過問卷進行調查；（丙）需要有個網頁來公告目前團隊相關資訊。請問下列哪一選項對應到的平台組合，最能夠依序符合她的需求呢？
(A)Google協作平台、Google協作平台、Google協作平台
(B)Google協作平台、Google表單、Microsoft 365
(C)Microsoft 365、Google表單、Google協作平台
(D)Microsoft 365、Google協作平台、Google表單。

主題六

資訊科技與人類社會

chapter 12
資訊科技合理使用原則 288

chapter 13
個人資料的保護 302

chapter 14
資訊科技對人類社會的影響與衝擊 316

chapter

12

資訊科技合理使用原則

誠如偉大的科學家牛頓（Isaac Newton，1643～1727）所言：「如果我能看得比別人更遠，那是因為我站在巨人的肩膀上。」這代表著許多偉大的研究和發明，是可以循著前人的智慧結晶，再接續發展下去的。因此若過度保護智慧財產，則可能會導致人類及整體社會的發展淪於緩慢或停滯。

為了增進人類文化發展，促進社會持續進步，並維持創作者們的研發意願，要如何在保護原創者的著作權同時，也能讓其他創作者合理地運用其智慧結晶，則「合理使用」就是很重要的一環。完善的「合理使用」原則，有助於讓大眾在明確、合理的使用範圍內引用他人的成果，並且同時兼顧原創者的權益。

12-1 智慧財產權

智慧財產權包含**專利權**、**商標權**和**著作權**（圖12-1-1），有別於實體財產的所有權，智慧財產權的目的是為了保障研發者無形的智慧資產。

一般民眾可以透過使用者付費的機制享受他人創作研發後的智慧結晶，而所有權人也可以受到智慧財產權的相關法律，保護其成果，並以此鼓勵大眾透過研發，進而獲取收益。

圖12-1-1 智慧財產權包含了專利權、商標權、著作權等領域

專利權

包含**發明專利**、**新型專利**、**設計專利**等與創作、研發、設計相關的內容。其中發明專利在審核期間，政府會先將提出申請的專利內容與技術公開，並在這段期間給予其專利保護，其專利保護期限是自申請日起往後推算。

例如：藥商為取得某藥品的專利權，經過了5年的審核期，則剩下的專利權時效為15年。由於此藥品即使化學原料便宜，但藥商付出的研究時間和人事成本，都是讓藥品在專利權保護期間，價格居高不下的原因。一旦專利期屆滿，其他藥廠便得以運用公開的成分配方進行製造，此為所謂的「副廠藥」或稱「學名藥」。因此通常專利期過後，市場不會再被獨占壟斷，藥品價格也會因為競爭而下降。

補給站

專利權個案討論

大家是否納悶過：幾年前通訊軟體LINE「搖一搖」的加好友功能怎麼消失了呢？

當時LINE的這項功能可以讓距離近、有開啟GPS的使用者，藉由晃動手機便能搜尋到ID，互相加為好友。但在LINE被日本京都市的IT企業「future-i」控告侵害專利，經法院裁定侵權後，這項功能也在2020年被取消了。

LINE雖主張「此功能容易發明，因此專利應作廢」，但法官認為，「透過搖晃的動作就可以搜尋附近使用者的LINE ID」並不是一項容易辦到的發明，future-i的專利確實遭受侵害，因此判決LINE需要支付future-i逾1400萬日圓（當時約358萬元新臺幣）的侵權賠償金。

圖12-1-2 消費者可透過商標來辨識石油品牌

商標權

舉凡任何具有辨識性，且足以使消費者區分商品或服務來源的不同之標識。其形式可以文字、圖形、記號、顏色、立體形狀、動態、全像圖、聲音等所組成。商標和專利一樣，必須主動向政府申請並經審核註冊後才享有商標權。

例如：對於數家不同品牌的加油站，消費者可以透過招牌上的「商標」，迅速辨識出該家石油品牌公司（圖12-1-2）。商標權的期限自申請註冊日起往後推算10年，期滿前可提出申請「展延」再10年，展延次數不限。

IT補給站

商標權個案討論

連鎖咖啡店「星巴克」，為了保障自己的商標權，曾經對於有「抄襲」商標嫌疑的本土同業「壹咖啡」提告，最後法院做出判決，認為星巴克「戴皇冠的女性圖案」和壹咖啡「咖啡杯中冒蒸氣」的圖形並不相似，且eCOFFEE和STARBUCKS兩商標之英文字母數量、讀音皆不同，因而認定壹咖啡商標具極高設計性、獨創性，無致消費者混淆之疑慮，判星巴克敗訴。

著作權

著作權保護的範圍涵蓋文學、科學、藝術或其他學術範圍之創作。只要成果是具有原創性，皆享有著作權的保障：

1.著作完成當下即享有著作權保護，並可以向他人主張權利，不需另外向政府部門進行申請、註冊等程序，此為和其他智慧財產權相異之處。

2.著作財產權包括：重製權、改作權、編輯權、出租權、散布權、公開播送、公開傳輸、公開口述、公開上映、公開演出、公開展示的權利。著作財產權得以轉讓或繼承。而著作財產權存續於著作人之生存期間及其死亡後50年。若著作於著作人死亡後40年至50年間首次公開發表者，著作財產權之期間自公開發表時起存續10年。若為共同著作之著作財產權，則存續至最後死亡之著作人死亡後50年。

3.著作人在著作完成時即享有著作人格權，不得繼承或轉讓，其權利如下：
⑴公開發表：著作權人可依自己的意願決定是否公開著作。
⑵姓名表示：著作權人可以決定是否具名，得以使用別名或真名標示。
⑶禁止改作：禁止他人將本公開著作進行不當修改的權利。

4.若著作存在委託人與受委託人互為雇傭關係時，著作財產權得以依契約另行約定其權利歸屬，而著作人格權是在著作完成時，依附在著作財產權人上，雖依法不得轉移，但在著作尚未完成即約定著作財產權為委託人時，著作人格權亦可為委託人所有。

科技即戰力

請同學們討論看看，下列各項行為是否正確、合法？

1.在Instagram上分享自己的照片，並結合具有版權的音樂播放，把帳號設定為不公開，因此音樂不須經過著作權人同意。
2.電腦軟體的著作權人死亡後，就可以當成自由軟體使用了。
3.社群媒體上面分享的國家公園風景照片是屬於公共環境，因此照片不受著作權保護。
4.參訪美術館展覽國外作品時，因為來自國外，故可任意翻拍作品。
5.可以將家裡面的「家用版」影片帶到學校公開播放、分享給同學欣賞。
6.學生想修改網路上的投影片檔案作為期末報告，由於是教育用途，因此不須徵得該簡報之著作財產權人的授權。
7.可以和同學合購網路串流播放平台的帳號使用權。
8.將合法購買的電影DVD進行複製備份，避免DVD受潮損壞。

IT 補給站

無所不在的智慧財產權

在大家的求學過程中，其實也涉及了許多關於智慧財產權的保護問題。舉例來說，某大學教授在進行化學實驗時，合成了一種可以滅菌的化合物，並且可作為製作口罩的材料，教授除了可在期刊上發表自己的學術研究，也可以向政府申請「發明專利權」；而教授在課堂上使用的授課投影片以及學生的書面報告則歸類為「著作」，在創作完成當下即受《著作權法》保護；教室中常見的版權軟體，如「Microsoft」、「Google」的圖像與文字標示，則是受到《商標法》保障的「商標權」。

Registered、Copyright

通常在商標圖像與文字標示的右上方。若有R的字樣，表示這個商標已合法註冊完成（Registered），而標示C（Copyright）則為版權所有的意思，表示受到《著作權法》保護。

著作權合理之使用要件

我們可以針對「**著作財產權**」進行合理的使用，但「**著作人格權**」無法引用，倘若著作財產權人授權給予使用其著作，則非著作權人的引用方就不須另行聲明合理使用其著作。著作權的合理使用是指，著作權以外之人，對於著作權人依法享有之專有權利，縱使未經著作權人同意或授權，仍得在合理範圍內，以合理方法，自由且無償加以利用之主張，在我國《著作權法》第44條至65條訂有規範。如教育、非營利等用途等。

而《著作權法》第65條中的相關規定，可以作為是否合理使用的判斷基準（圖12-1-3）：

一、利用之目的及性質，包括係為商業目的或非營利教育目的。
二、著作之性質是為了公益或是個人而創作。
三、所利用之質與量及其在整個著作所占之比例。
四、利用結果對著作潛在市場與現在價值之影響，是否影響現有市場的利益。

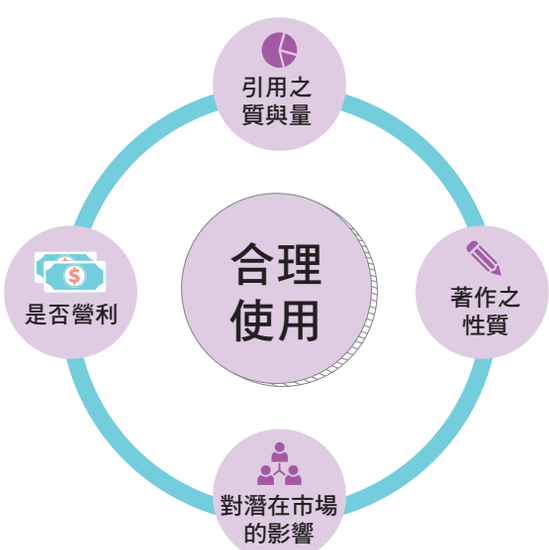


圖12-1-3 著作權的合理使用原則要素

IT 補給站

1.某知名YouTuber因製作「五分鐘看完電影系列」影片而爆紅，但因製作內容是節錄電影部分片段再加上自己對劇情見解的旁白，2018年遭檢方依違反著作權法起訴。該名YouTuber雖提出自己作品是屬於「二次創作」，但最終仍與五家受侵權的片商進行和解。

2.大學上課用的原文書一本通常都約千元以上，使得大學生常為省錢，請附近的影印店影印整本原版書；店家也因為有利可圖，甘冒違反著作權的風險。而檢調為杜絕影印店此種非法風氣，以往只追究業者違法，近年也開始追究學生的法律責任。

3.英國攝影師史拉特（David J Slater）之前到印尼蘇拉威島進行生態攝影、架好相機時，不料有一隻大膽的黑冠獼猴上前玩弄、按下快門，在鏡頭前露齒傻笑自拍，意外成為舉世聞名的猴子自拍照（圖12-1-4）。照片一出立即引來全球注目，有人將它上傳到維基共享資源，在史拉特主張著作權之下，維基第一次將照片刪除。但是當照片再一次被上傳到維基共享時，維基主張照片的原作者是猴子，而猴子並不適用於人類的著作權法，因此拒絕將照片刪除。此事引起攝影師史特拉與維基的版權之爭，鬧上美國著作權局。

而美國著作權局做出解釋：不管是動物、植物，或者是神靈、鬼怪產生的相關創作作品，都不能註冊著作權。也意味著維基共享資源的主張沒錯，讓這段版權爭議暫時落幕。

圖12-1-4 舉世聞名的黑冠獼猴自拍照

科技即戰力

請以著作權合理使用的四項原則，分析前述案例中的知名YouTuber「五分鐘看完電影系列」的二次創作是否合理。

12-2 創用 CC（Creative Commons）與軟體授權

12-2-1 創用CC的概念

在《著作權法》中，多數的條文是傾向保護著作權人。縱使我們在網路世界裡可以獲取多數想了解的資訊，但只要沒得到原創者的同意便擅自引用，即有可能造成侵權。

因此，為了能增加著作利用人合理引用並創作的意願，我國中央研究院資訊科學研究所導入美國的**創用 CC** 授權模式。雖然其並非為法律位階的條文，但透過著作權人主動選擇的授權組合來釋出自己的創作，更能促進知識的開放與延伸性。臺灣將之命名為「**創用 CC 授權條款**」，強調其授權模式出於著作的「**創**」作與使「**用**」。

創用CC授權有四個核心授權要素：

1 **姓名標示**（Attribution）
利用人必須按照作者或授權人所指定的方式，標示其名稱或筆名。「指定的方式」是指若授權人或作者另有指定方式，例如：僅要求標示贊助機構，則必須依其指定方式標示之。

2 **非商業性**（Noncommercial）
不得以獲取商業利益或私人金錢報酬為主要目的來利用作品。例如：分享他人在社群媒體所發表的文章，因為沒有商業上或金錢上的往來，因此符合非商業性。

3 **禁止改作**（No Derivatives）
利用人不得更改原著作內容，僅可重製作品。
利用人不能改變授權人的作品。常見的改作如翻譯本、詞曲、劇本、小說、電影等作品。

4 **相同方式分享**（Share Alike）
利用人將原著作變更或修改後所衍生的作品，必須以原著作相同或類似的創用CC授權條款繼續分享該衍生作品。若並未修改原著作內容，而是以重製、散布的方式分享原著作，無論原作者有無「相同方式分享」的標示，仍必須以原著作的創用CC授權為主。

上述四項創用CC的核心要素，可以組合成下面六種主要的授權條款，而在這六種條款組合中，每種都具有「姓名標示」的要素，可以將這些標章標示於作品頁面上，向所有人聲明作品的聲明模式（圖12-2-1）：

姓名標示	姓名標示-禁止改作	姓名標示-相同方式分享
BY	BY ND	BY SA
姓名標示-非商業性	**姓名標示-非商業性-禁止改作**	**姓名標示-非商業性-相同方式分享**
BY NC	BY NC ND	BY NC SA

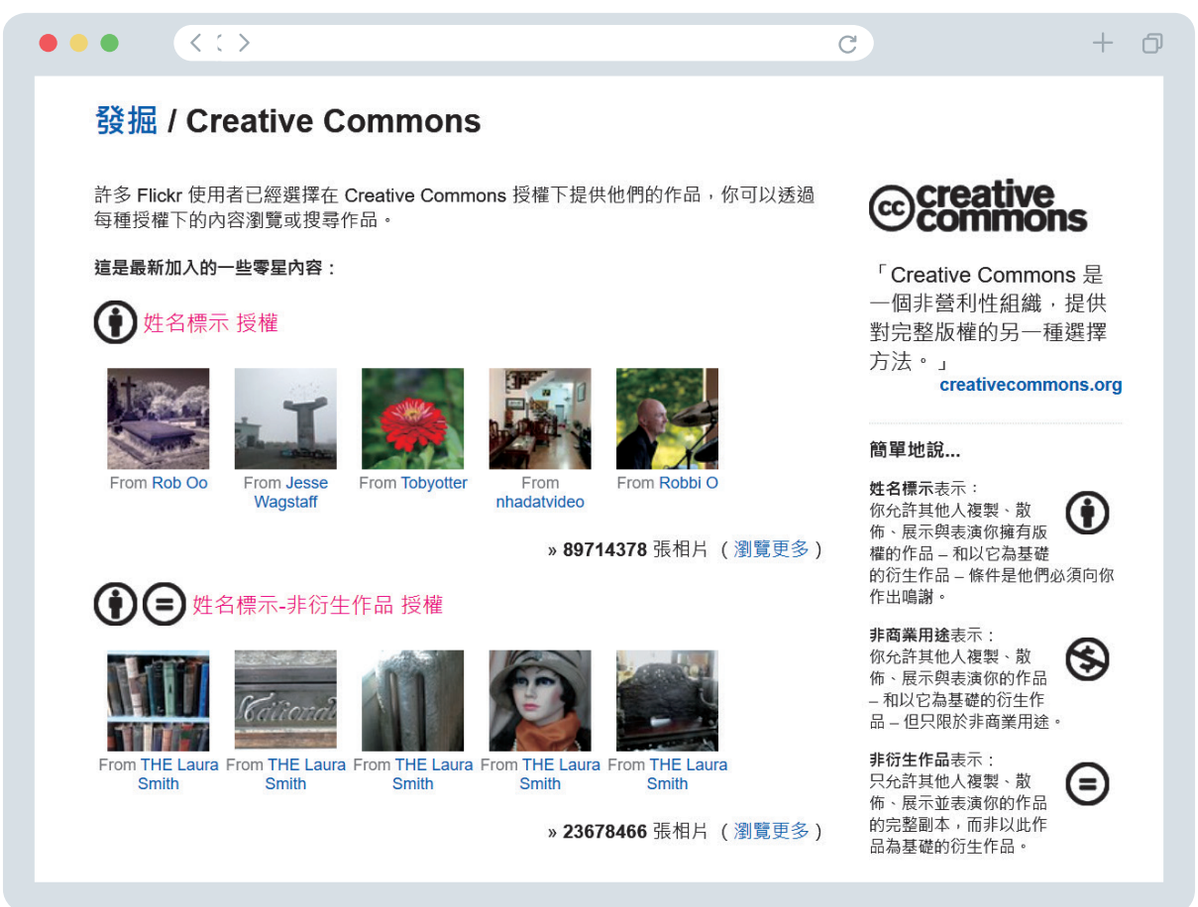


圖12-2-1 flickr相片平台以創用CC授權分類作品

科技即戰力

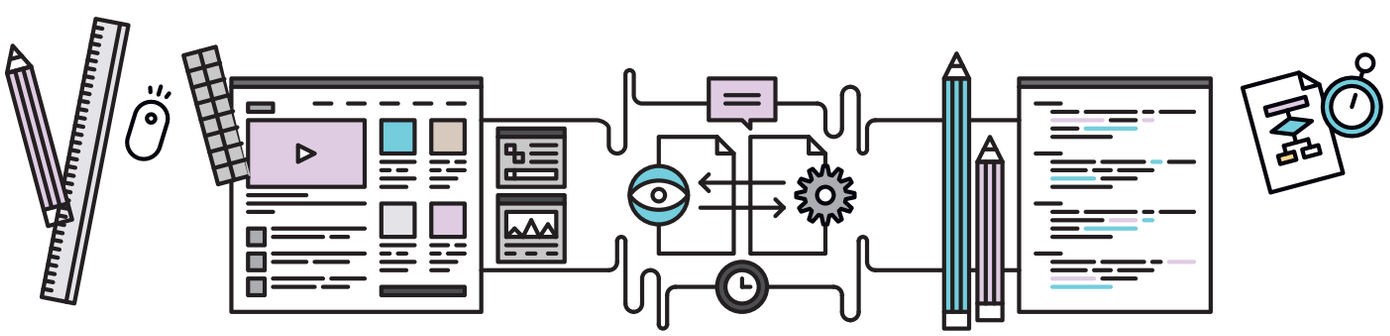
CC0「公眾領域貢獻宣告」

有些作品若有標示「CC0」，代表著作權人已拋棄他們對自己著作的各項權利，其中多數人的目的是為了使更多創作者能運用這些素材進行更多的創作、提升或再使用等行為，而不受著作權等相關法律的限制。創用CC和CC0授權的差別在於，創用CC允許著作權人在一定範圍內保留部分權利；而CC0則提供「**不保留權利**」的授權選擇。

除了上述flickr網站平台外，同學若要製作專題或是報告時，還可以去哪些平台獲取標註CC0的資源呢？（例如：下表可以找到部分可合法取用的音樂、圖片／照片、影片／動畫、文獻等相關資源）

1.除了下表的網站平台外，請再列出三個提供關於「CC0」的創作資源平台。
2.並請嘗試將三項作品及其CC0宣告，透過擷取檔案的方式和同學分享。

檔案類型	網站平台
音樂	unminus：https://www.unminus.com/ bensound：https://www.bensound.com/ dig.ccmixter：http://dig.ccmixter.org/
圖片／照片	CC0 免費圖庫搜尋引擎：https://cc0.wfublog.com/ FOCA Stock：https://focastock.com/ LibreStock：https://librestock.com/
影片／動畫	pixabay：https://pixabay.com/videos/ Distill：http://www.wedistill.io/ mixkit：https://mixkit.co/
文獻資源	典藏臺灣：https://digitalarchives.tw/



12-2-2 軟體授權的類型

除了以創用CC的方式分享的各類型數位作品外，程式設計後所伴隨的軟體授權也包含了許多類型。例如：我們熟知的Microsoft的Windows作業系統即是一種**專利軟體**，必須先付費後才能取得軟體的使用權，且使用者不能對該軟體進行任何修改、複製、散布等行為；而購買同一軟體的不同版本，也有不同的授權限制（表12-2-1），市面上販售的軟體通常分為**OEM隨機版**、**一般家用版**、**大量授權**等。

表 12-2-1 軟體授權之使用對象與限制

	使用對象	限制
OEM隨機版	一般使用者，通常作為購買套裝電腦的選配軟體	不能將軟體轉換到另一臺電腦使用
一般家用版	一般使用者（零售）	可以將軟體轉換到另一臺電腦使用，但一個序號在同時間仍以一臺電腦使用為限
大量授權	校園、企業、政府部門	大量授權所購買的套數，或是在該組織內同一時間的連線數

相對於須付費的專利軟體，其實仍有許多可免費使用的軟體，如**公共財軟體**（Public Domain Software）、**免費軟體**（Freeware）、**共享軟體**（Shareware）、**自由軟體**（Free Software）等。雖然多數皆為免付費即可使用，但仍有使用權限的差異性。

公共財軟體

免費且無著作權的軟體。當程式著作已逾著作權保護期間，或是著作權擁有者宣告放棄此程式的著作權，則該軟體將被視為不屬於私人專屬財產或利益，而屬於全體人類社會公共所有的技術、資訊、知識等。

免費軟體

使用者無須付費即可自行下載、安裝、使用的軟體。雖然軟體免費，但大多並沒有開放其原始碼，因此使用者並沒有複製、研究、修改和散布的權利，且通常都會有其他的限制和使用不便之處，如功能限制、廣告播放等。通常這類軟體的開發廠商是先使用免費的方式，達到一定的市占率後，再開始透過廣告或付費的加值應用進行商業獲利。如Adobe PDF、Whoscall等軟體。

共享軟體

軟體會給予使用者一段試用期間，時間一到即結束該軟體的合法授權，除了付費以外，不得使用其他任何非正規的方式解除其限制。

自由軟體

根據GNU GPL（GNU General Public License，GNU通用公眾授權條款）的規範，自由軟體其實是奠基於學術研究上的構想和理念，其強調的是軟體的開放性，如使用、研究、散布、改良的自由，但不一定免費。其開放原始碼著重在軟體修改、研究的開放性，可集結眾人的智慧不斷將該軟體修正演進。自由軟體在發行的時候，除提供該軟體的原始碼，並授權允許使用者使用、修改、散布著作，透過更多人共同參與該軟體的設計，可以積極改善軟體的缺失和漏洞。

因此，自由軟體除了提倡知識共享的理念，同時也鼓勵人們從事研發工作作為獲利的方式，進而保護程式設計師進行開創知識經濟的動機。

表 12-2-2 各類軟體的授權

軟體類型／授權項目	保留著作權	免費使用	開放原始碼	相關軟體
專利軟體	V			Window 10 Mac OS
公共財軟體		V		SQLite（資料庫軟體） FreeCAD（3D製圖）
免費軟體	V	V		報稅軟體 Instagram
共享軟體	V	試用期間免費		WinRAR CamScanner
自由軟體	V	是否免費由開發者決定	V	LibreOffice Android Open Office

表 12-2-3 各用途之專利軟體及其可替代的自由軟體對照表

軟體用途／軟體類型	專利軟體	自由軟體
向量繪圖軟體	Adobe Illustrator	Inkscape
點陣圖繪圖軟體	Adobe Photoshop	Krita、Gimp
書刊編輯軟體	Adobe InDesign	Scribus
3D 繪圖軟體	3ds max、Maya、SketchUp	Blender
影片剪輯軟體	威力導演、繪聲繪影	Kdenlive
辦公室軟體	Microsoft 365	LibreOffice
壓縮軟體	WinRAR	7-Zip

科技即戰力

尋找合適的自由軟體

教育部曾推動的「校園自由軟體數位資源服務計畫」中，將自由軟體分為下列六大軟體類型。同學們可以到各大學的自由軟體專區（如國立臺灣師範大學自由軟體介紹網站：www.itc.ntnu.edu.tw/index.php/free_software/），分別填入兩種感興趣的軟體，並嘗試下載使用後，與同學分享軟體的操作心得。

軟體類型	軟體名稱
辦公	例如：心智圖軟體（XMind）、PDF 編輯器（PDF-Shuffler）
美工	
影音	
網路	
程式	
教育	

課堂檢測

（　　）1. 英國柏納李（Tim Berners-Lee）是全球資訊網（www）的發明者，他認為人類所有的成就都是互動的結果，應彼此分享。因此，為了讓每個人都有機會利用網路，他堅持不申請下列何項權利，促成今日網路資訊的蓬勃發展？
(A)著作權　(B)商標權　(C)專利權　(D)人格權。

（　　）2.（甲）知名網紅打著二次創作的名義，製作「X分鐘看電影」短片，許多國際電影公司跨海來臺指控其侵權；（乙）「周董雞排」、「成龍數學」等運用藝人藝名作為招牌的行為，經紀公司表示恐有侵權疑慮；（丙）生技大廠莫德納公司控告輝瑞公司盜用mRNA技術開發COVID-19疫苗。上述三件案例，侵犯的智慧財產權依序為何？
(A)著作權、商標權、商標權　(B)著作權、專利權、專利權
(C)著作權、商標權、專利權　(D)專利權、商標權、著作權。

（　　）3. 某大學圖書館館員將採購且具有版權保護的音樂CD轉錄成MP3，分送同仁欣賞。下列敘述，何者正確？
(A)只要不公開播放即不算違法　(B)此舉觸犯著作權法
(C)只要不提供使用者下載即不算違法　(D)轉錄為MP3屬於合法的重製行為。

（　　）4. 為了促進文明發展與社會進步，各國政府莫不致力於智慧財產權的保護，有關智財權的敘述，下列何者正確？
(A)先申請先贏，是著作權的特性
(B)作品完成就擁有，是專利權的特性
(C)著作人格權是經濟權利，可繼承亦可轉讓
(D)商標權代表廠商的信譽，具有經濟利益。

（　　）5. 宗升是受雇於某公司的程式設計師，請問他替公司設計的軟體程式碼，在雙方無特別約定的前提下，其程式碼的著作權歸屬何者正確？
(A)著作人為宗升，著作財產權屬於公司　(B)著作人與著作財產權皆屬於宗升
(C)著作人為公司，著作財產權屬於宗升　(D)著作人與著作財產權皆屬於公司。

（　　）6. 云云下列的生活習慣中，何者為侵害商標權的行為？
(A)有事沒事就請友人從國外代為購買LV的仿冒特A貨回國販售
(B)無聊的時候把愛情小說的章節抄寫放置到自己的部落格，並署上自己的姓名
(C)未經合法授權免費下載藝人整張專輯
(D)將麥當勞招牌上的M字樣噴漆毀損。

（　　）7. 我國著作權法所規範的「合理使用」（Fair Use）指的是：
(A)凡使用著作權法所保護之資料，必須徵求著作權人的同意才可使用
(B)允許他人在一定的規範下合理使用該著作，而不需徵求著作權人的同意
(C)使用著作權法所保護之資料，必須徵求著作權人的同意，且給付合理的版權費
(D)使用著作權法所保護之資料，若未事先徵求著作權人的同意，事後需繳交雙倍版權費。

( ) 8. 阿館發現某健身中心未經許可，將他自己繪製的原創角色圖片登記作為商標，請問阿館可以主張該健身中心侵犯他的哪一項權利？
(A)商標權 (B)肖像權 (C)著作權 (D)專利權。

( ) 9. 小柯想剪輯短片作為期末報告，但必須使用到一些圖片、音樂等素材。請問下列何者不是合法取得素材的方式？
(A)透過 E-mail 詢問原作者是否可以授權使用
(B)直接在圖片下方留言區詢問是否可以授權使用
(C)運用修圖軟體進行二次創作後再引用至短片
(D)到網路上尋找可供授權使用的免費圖庫。

( ) 10. 小寧在網路購物平台上購買了一臺筆記型電腦，請問其中的微軟作業系統可能是下列何種版本呢？
(A) OEM 隨機版 (B)一般家用版 (C)大量授權版 (D)個人版。

( ) 11. 下列何種情形未侵害著作權或製版權？
(A)擁有著作財產權者已死亡，任意人可以重製銷售
(B)免費軟體可以提供個人免費使用
(C)共享軟體可以拿來任意轉拷銷售
(D)只購買一套軟體即可以用來安裝多臺電腦。

( ) 12. 近年政府極力宣導尊重他人智慧財產權，雖不容許智慧財產權遭到侵犯，但是也允許合理的使用。下列何種行為是屬於著作權的合理使用？
(A)迪希下載免費軟體（Freeware）供自己使用
(B)柏毅將其閱讀過之優美的文章，貼在自己的部落格上，並未引述文章出處
(C)拉拉定期以電子報方式大量轉寄別人在部落格上發表的文章
(D)夢夢把從別人部落格上收集來的文章整理出版。

( ) 13. 下列關於自由軟體（Free Software）的敘述何者有誤？
(A)根據 GNU 組織的 GPL 條款發行
(B)公開原始碼
(C)享有著作權保護，允許範圍外的重製、散布，但修改行為是違法的
(D)允許使用者重製或修改。

( ) 14. 下列何者是最符合「著作權合理使用」的正確行為？
(A)基於公益性質，引用某著作的大量資料
(B)基於公益性質，引用某著作的精華資料
(C)基於公益性質，引用少量著作的非精華資料
(D)修改他人著作後投稿雜誌，賺取自己的學費。

( ) 15. 大明撰寫了一本電腦程式相關的書籍，請問下列行為何者合法？
(A)大明可讓與姓名權給其他知名的電腦程式設計師
(B)大明的女兒日後可以繼承其著作人格權
(C)大明可以用別名發表這本著作
(D)大明可以將其著作人格權轉讓給其他人。

chapter

13

個人資料的保護

「隱私」指的是關於自身，且不希望被公開的資訊。而人們都有權利去保護自己的隱私不被公開。隱私的重要性在於其為影響個人自尊、主體性與人格發展的重要因素。我們通常都有過自身的一些訊息在未經同意的情況下被公開的經驗，如成績、地址等等，當下通常伴隨著不舒服甚至是羞怯的情緒反應。再者若是因為隱私的外洩而遭到他人騷擾，容易讓人產生精神壓力、退縮、畏懼的負面心理。

而隱私權設立的宗旨即是為了保障個人生活私密領域免於受到他人侵擾，以及保障個人資料之自主控制。其中個人資料的自主控制對於資訊隱私權來說，就是保障自身是否揭露個人資料的權利，以及在何種範圍內、於何時、以何種方式、向何人揭露之決定權，並保障人民對其個人資料之使用有知悉與控制權，以及資料記載錯誤之更正權。

13-1 個人資料保護法

1995年，政府公布了《**個人資料保護法**》的前身——《**電腦處理個人資料保護法**》，但此法僅限於透過電腦處理的個人資料，且實施對象只針對八類行業、十三項特定事業。

而隨著民眾重視個人隱私的程度愈來愈高，2012年正式實施《個人資料保護法》各項細則，無論是紙本或是電腦處理的資料皆能受到《個人資料保護法》的保障，實施對象為所有**自然人**和**法人**，且本法的修訂參照了國際上的個資保護原則，因此保障的範圍更為全面。

《個人資料保護法》的立法精神在於讓個人資料的蒐集有所規範和依據，其目的在於保障我們的**人格權**與**隱私權**，避免遭受他人的侵害。並在此前提下，促進個人資料合法、合理地使用，增加個人對自身個資的掌控與決定權，而非限制個資的使用與蒐集。在《個人資料保護法》中有定義個人資料的種類，如圖13-1-1：

圖13-1-1 《個人資料保護法》所定義的個人資料種類

《個人資料保護法》將所規範的對象分為公務機關與非公務機關（包含個人），前者指依法行使公權力之中央、地方機關或行政法人；後者指前述公務機關以外之自然人、法人或其他團體。《個人資料保護法》對於前述兩對象所蒐集到的個資有相關行為規範。「行為」是指個資的蒐集、處理和利用（表13-1-1），在《個人資料保護法》第5條中，明訂蒐集、處理或利用個人資料時，必須遵循的四個原則如下：尊重當事人的權益、採取誠信及信用的方法、不得逾越特定目的之必要範圍、應與蒐集目的具有正當合理的關聯（圖13-1-2）。

表 13-1-1 《個人資料保護法》規範的行為

蒐集	以任何方式取得個人資料
處理	建立或利用個人資料檔案所作的資料記錄、輸入、儲存、編輯、更正、複製、檢索、刪除、輸出、連結或內部傳送
利用	將蒐集的個人資料作處理以外的利用

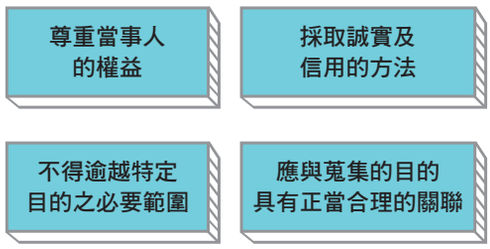


圖13-1-2 個資合理的使用原則

公務機關與非公務機關可依據《個人資料保護法》實行下列11項安全維護措施（表13-1-2）：

表 13-1-2 《個人資料保護法》的11項安全維護措施

安全維護措施	實施重點
1.配置管理的人員及相當資源	指派專人或成立個資因應小組，有專人負責推動個資因應措施，並有適當的資源來支持。
2.界定個人資料的範圍	盤點內部所有的個人資料，確定該保留的個資、定義個資管理的權責。
3.個人資料的風險評估與管理機制	評估個人資料的價值與風險，依據風險擬定個資管理機制與應變措施。
4.事故的預防、通報及應變機制	依個資的風險建立事故預防、通報及應變機制，確保可提早預防個資侵害事件，或於侵害事件發生時可立即因應，降低損害。
5.個人資料蒐集、處理及利用的內部管理程序	依個人資料的蒐集、處理或利用的規定，擬定內部作業管理流程。
6.資料安全管理及人員管理	擬定資料分級、安全管理機制，以及管理人員存取資料的權限。
7.認知宣導及教育訓練	宣導《個人資料保護法》，提升個資意識，並定期舉行教育訓練。
8.設備安全管理	針對資訊設備或與個資處理有關的設備，擬定管理制度，落實管控措施。
9.資料安全稽核機制	擬定資料安全稽核機制，落實個資管理制度，持續找出問題。
10.使用紀錄、軌跡資料及證據保存	保存日後可用來舉證，證明已善盡管理義務的各種紀錄、證據、軌跡資料等。
11.個人資料安全維護的整體持續改善	定期持續檢討，找出問題，不斷改善，落實個資管理制度。

個資的當事人擁有五項權利，包括**查詢或請求閱覽**、**請求製給複製本**、**請求補充或更正**、**請求停止蒐集**、**處理或利用**、**請求刪除**等。因此，個資當事人可查詢個資蒐集者擁有哪些個資，以及資料的正確性，若資料不正確可提出更正，並可禁止個資蒐集者繼續使用其個資。

科技即戰力

請同學就《個人資料保護法》進行下列兩項問題討論：

1.請問在防疫期間的實聯制簡訊，疫情指揮中心是否可以提供給警方，作為追蹤疑犯的工具呢？

2.請問升學補習班可能是由何種管道取得學生的聯絡方式呢？是否有權利請求補習班刪除自己的個資呢？

而若違反《個人資料保護法》，相關罰則可能涉及民事責任、刑事責任與行政處罰（表13-1-3）。

表 13-1-3 違反《個人資料保護法》的相關罰則

民事責任	・每人每一事件可求償新臺幣5百元～2萬元。 ・同一件事，最高可求償2億元。
刑事責任	・最高可處2年以下有期徒刑、拘役或併科新台幣20萬元以下罰金。 ・意圖營利可加重求處5年以下有期徒刑，得併科新臺幣100萬以下罰金。
行政處罰	・最高可處新臺幣5萬元以上，50萬元以下罰鍰，並令限期改正，屆期未改正者，按次處罰。

事實上，政府部門就是國家中擁有最多個資的機關，即使政府是以合法且非營利為前提作蒐集，但也因為其個資資料庫數量龐大，種類豐富，最能吸引有心人士想透過政府竊取個資，因此即使是政府，也曾有個資外洩的案例（圖13-1-3），更何況是以營利為目的的企業，因此對於自己個資的提供仍須保持謹慎。

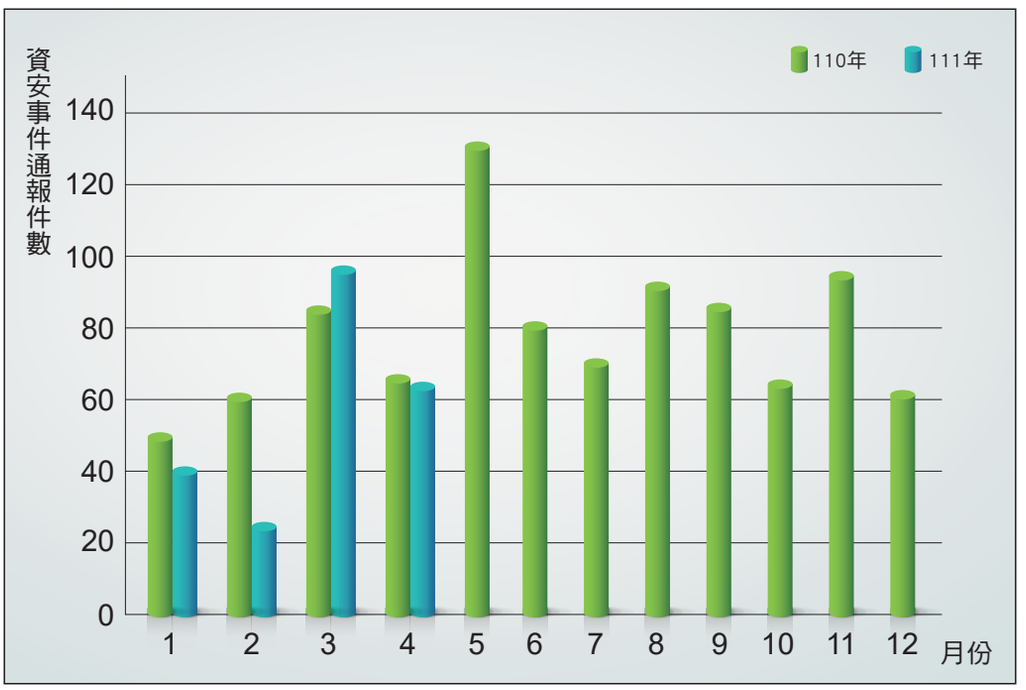


圖13-1-3 近年政府資安事件通報統計

資料來源：資通安全處

IT 補給站

個資案例討論

Q：求職時，公司私下請該名同學的畢業學校行政人員提供其在校成績等相關資料，請問學校是否可以提供？

A：學校無法判斷學生是否同意求職的公司到校蒐集其個資，若學生同意，應由學生主動向學校提出申請，並由學生或學校直接提供給公司。

Q：某校在公布欄上張貼上週遲到與缺曠課的學生姓名與相關紀錄，請問該校有違反個資法嗎？

A：若學校作為符合學校所訂定「教師輔導與管教學生辦法」之教育目的，則公布並不違反個資法。

Q：學校為了進行校外教學，從家長同意書中蒐集個別學生的健康狀況，當中有學生病史、健康資料等欄位資料，請問學校是否有權蒐集此類特種個資？

A：有關病史、健康檢查部分，需由學校的主管機關教育部，在蒐集學生個資上進行說明或是法制化，以便學校單位遵行。

Q：學校是否可以寄發建校週年紀念商品廣告給學生或校友？

A：學校蒐集學生個人資料之最初目的是為了「教育行政」，或學生資料管理。學校寄發非關教育的文書資料給學校或校友，可能會有踰越上述以教育為目的，除非取得校友之書面同意，否則應不得為之。

Q：是否可以在網路平台上公開街頭隨機攝影或行車紀錄器影像？

A：為避免造成一般民眾生活上的困擾，新版個資法第51條規定，只要單純為了個人或家庭活動，而去蒐集、處理或利用個人資料，就不在個資法的限制條件內，亦即不用對陌生但入鏡的人、物，主動告知照片使用範圍及使用目的，在公開場合拍攝的人物，只要沒加上足以識別該人物的個人資料，就沒有違反個資法的疑慮，但仍須注意肖像權的問題。而行車記錄器所拍到的車牌，許多人會做影片處理再放上網路，車牌所有人並不代表就是實際駕駛人，所以不牽涉到個資法問題，但卻跟隱私權有關，為求慎重將車牌去識別化處理是合理的方式。

Q：網路平台上進行「肉搜」、提供懶人包，是否構成違反個人資料保護法？

A：網友們基於興趣或者維護社會正義等原因，時常在網路上發動「人肉搜索」，但以這種方式「蒐集」而來的資訊，絕大多數也都是當事人主動在網路上公開的個資，這部分並無觸法之虞，只要不是透過非法入侵資料儲存設備所取得，就沒有問題。但是在個資的「 用」的層面來看，因為在網路上仍然可能有少部分是當事者未主動公開的個資，而是透過當事人的朋友在未經同意下公開在網路上，則無論是整理懶人包的網友，或是將當事人個資放在網路平台上的朋友，由於在無關公眾利益的情況下隨意將他人的個資公布到網路上，因此確實有可能觸法。

13-2 網路上的隱私與保護

隨著我們生活中對食、衣、住、行、育、樂的需求漸漸地都可以透過網路來實現，生活便利性提升的同時，也讓網路隱私問題逐漸浮上檯面，例如：多數人在網路上瀏覽商品之後，會發現自己的社群網站頁面也開始推播方才商品的相關廣告；又或者恰巧路過某個景點，手機的 APP 卻開始提供周遭的訊息。這是因為巨量資料時代的來臨，網路上的行銷已經化被動為主動，許多業者為了能提供商品資訊，並達到精準行銷的效果，默默地藉由瀏覽器所記錄的Cookie檔，分析我們在瀏覽器上的一舉一動。

在網站上，除了我們公開發表的文章以外，上傳的多媒體資料，社群媒體的按讚和打卡等，甚至是每一次超連結的點擊瀏覽、輸入資料表單、頁面停留時間、滑鼠游標位置……，所有在網路上活動的過程，都會留下我們個人的數據和蹤跡，此即為**數位足跡**（Digital Trace），也是廠商拿來進行巨量資料分析的一部分。由此可知，有太多的資料在我們未被告知的狀況下被蒐集，甚至是進行販售。

13-2-1 Cookie與瀏覽器的關係

傳統 Cookie 網頁瀏覽紀錄的儲存技術是在本地端系統，儲存使用者瀏覽的部分數位足跡。例如：在填寫某些網路表單時，瀏覽器可以協助我們填入先前填過的姓名、e-mail、電話、地址等個人資料；又或是再度造訪某個網站時，可自動登入網站服務。

Cookie 的原意是為了增加再訪網站時的便利性，將每一位使用者的瀏覽紀錄儲存在 Cookie 中，但這些巨量資料透過演算法的分析組合，雖然替我們帶來相對的便利性，卻也增添了隱私暴露的風險。

因此有許多瀏覽器也推出了無痕視窗的瀏覽模式，如 Google Chrome、Apple Safari、Mozilla Firefox、Microsoft Edge 等，使用者可以藉由無痕視窗瀏覽網頁內容、輸入資料，不會留下任何資料或紀錄。例如：搜尋紀錄、點擊過的超連結、登入帳密和 Cookie 都不會儲存在該裝置上，透過無痕視窗就不必擔心別人知道我們瀏覽過哪些網站。但無痕視窗並不代表傳送出去的資料有經過加密，駭客還是有可能從中竊取資料。

IT 補給站

第三方 Cookie 的應用

在現今的網際網路世界中，Cookie是洩漏使用者數位足跡與個資隱私的原兇之一。當使用者瀏覽某個網站時，網站伺服器會將使用者提供的資料和瀏覽行為傳遞至瀏覽器端，瀏覽器再將這些伺服器傳來的資料一一存放在本地端的Cookie.txt檔案中，當中包含了使用者帳號、密碼、姓名、電話等個資。

然而，Cookie也被電商平台用來追蹤使用者的數位足跡，當使用者再度造訪某個網站時，瀏覽器會將Cookie的資訊一併傳給伺服器，伺服器再藉由這些數位足跡，推送相關的商品或廣告，此即為第一方Cookie，當使用者離開這個網站時，其他網站的伺服器則無法使用這個網站的Cookie資料。

但我們常在瀏覽某些商品資訊後，進入另一個網站時，卻可以看見自己在其他網站瀏覽過的商品廣告推播，此為第三方Cookie的應用。例如：目前使用者欲購物的電商平台A網站運用Cookie，以跳出新視窗或是嵌入社群網站B的廣告橫幅等方式，讓使用者進入B網站時，也等於是在該網頁瀏覽了A網站的商品廣告區塊，同時也會自動擷取先前使用者在A網站中的Cookie瀏覽紀錄，因此可以更精準的推播商品廣告訊息給使用者（圖13-2-1）。

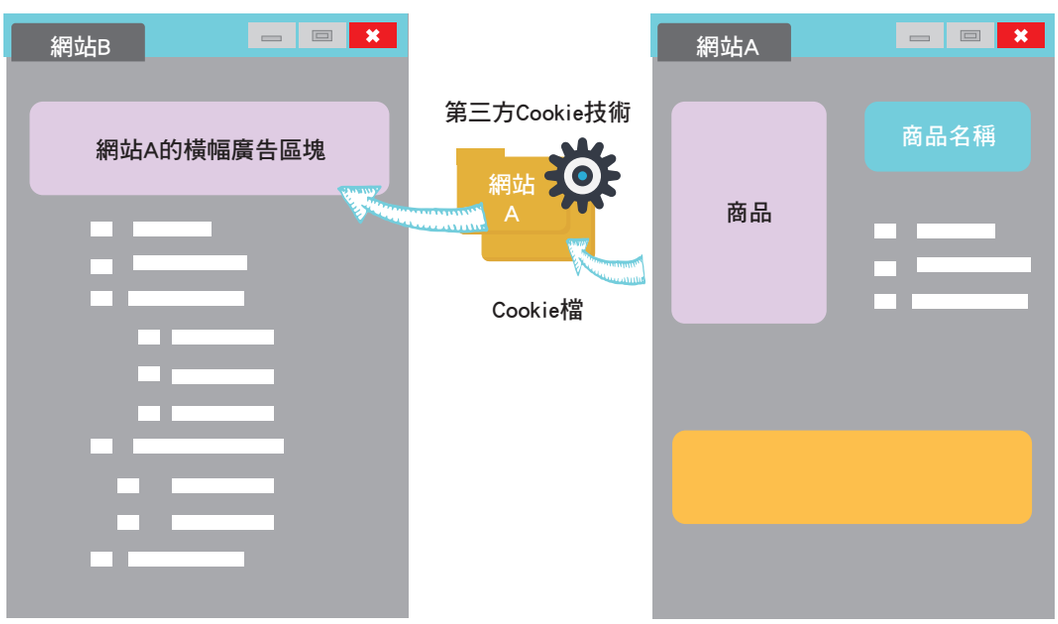


圖13-2-1 網站A運用Cookie方式嵌入第三方網站B的橫幅廣告區塊

如今，各國相繼限制電商平台在網路中蒐集消費者資料的行為，因此許多瀏覽器陸續不支援Cookie的技術，Google也在2021年初正式宣告自家瀏覽器Chrome將不再支援Cookie網頁紀錄儲存的技術。

請同學嘗試透過各自的瀏覽器（如Chrome、Opera、Edge等），刪除瀏覽紀錄、下載紀錄、Cookie檔、快取圖片、密碼和其他登入資料等等的各種瀏覽資料。

13-2-2 網路使用者保障隱私之具體行動

許多的社群軟體上的小遊戲或APP都會在使用前要求使用者提供存取個資的權限，若選擇不同意，則無法繼續使用該軟體，因此我們常常被迫同意，交出資料換取使用權限。

即使我們相信大企業會妥善保存我們的資料，但社群軟體中開發程式的第三方軟體或許只有蒐集資料的能力，卻沒有保護個資的能力。因此交出自己個資之前，須審慎評估後續造成的影響，一旦按下同意隱私權政策與蒐集的資料，其實是用我們珍貴的個資換取對方的服務。

另外，也因為物聯網的時代來臨，許多電子產品也能隨時和網路連接，並在我們不知情的狀況下，回傳資料或畫面至廠商伺服器，如網路攝影機、智能音箱、智慧手環、電動車、行動支付等，因此相關產品在選擇及使用上，都不得不謹慎。

以下介紹幾種保護自己隱私的方式：

1. 不直接以第三方帳號登入 APP 或網站服務

若以Facebook、Google等帳號登入第三方服務，第三方程式取用的不僅是會員基本資料，連帶著帳號所有資料都可能會被取用。如通訊錄、GPS資訊等。

2. 避免點擊來路不明且吸引人的連結

避免點入社群網站或通訊軟體上來路不明的心理測驗、小遊戲或是LINE免費貼圖連結，通常駭客常以這些活動偽裝騙取個資，此即釣魚網站（Phishing）。

3. 使用雙重驗證（Two Factor Authentication，2FA）程序

使用兩種不同的方式或因子，合併在一起並確認使用者的身分。除了登入該購物網站或APP外，若要進行付費，則需要經過第二道付款驗證程序確認身分，例如：銀行端透過客戶所註冊的手機號碼發出驗證碼的簡訊，達成刷卡人身分的驗證。

知名通訊軟體LINE，若在電腦版登入後，系統察覺IP位置為首次紀錄，在登入後會要求使用者將手機版LINE跳出的驗證碼輸入至手機端中，進行多道的身分確認，以防帳號、密碼遭竊取。

4. 加入 APP 會員要審慎

許多企業都推出自家的APP軟體供使用者加入會員，當消費時可以透過APP獲得點數、換取折扣或優惠，其實這都是企業為了誘使消費者加入、蒐集會員個資而提供的蠅頭小利。

5. 遠離盜版影音網站

盜版的串流影音網站多數是藉由廣告營利，因此會藉由前面所介紹的第三方Cookie嵌入廣告的方式，推送廣告給使用者。部分網站更可能在瀏覽影片的同時，植入病毒或木馬程式以竊取使用者個資。

6. 使用有提供閱後即焚功能的通訊軟體

目前在Instagram與FB的Messenger都有支援「閱後即焚」模式，這項功能是在對方讀取訊息內容或是圖片後，在一定的時間內自動消失，不會留下任何對話紀錄。當進入Messenger悄悄話模式（圖13-2-2）時，此時資料的傳輸是採裝置間、點對點的加密訊息傳送，等同於對話沒有透過Messenger的伺服器，而是直接傳輸到對方裝置，因此Messenger的系統管理者無法得知這段對話內容。

Messenger的悄悄話模式並不會主動進行閱後即焚的動作，而是必須啟動輸入文字框的倒數計時器功能，這項功能除了能讓對話有隱私性，也可以防止訊息內容、圖片不會因為帳號、密碼被竊取，而使重要的訊息內容外流。

STEP 01
前往「悄悄話」功能。

STEP 02
自動銷毀的倒數計時器最少5秒，最長為1天（24小時）。

圖13-2-2 Messenger設定悄悄話步驟

7. 數位簽章（Digital Signature）與加密技術

在資訊安全的領域裡，有項應用是確認資料由傳送方本人所傳送（**身分驗證**）且未遭竄改（**完整性**），傳送方也無法否認自己傳遞資料的事實，此為**不可否認性**（Non-repudiation）。我們必須確認某些活動是本人所提出的，例如：網路交易、以自然人憑證申辦各類服務等，這都可以運用**公開金鑰**技術來完成。

而公開金鑰技術，簡單來說，就是每個人都會有一對金鑰，分別為可以公開在網際網路上的**公鑰**（Public Key）與只能自己留存的**私鑰**（Private Key）。

以下列A想要傳送訊息給B這個例子（圖13-2-3）來說，B要如何確認訊息是來自A本人所傳送的呢？

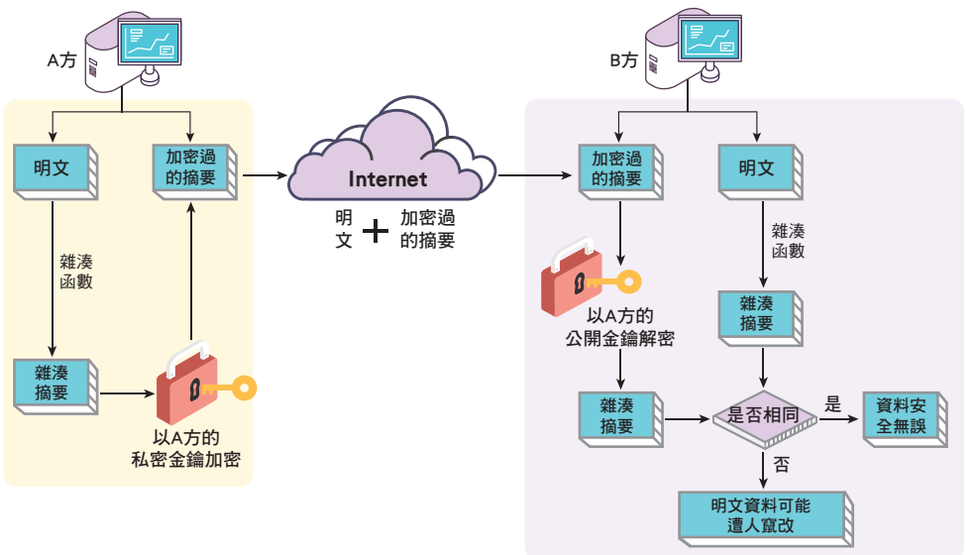


圖13-2-2 以公開金鑰技術實作數位簽章

A在傳送訊息前，先將明文訊息以**雜湊函數**（Hash）計算出**摘要**（Digest），再以自己的私鑰將摘要加密後，把明文與加密後的摘要一同傳送給B，B接收後再以A的公鑰進行摘要解密，而後比對明文計算出的摘要，兩者摘要是否相同。

倘若摘要相同，就可以確認這則訊息是由A本人發出的；萬一訊息在傳輸過程中被駭客所截取，即使駭客可以取得A的公鑰，解密並竄改原本的訊息，仍無法取得A的私鑰再次加密送至B端，因此可以達到身分驗證、訊息的完整性與不可否認性。

在上述例子中，A以自己的私鑰將訊息加密後，在傳輸至B前，會先取得B的公鑰加密，再傳輸至B端，而這則最後經由B公鑰加密後的訊息，唯有B的私鑰才能解開，如此便能完成傳輸過程的**隱密性**。

科技即戰力

1.請嘗試使用具有閱後即焚功能的通訊軟體，和同學進行訊息傳輸。
2.請找出任何可能含有釣魚網站的郵件、訊息或頁面，將此訊息截圖，並和同學討論此連結的風險性。
3.請找出安裝APP時，其對使用者要求哪些疑似不合理的權限設定，並在課堂上進行討論。（如導航軟體要求存取相簿功能、手電筒軟體要求存取通訊錄等）
4.請分析臺灣人工智慧實驗室在防疫期間所開發的「臺灣社交距離」APP，是如何得知使用者與確診者曾經近距離接觸呢？
5.在Google Map中，「你的時間軸」記錄了你的哪些隱私資訊呢？

課堂檢測

（　　）1.「《個人資料保護法》強調須經當事人同意才能蒐集與處理其個資，而且僅能用於特定目的。民眾為了使用地圖導航或方便與親友聯繫，只能同意交付『行蹤』與『通訊內容』等個資給Google或LINE這些資訊服務業者，若不同意，就不能使用業者提供的服務。然而，業者也因此得以持續獲得使用者源源不絕的個資。」根據這段文字，下列敘述何者最恰當？
(A)使用者交付行蹤與通訊內容，目的是為了讓業者蒐集個資
(B) Google與LINE的安全機制可以有效防止個資外流
(C)《個人資料保護法》規定若不同意交付個資就不能使用服務
(D)「當事人同意」及「用於特定目的」讓業者可合法處理個資。

（　　）2. 下列關於《個人資料保護法》的敘述，何者不正確？
(A)醫院裡的病歷與診斷資料雖然是由醫療人員負責記錄，但仍屬於患者的個人資料
(B)公務機關因執行公權力，故不受《個人資料保護法》限制
(C)學生的出缺席紀錄屬於個人資料
(D)使用紙本或網路表單所蒐集的個人資料，都受到《個人資料保護法》保護。

（　　）3. 啵啵畢業後任職銀行行員，對於其客戶個人資料之處理方式，何者敘述有誤？
(A)依照電腦處理《個人資料保護法》辦理
(B)可放置於公開場所，以便其他同仁隨時連絡服務
(C)須經客戶同意，方可將客戶之基本資料、財務資料共用
(D)資訊人員應對客戶資料具保密之義務。

（　　）4. 僑虎今天去科學園區面試新工作，公司拿出空白履歷表請他填寫，在面試後結束後，依據個資法，僑虎可對這間公司所蒐集的個人履歷行使下列何種權利？
(A)請求變更資料使用之方式　(B)請求變更資料蒐集之內容
(C)請求停止蒐集、處理或運用，並可請求刪除　(D)請求協商付費方式。

（　　）5. 續上題，僑虎發現履歷表中有項表格似乎不太符合個人資料保護法規定所能蒐集的資料，試問可能是下列哪項呢？
(A)身分證統一編號　(B)犯罪前科　(C)目前年收入　(D)畢業學校。

（　　）6. 某校為加速畢業生之就業 ，與某人力銀行合作，預計提供學生學習歷程資料，以快速執行工作配對之作業，前述案例是否符合個資法之規定？
(A)符合，學校只要是為學生著想，所有學生個資均可任意處理及利用
(B)符合，因為學生在校學習歷程的資料屬校方擁有，學生不可有異議
(C)應進一步確認是否逾越特定目的之必要範圍， 如果逾越則須取得學生書面同意
(D)以上皆非。

（　　）7. 安葆發現在搜尋引擎上搜尋自己的帳號名稱，可以看到自己以前在社群平台的發文和留言等，這是屬於下列哪一項紀錄？
(A)流量分析　(B)瀏覽器紀錄
(C)個人資料　(D)數位足跡。

( ) 8. 在瀏覽某些商品資訊後，進入了另一個網站時，卻可以看見自己在其他網站瀏覽過的商品廣告推播，是指下列哪一項技術的應用？
(A)第一方 Cookie (B)第二方 Cookie (C)第三方 Cookie (D)第四方 Cookie。

( ) 9. 以下是某學校想對學生宣導的資訊安全標語，其中何者不正確？
(A)不點擊來路不明的連結
(B)將智慧型手機交由師長管理
(C)不下載來路不明的免費貼圖
(D)遠離未經授權的影音平台。

( ) 10. 曉華到臺北念大學後在銀行開了一個新帳戶，方便臺中老家的家人匯學費、生活費給她。請問她在提供銀行帳號給家人時，下列哪一種方式最能有效降低資料外流的風險？
(A)採用閱後即焚模式傳送銀行帳號資訊
(B)將存摺封面拍照傳給家人
(C)傳輸完畢後自己把訊息刪掉
(D)傳輸完畢後刪除 APP，暫不使用。

( ) 11. 小希在搭公車回家的路上，因為坐著睡著而沒有讓位給站立的孕婦，進而遭到了別校的同學拍照並上傳至社群平台，許多人也開始「肉搜」小希的個資並留言在文章下方，試問下列哪一項非屬小希的個資呢？
(A)就讀學校 (B)住處地址 (C)公車路線 (D)家裡電話。

( ) 12. 柏榕以網路報稅向國稅局申報所得稅時，國稅局能依下列哪一種技術確認此報稅紀錄是柏榕本人的填報資料？
(A)霍夫曼編碼 (B)數位簽章 (C)數位審核 (D)數位足跡。

( ) 13. 下列哪一項有關數位簽章的敘述有誤？
(A)接收端使用傳送端的公鑰驗證 (B)可用來確保被傳送資料的機密性
(C)將明文轉換為雜湊值，在加密後送出 (D)用來確認訊息傳送者的身分。

( ) 14. 以下何種情境非使用雙重驗證？
(A)去郵局提款，插入晶片卡並輸入密碼
(B)去商店用信用卡消費，刷卡後簽名
(C)登入網站會員頁面，輸入使用者名稱和密碼，再輸入網站透過手機傳來的驗證碼
(D)用登記過的指紋或密碼開啟 iPhone 手機主畫面。

( ) 15. 元元熱衷於網路平台購物，今天他打算註冊加入一間新網站的會員，試問透過下列何種方式，元元比較能掌控自己提供給該網站的個人資料？
(A)填寫該網站註冊頁面上提供的表單後加入會員
(B)透過通訊軟體（如 LINE）的帳號加入會員
(C)連結社群網站（如 Facebook 或 Instagram）的帳號加入會員
(D)綁定學校提供的 Google 帳號後成為會員。

chapter

14

資訊科技對人類社會的影響與衝擊

資訊科技為我們帶來了許多的正面的影響，如機器的自動化、**網紅產業**的誕生、**共享經濟**的興起等，也可以透過網路平台，直接參與各類科學研究計畫與國家政策規劃的過程。但科技除了帶來上述的好處外，同時也造成了整體社會的衝擊，如就業環境的改變、城鄉差距與**數位落差**的擴大、假訊息的危害等。

隨著現代科技日新月異，我們的生活也越來越便利，但同時我們也應探討如何面對科技變革所衍生的危機，期許在不斷追求創新的同時，也能讓社會與環境永續發展。

14-1 對就業環境的影響與衝擊

由於科技領域不斷地創新，所驅使的產業自動化為社會的整體就業環境帶來了巨大的變遷，並徹底改變工作的型態、就業市場的需求、薪資的結構等面向。

14-1-1 整體就業機會的更動

科技的創新會帶來新的工作機會是無庸置疑的，但相對地，也因為科技造就了各行各業的**自動化**，使得許多工作機會逐漸減少或消失，而減少的勞動力數量與新增的工作機會之間的消長，也是人力資源中值得關注的議題。在多數工作的**生命週期**大幅縮短的情況下，就業者必須不斷更新自己的能力，以滿足未來工作型態的需求。在求學階段也要形塑自己跨領域學習知識的能力，以備將來能迅速適應就業市場的變動。

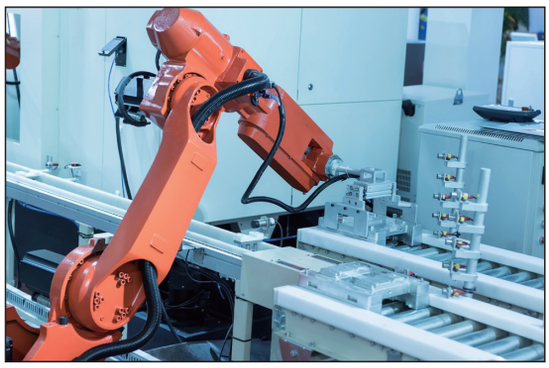


圖14-1-1 電腦視覺與手臂整合，讓產線作業效率更高

雖然人類無法像機器那樣善於快速、精確地處理大量數據及重複性高的例行性工作（圖14-1-1），但大多數的機器只能解決特定領域的任務，而人類則有辦法隨時調整、學習工作內容。因此在面對團隊合作、溝通以及解決問題等情境時，人類相較於機器就顯得更有彈性。

14-1-2 誕生全新的工作類型

新科技帶來許多全新的工作類型，例如：對於物聯網、巨量資料、人工智慧等領域，目前的就業市場仍是求才若渴，也因為數位平台媒合工作快速，使得這些工作可以依據公司的營運策略，將人力進行不同的規劃，如工作委外、短期顧問、約聘派遣、共享經濟等模式。拜科技所賜，讓需要工作的求職者和求才單位中間媒合的過程扁平化，也讓不少個人的工作室如雨後春筍般地冒出，如接業配廣告的**網紅經濟**、Uber多元計程車的**共享經濟**等新的工作型態。

共享經濟是指個人將擁有的閒置資源藉由網路平台與他人共享（表14-1-1），進而創造收益的經濟活動。當中包含了四個要素：

1.個人：
為提供服務的主體，但相繼也有企業或機關投入。如各縣市政府的公共腳踏車租賃。

2.閒置資源：
包括資金、房屋、物品等動產或不動產，或是個人知識、技能、經驗等。

3.社群平台：
透過網路平台作為實現大規模的共享經濟模式。

4.收益：
主要有租借、二手交易和人力資源共三種模式。如網路平台租借場地、網路跳蚤市場、美食外送員等方式進行收益。

表 14-1-1 臺灣常見的資源共享平台

汽車共享	Uber	房屋共享	Airbnb
	Zipcar		共生公寓
	LINE 私人叫車群組		
機車共享	GoShare	空間共享	PickOne
	iRent		WeWork
單車共享	uBike	廚房共享	DearChef
			私廚
車位共享	USPACE	蛋共享	臺灣好蛋大平台
物流共享	GoGoVan	家事共享	潔客幫
	Lalamove		

科技即戰力

請嘗試使用共享平台的資源，和同學分組規劃兩天一夜的班遊。

IT 補給站

群眾募資

科技的創新造就了許多新穎的經濟型態，除了網紅經濟、共享經濟外，藉由網路進行**群眾募資**也是種新興的商業行為。目前在臺灣較普遍的群眾募資類型為**回報型群眾募資**（Rewards Crowdfunding），當募資者需要資金進行製造、量產商品時，向消費者以預購的方式先募得足夠的資金，而消費者則可以用較優惠的價格預購商品，或是先預購才能獲得的限量贈品，藉此吸引群眾投資其生產。在臺灣有Flying V、嘖嘖、挖貝等相關募資平台，在上述平台中，可以發現作為募資的標的物非常多樣化，從一般常見的科技產品乃至於個人專輯創作、舞臺劇的演出、公益活動等也都能藉由群眾募資達到其目的。

圖14-1-2 群眾募資網站也能創造出友善環境的公益性質商品

14-1-3 城鄉差距擴大

由於許多科技產業的技術具有高度的專業性，如晶片設計、程式設計等，使得進入這類產業的門檻相對地提高，而其內部核心技術人員的薪資相較於一般受薪階級也是屬於所得較高的一群，因而產生了貧富差距擴大的隱憂。而這些問題從個人、家庭開始，甚至可能導致整個國家的區域發展失衡，例如：我國主計處指出，平均薪資最高的地方就是新竹科學園區，顯示這個區域的發展會因為收入的提高、人口的驟增，加速該區域的建設和發展，使得城鄉的差距因而擴大。

14-2 對文明社會的影響與衝擊

14-2-1 對社會的負面影響

假訊息充斥

假新聞與**假訊息**的散播儼然造成社會進步的阻礙。過去我們彼此間因為互相信賴而在網路上傳遞訊息、發表文章；但在近幾年，有心人士為了特定目的，在網路上有計畫地散布假訊息，這種情形甚至有愈來愈氾濫的趨勢。尤其透過資訊科技傳遞訊息，大幅縮短了傳遞的時間和空間，這些假訊息往往會像漣漪般地藉由網路迅速擴散開來，如先前駐日代表於日本風災期間受到假訊息的波及，進而受到國內外網路輿論的撻伐。

2018年，瑞典哥德堡大學研究調查發現，臺灣是全球「遭受境外假資訊攻擊」最嚴重的國家，顯示臺灣除了內部政黨間為了利益鬥爭而興起的網軍，國外網軍散播的假訊息也影響國內的安定。世界各國在大選期間也同樣遭受境內外網軍散布假新聞的肆虐，以抹黑的手段攻擊對手，或是以大量的文章訊息掩蓋真相，試圖左右選舉的結果。

假訊息通常有以下特徵：

1.使用聳動或隱晦不明的標題，吸引大眾點閱連結
2.消息來源難以追溯
3.文章內的時間不明確
4.欠缺研究數據佐證

避免假訊息的散播有以下方法：

1.文章中專家發表的資訊要查證
2.親友分享的文章或圖片需查證
3.不點擊來路不明或標題聳動的連結
4.不任意分享、散播沒查證的文章或圖片

科技即戰力

請試著找尋自己的社群軟體（Facebook、Instagram）或是通訊軟體是否有疑似的假訊息，並請嘗試運用以下網站進行求證。

1.TFC台灣事實查核中心網站　　2.LINE訊息查證
3.MyGoPen　　4.蘭姆酒吐司

資訊科技的數位落差

由於網路科技對於性別、年齡、教育程度、都市化程度、種族、職業、收入、區域或國別之間等因素存在著程度不一的狀況，尤其以開發中國家透過網路科技獲得資訊的機會落差最為顯著，這種衍生的差異性稱之為**數位落差**。

現今對於數位落差一詞也逐漸由數位產品的「量」，轉變為「質」的落差。近年來，國家以政策極力地減少網路基礎設施與家庭數位硬體設備上的不足，由偏鄉校園推動至弱勢家庭，硬體這部分的落差程度已逐漸縮小。

然而，在人人都有智慧型手機的社會下，如何使用資訊科技獲取正確資訊才是目前需要加強的素養。國外也有研究發現，過度接觸數位產品會導致學童社交及語言能力的下降，如何正確地使用科技也是資訊素養的一環。

14-2-2 公民參與

公共政策網路參與平臺設立及推動說明

隨著資訊與網路的發展，民眾經常透過各大社群平台，表達對政府決策的意見，如Facebook、Twitter、臺大批踢踢實業坊等，上述工具縮短了人民參與政治的距離，甚至可以運用網路輿論，影響整個國家政策的走向。

對於民主國家來說，政策制定和決策過程都希望是公開透明的，美國白宮就曾經藉由網站「We the People」，作為人民和政府正式溝通的管道，公民可以在此平台提出請願，若達到一定人數的附議，政府便必須回應此請願內容。

2015年起，我國也效仿白宮的網路公民參與模式，建立**公共政策網路參與平臺**（圖14-2-1），期許在決策開始的過程能讓政府聽見民眾的聲音，縮短政府決策和公民距離，降低公民參與公共事務的門檻。

我國的代議制仍有少數民代做出不利多數人決策的盲點，因此透過平台的參與，廣徵民眾的建議，集結公民的集體智慧，對於政府政策提出不同的考量面向，也大大地影響政府部門在制定法案或規劃建設的決策。除了參與決策過程外，平台上的「眾開講」、「來監督」也能預告即將實施的政策草案，同時也將施政成果公告在此平台，讓民眾從政策的制定、決策、實施、成效完全都能參與其中，而不用走上街頭表達訴求，也能降低社會成本的付出。

圖14-2-1 公共政策網路參與平臺

由國家發展委員會推行的公共政策網路參與平臺，網站中「提點子」是瀏覽度最高的項目之一，民眾可以自主發起提案，若在60天內達到5000位以上認同此案的附議數，相關部門就必須在2個月內對於本提案進行回應（圖14-2-2）。有些提案即使未能成案，但也成功地讓政府部門看見少數族群的聲音和需求，進而參考其提案內容。

例如：提案之一的「請國家生產0.01%的Atropine，抑制近視」，雖未達附議通過成案，但其提案之藥物經新加坡國家眼科中心進行臨床實驗，確實以0.01%的阿托品（Atropine），在幾乎無副作用的情況下，有效減緩六歲和以上孩童近視加深的速度達五、六成。政府在評估其可行性後，也協調衛福部與藥廠加速進行開發與審查。

除了公共政策網路參與平臺，臺北市也嘗試過網路海選市府局長、參與式預算i-Voting，雖然成效不佳，但網路參與政策仍處於萌芽階段，民眾和政府都需要時間適應與互信。

「透過不同管道蒐集意見，最重要的好處是蒐集不同聲音」，網路的特性讓每個人都有機會發聲，社會有多元管道反映民意，才能讓小眾、弱勢族群的聲音也能被聽見。

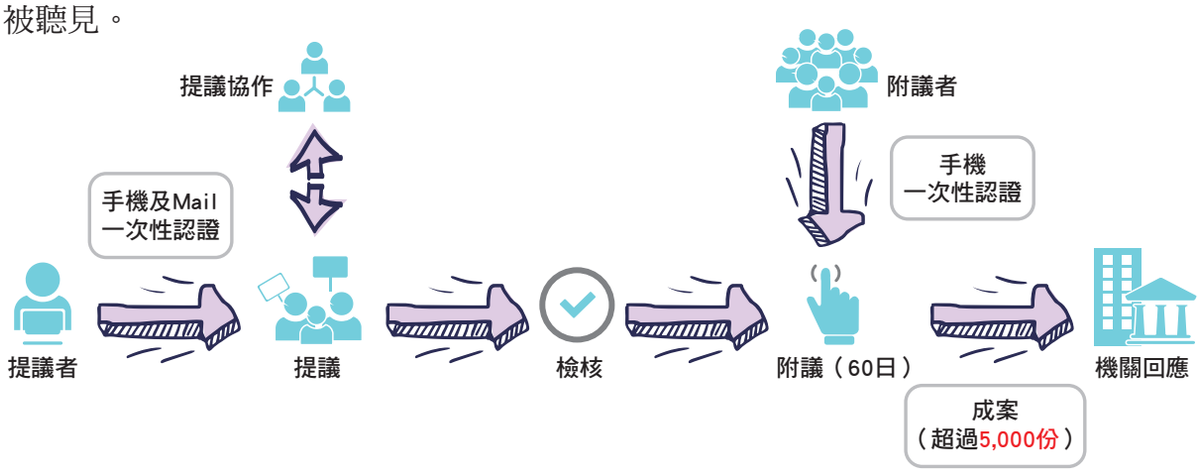


圖14-2-2 公共政策網路參與平臺之提議附議流程圖

科技即戰力

請同學進入公共政策網路參與平臺網站（https://join.gov.tw/），找出三項支持的議題進行附議。並在「來監督」中，找出今年度行政院所屬各機關年計畫經費的分配。

政府開放資料

在防止洩漏民眾個資的前提下，政府部門藉由有系統地蒐集與開放數據，可提升公民參與國家政策的推行，民眾能更了解政府的施政成果（圖14-2-3）。

我國政府在開放資料的政策上，聚焦於四大焦點：「**資料的品質**」、「**制定開放資料的規範**」、「**推動共用平台**（data.gov.tw）」、「**示範宣導及服務推廣**」上（圖14-2-4）。

圖14-2-3 政府系統化地蒐集及提供開放資料，對民眾及企業所帶來的好處

資料開放的種類以提升民眾生活品質為主，例如：在食、衣、住、行、育、樂等方面，透過政府資料的開放，橫跨機關與民間，促成更多合作與創新的服務。

公開資料的透明化，除了讓政府內的各部門充分利用數據進行決策，對外也可強化民眾對於知的權利。

民眾除了透過公開的數據了解各部門執行政策的進度與成果外，政府或民間也可以結合日益成熟的AIoT與5G技術，開發更多便民的應用服務。根據平台統計，下載量最大的前四名分別是天氣、交通、實價登錄、空氣品質，都與民生需求息息相關。開放資訊的內容愈多元，提供的服務才能更多元（圖14-2-5）。政府在開放資料這方面，是屬於幕後提供數據的角色，讓民間運用無限的創意，將原始資料創造出更多元的加值應用。

2020年，在全球疫情的蔓延下，臺灣也因為擁有完善的資料開放基礎建設，許多關於防疫的APP應用也迅速地推出，如「口罩地圖」就是由民間發起製作。因此政府開放資料能夠替我們帶來諸多好處，如增進政府施政透明、發展各項跨機關便民服務、強化民眾監督政府的力量、開放資料給予民眾與企業應用等優點。

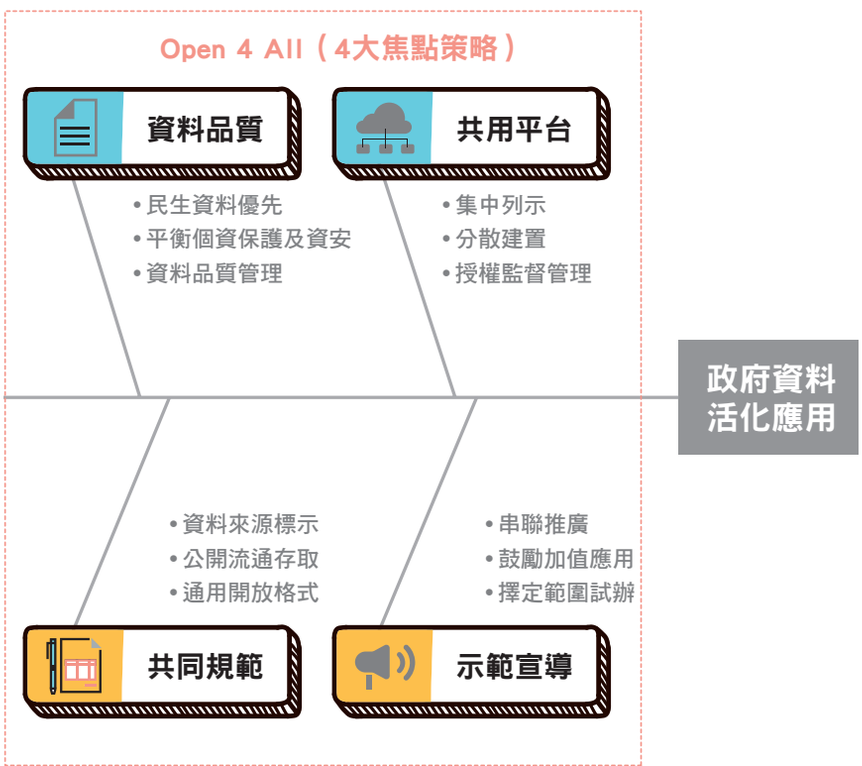


圖14-2-4 我國政府對於開放資料的工作，以期能將資料活化並達成更多的應用與服務。

圖14-2-5 政府資料開放平臺的資料類型以利於民生為主

14-2-3 公民科學

許多生物與環境的研究必須進行大規模的採樣和記錄，這是一般研究方法難以達成的目標。而**公民科學**藉由網路無遠弗屆的特性，透過一般民眾群策群力，大範圍地記錄生態和其他環境問題，例如：生物的遷徙路徑、物種的數量、環境的改變和汙染等。

對一般民眾來說，只要具備使用任何工具進行科學上的計量或記錄能力的人，都可以成為一位公民科學家。公民科學可分為以下三種類型：

貢獻性計畫	研究計畫先由科學家擬訂，民眾主要負責蒐集數據。
合作性計畫	一般是由科學家擬訂研究計畫，再由民眾蒐集資料，但一般民眾在蒐集數據的過程中，由於其在現場第一線的特性，可以協助科學家進行研究計畫的修正、數據統計與分析，或是推廣並宣傳研究成果。
共創性計畫	由科學家和具備與研究計畫之相關知識的一般民眾共同設計研究方法和步驟，當中會有部分的一般民眾積極參與整個或多數的科學研究歷程。

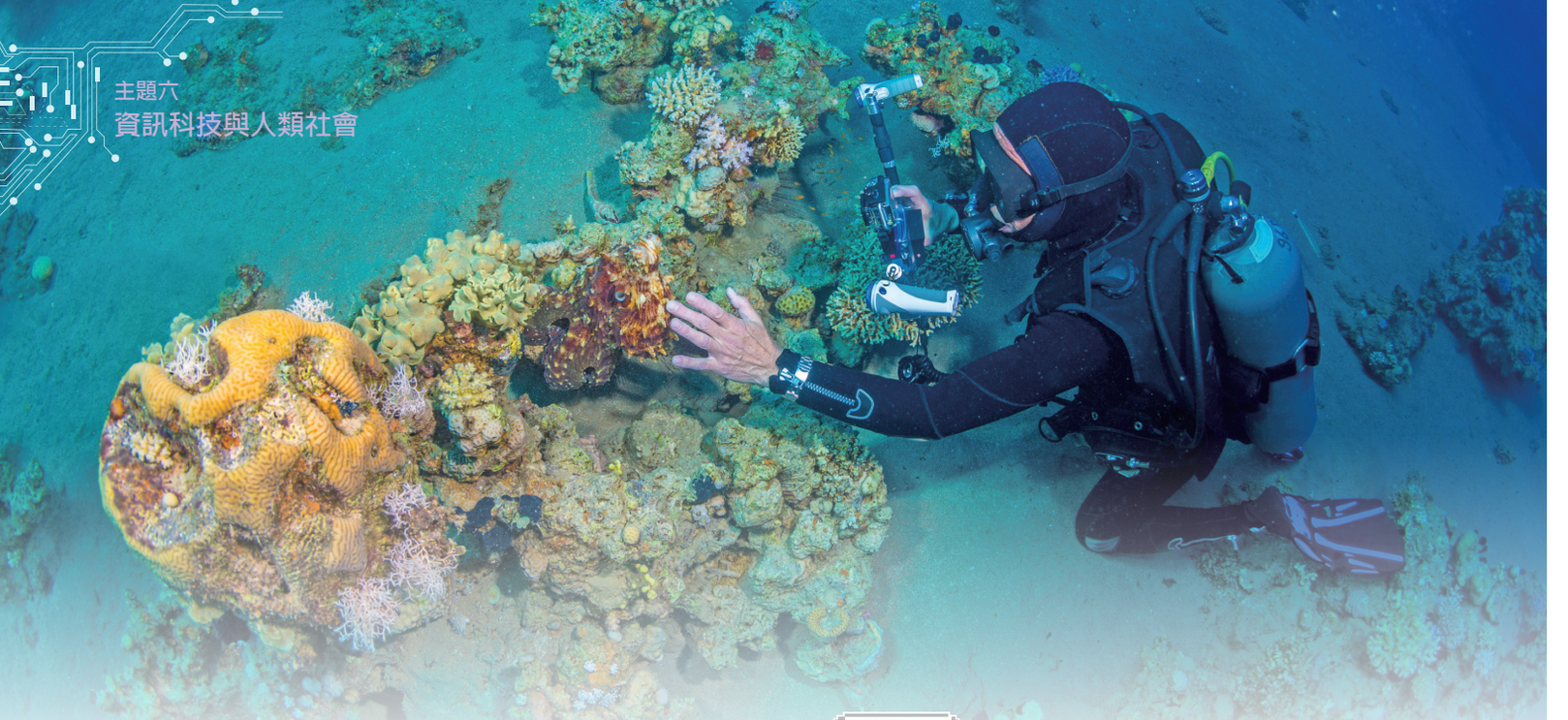


圖14-2-6 潛水員協助進行珊瑚礁復育計畫

共創性計畫之實例，如一般民眾搭乘直升機空拍臺灣，透過鏡頭記錄我們所看不到的開發和汙染、外洩的事件；墾丁海域透過具有潛水能力的潛水員，協助生物科學家進行海底的珊瑚礁復育計畫（圖14-2-6）。近期更發現公民科學在保育、生物多樣性和氣候變遷研究上的應用最為廣泛，其中又以候鳥與蝴蝶遷徙是最受歡迎的研究對象。

科技即戰力

請同學進入臺灣公民科學入口網（https://pansci.asia/tw-citizen-science），找尋有興趣的三項研究計畫，並試著和其他同學分享你將如何參與這三項計畫。

14-3 對未來的展望

科技未來與性別議題

現今，多數的科學研究領域仍以男性為主，即使許多研究指出生理性別和智力上的發展，兩者其實並無顯著的相關性，但由於傳統社會的刻板印象，造成女性從求學階段乃至於就業的限制和阻礙，在科技領域也沒有例外，能在科技領域就業的女性仍是少數。

從過去到現在，許多女科學家的成功典範也讓多數國家開始透過政策的制定，如我國的育嬰假、生理假等，積極地鼓勵女性在STEM（Science, Technology, Engineering, and Mathematic）科學研究領域從事發展。此時此刻，無疑是啟發下一代女性勇於踏入科技領域的最佳契機，藉由不同性別的觀點，能豐富科學領域的多樣和多元性。

美國為了鼓勵年輕女孩勇於踏入科學研究領域，除了以許多成功的女科學家作為楷模外，讓學齡階段的女孩親近科學也是一個重要方式，「科學女孩俱樂部」（Science Club for Girls）就是為了鼓勵年輕女性參與科學而設立的網站（圖14-3-1）。

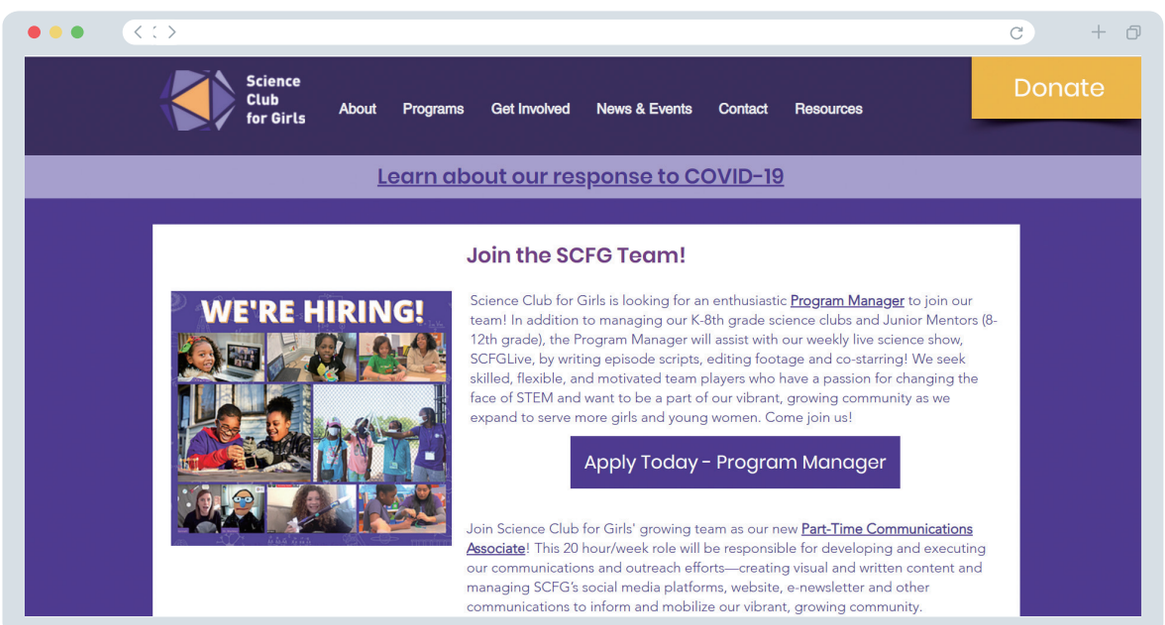


圖14-3-1 科學女孩俱樂部網站

2030科技生活

我國工研院提出的「2030科技生活」為科技未來趨勢與生活幻境衝擊的探討，在預想未來各種生活情境下，運用科技發展出**智慧生活**（Smart Living）、**健康樂活**（Quality Health）與**永續環境**（Sustainable Environment）三大應用領域。

為了達成上述三項目標，將未來研發與產業的發展著重於四大共通的基礎技術架構，其中包含了**人工智慧**、**半導體晶片**、**通訊**、**資安與雲端**共四大重點發展領域（圖14-3-2）。

圖14-3-2 工研院2030技術策略藍圖

智慧生活

在未來的10年，當手機仍是每個人生活上依賴的工具時，實際上人類正在經歷一場科技與生活相互影響的過程，更多的生活應用仰賴科技產品，影響的速度愈來愈快，涵蓋的層面愈來愈廣泛，透過各種手機的APP的研發與應用，滿足我們在食、衣、住、行、育、樂的需求。

健康樂活

當科技大幅提升我們的生活品質後，人類的壽命隨之增長，人口結構也趨向於高齡化，必須更重視在老年人口的健康與照護的需求，如科技與醫療，獨居長輩的線上關懷社群等。

永續環境

人口的增加成為了地球環境資源的負擔，慢慢地消耗著地球上有限的新鮮空氣、水資源、土地、食物、能源等，伴隨而來的是氣候變化、環境汙染，以及新型傳染病的出現，而這些因子正全面衝擊生活環境，有待於運用資訊科技化解這些未來逐漸浮現的危機。如智能環境監控、巨量資料分析氣候變遷、生產人工智慧節能科技產品、加速疫苗的生產（圖14-3-3）等。

圖14-3-3 生技公司分析疫苗及生物製劑之品質

人工智慧技術的持續發展將是未來各國的主流，運用人工智慧將可以使生活型態更為便利，企業也能提升營運效率，增加獲利。雖然人工智慧的技術仍有許多待解決的挑戰，但同時也是許多技術創新的契機，人工智慧與各領域的結合將是決定科技於未來的走向的關鍵技術之一。

課堂檢測

( ) 1. 銘銘是一間生產半導體公司的負責人，他想把「自動化」的概念引進公司，取代傳統大量人力的經營模式。下列哪一項既有的職務內容比較適合應用機器來達到自動化？
(A)開發新客戶
(B)嘗試研發新材料與製造程序，以節省成本
(C)找出生產機臺突然停止運作的原因
(D)在工廠內搬運物料到各個產線。

( ) 2. 下列何者不是高科技產業的特色？
(A)產業快速成長　(B)產品市場具有世界性
(C)產品生命週期長　(D)涉及先進或多領域技術。

( ) 3. 下列何者不屬於共享經濟的模式？
(A)透過App訂購美食，並且由外送員直接送達指定地點
(B)小李在下班後運用自己的汽車，加入多元化計程車服務，增加收入
(C)政府與民間合作投入大量的腳踏車，供人租用
(D)校園合作社於期末回饋盈餘給教職員生。

( ) 4. 下列哪一種情境較符合因數位落差而產生的影響？
(A)泱泱進行線上遊戲時，因為伺服器壅塞，導致畫面傳遞延遲
(B)偏遠地區的網路建置不普及，亮亮回到家無法進行線上學習
(C)華華無法馬上買到最新的iPhone版本進行線上學習
(D)云云沒有足夠的資金購買線上數學學習課程。

( ) 5. 透過電腦與網路科技的應用，人類社會起了重大變革，網際網路的便利為現代社會帶來一些正面的影響。下列敘述何者最能顯示網際網路在臺灣社會的正面功能？
(A)可減少人際摩擦、衝突，並提升互動的品質
(B)將精緻文化或傳統文化數位化讓民眾充分享用，促使文化融合
(C)可解決城鄉、貧富差距加大，以及資源分配階層化的問題
(D)提供最迅速確實的第一手訊息。

( ) 6. 華華在萬聖節收到了一則「限時！快來搶免費貼圖！！！」的連結訊息，可是老師提醒他要小心惡意連結。請問華華可以透過下列何者最即時、迅速地查證該訊息連結是否屬實呢？
(A) MyGoPen　(B)台灣事實查核中心　(C)蘭姆酒吐司　(D)Line訊息查證。

( ) 7. 崌崌在某次國內大選開票時發現，不只各家新聞臺的開票數差異極大，甚至有時還會出現「票數倒退／減少」的情形。試問下列哪一種方式能讓崌崌得到最準確的開票資訊？
(A)相信總開票數最少的新聞臺　(B)相信總開票數最多的新聞臺
(C)到中央選舉委員會網站查詢　(D)到台灣事實查核中心查詢票數。

(　　) 8. 科技產業中，往往都是男性員工多於女性，為了增加女性投入科技產業，政府也透過政策的制定等，鼓勵女性朝向科技產業發展，下列哪一項為科技領域中，導致男女員工人數懸殊的主要因素？
(A)男性的智力發展較趨向於理工科系
(B)傳統社會對於性別與志向的刻板印象
(C)女性的智力發展較趨向於文法商的科系
(D)公司可以拒絕女性員工提出育嬰假、生理假。

(　　) 9. 下列何種方法不能用來消除數位落差？
(A)廣布網路
(B)立法獎勵推行低收入區域之電腦化
(C)增設公共圖書館內的電腦設備
(D)降低關稅以獎勵投資。

(　　) 10. 關於目前常見的「回報型群眾募資」，下列敘述何者正確？
(A)類似資助者預購的模式，提供贊助者得以用較優惠的方式取得產品
(B)群眾向募資者提供想法和創意，待產品開始銷售後，共享一部分盈餘
(C)資金來自於政府與民間創投公司的投資，必須支付貸款利息
(D)群眾向募資者提供資金，待產品開始銷售後，共享一部分盈餘。

(　　) 11. 果果想提出對於學習歷程檔案的建言，試問她可以透過下列管道提案，較能獲得適切的回覆呢？
(A)透過公共政策網路參與平臺提案
(B)透過班會課提案
(C)向補習班老師提案
(D)在 Instagram 發文提案並標註國教署。

(　　) 12. 在 Covid-19 的防疫期間，有民眾運用自己的資訊能力，發展了口罩地圖讓社會大眾知道口罩銷售的據點及其即時數量，此為政府開放資料的哪一項好處？
(A)發展各項跨機關的便民服務
(B)增進政府施政的透明度
(C)強化民眾監督政府的力量
(D)資料開放民眾與企業運用

(　　) 13. 在學校的作業報告中，常常必須引用到政府資料開放平臺（data.gov.tw）的資料，下列何者是最正確的描述？
(A)政府資料開放平臺秉持使用者付費的精神，提供收費服務
(B)在政府資料開放平臺可以輕易下載和教育相關的巨量資料
(C)政府依據「制定開放資料的規範」，禁止修改、分享
(D)政府資料開放平臺不能提供學生使用，無法完成巨量資料專題製作。

(　　) 14. 一位熱愛登山的民眾常在登山途中遇見臺灣黑熊，因而決定將這些黑熊出沒的座標主動通知保育臺灣黑熊的研究人員，試問這是屬於哪一類的公民科學參與？
(A)發展性計畫　(B)合作性計畫　(C)貢獻性計畫　(D)共創性計畫。

索引表

A		
Access Point，AP	無線網路基地臺；無線存取點	191
Algorithm	演算法	038
Application，APP	應用程式	039
Application Layer	應用層	196
Application Software	應用軟體	171
Apriori Algorithm	關聯分析	243
Array	陣列	007
Artificial Intelligence，AI	人工智慧	204
Artificial Intelligence of Things，AIoT	人工智慧物聯網	214
Assembler	組譯程式	170
Atropine	阿托品	322
Attribution	姓名標示	294
B		
Big Data	巨量資料	065
Binary Search	二分搜尋法	158
Binary Search Tree	二元搜尋樹	016
Binary Tree	二元樹	016
Bits	位元	020
Block	區塊	182
Block Chain	區塊鏈	182
Border Gateway Protocol，BGP	邊界閘道協定	201
Broad Network Access	無所不在的網路存取	211
Brute Force	暴力法	044
Bubble Sort	氣泡排序法	143
C		
Case of Right to be Forgotten	被遺忘權	232
Char	字元	020
Child	子節點	014
Classification Analysis	分類分析	245
Client-Server Network	主從式網路	187
Cloud Computing	雲端運算	204
Cloud Service	雲端運算服務	180
Cluster	群集	248
Cluster Analysis	分群分析	245
Compiler	編譯器；編譯程式	076
Computation	運算	010
Computer Vision	電腦視覺	256
Confidence	信心水準	245
Connectivity	連接性	216
Conquer	解決	047
Copyright	版權所有	292
Creative Commons	創用CC	294
Cross Validation	交叉驗證	247
Cycle	循環路徑	013
D		
Data	資料	006
Data Link Layer	資料連結層	198
Data Mining	資料探勘	242
Data Structure	資料結構	006
Deadlock State	死結狀態	173
Debug	除錯	077
Debugger	偵錯程式	170
Decision Tree	決策樹	246
Deep Learning	深度學習	205
Deep Neural Network	深度類神經網路	205
Definiteness	明確性	040
Degree	分支度	015
Depth	深度；高度	015
Dequeue	刪除	010
Digest	摘要	313
Digital Signature	數位簽章	312
Digital Trace	數位足跡	308
Dijkstra’ s Algorithm	戴克斯特拉演算法	045
Distance Vector	距離向量	200
Distributed Systems	分散式系統	180
Divide	分解	047
Divide and Conquer	分而治之	047
E		
Edges	邊	013

Effectiveness	有效性	040
Element	元素；資料	008
Enqueue	加入	010
Exterior Gateway Protocol，EGP	外部閘道協定	201
F		
Fairness	公平性	207
Feature	特徵	252
Feature Learning	特徵學習	205
File Transfer Protocol，FTP	檔案傳輸協定	196
Finiteness	有限性	040
First Come First Served，FCFS	先到先服務	174
First in First out Page Replacement，FIFO	先進先出替換演算法	178
First In First Out，FIFO	先進先出	010
Float	浮點數；小數	083
Floating-point Operations Per Second，FLOPS	每秒浮點運算次數	063
Flowchart	流程圖	041
Frame	頁框；訊框	177；197
Free Software	自由軟體	297
Freeware	免費軟體	297
Front	前端	010
G		
Gantt	甘特圖	269
General Data Protection Regulation，GDPR	一般資料保護法規	233
Global Positioning System，GPS	全球定位系統	191
GNU General Public License，GNU GPL	GNU 通用公眾授權條款	298
Graph	圖	030
Greatest Common Divisor，GCD	最大公因數	041
Greedy Algorithm	貪婪演算法	044
Greedy Method	貪婪法	044
H		
Hash	雜湊函數	313
Height	高度；深度	015
Huffman Coding	霍夫曼編碼	020
Huffman Tree	霍夫曼樹	016
Hyper Text Transfer Protocol，HTTP	超文本傳輸協定	196
Hyperscale Data Center	超大型資料中心	213
I		
Index Value	索引值；序號	008
Infrastructure as a Service，IaaS	基礎設施即服務	210
Inorder Traversal	中序走訪方式	022
Input	輸入	040
Interior Gateway Protocol，IGP	內部閘道協定	201
Internet	網際網路	189
Internet Layer	網路層	197
Internet of Things，IoT	物聯網	194
Internet Protocol，IP	網際協定	197
Internet Service Provider，ISP	固網業者；網際網路服務提供者	189
Interpretability	可解釋性	207
Interpreter	直譯器	076
Interrupt	中斷訊號	173
IP Address	IP位址	197
Iteration	重複結構	092
K		
K-means	K平均演算法	248
K-Nearest Neighbor，KNN	K最鄰近分類法	246
L		
Label	標籤	205
Last In First Out，LIFO	後進先出	009
Leaf	樹葉	015
Leaky Cable	洩波電纜	218
Least Frequently Used，LFU	最近最少使用替換演算法	179
Least Recently Used，LRU	最近最久未使用替換演算法	178
Level	階層	015
Linear Regression Analysis	線性迴歸分析	249
Link Cost	連結成本	067
Link Layer	連結層	197
Linker	連結程式	076
Loader	載入程式	076
Local Area Network，LAN	區域網路	187
Location Base Service，LBS	適地性服務	238

Logical Error	邏輯錯誤	078
M		
Machine Language	機器語言	077
Machine Learning	機器學習	205
Measured Service	可計量的服務	212
Merge Sort	合併排序法	047
Model	模型	205
Module	模組	092
N		
Natural Language Processing，NLP	自然語言處理	255
Network Layer	網路層	198
New	新建	172
No Derivatives	禁止改作	294
Nodes	節點	013
Non-repudiation	不可否認性	312
Noncommercial	非商業性	294
O		
On-demand Self-service	隨選即用服務	212
Open Data	開放資料	231
Open Shortest Path First	OSPF 路由協定	200
Open System Interconnection Model	OSI模型	198
Operating System	作業系統	170
Output	輸出	040
P		
Packet	封包	196
Page；Paging	分頁	177
Page Fault	分頁錯誤	177
Page Replacement	分頁替換	177
Page Replacement Algorithm	分頁替換演算法	178
Parent	父	014
Pattern	特徵	205
Pay As You Go	用多少付多少	212
Peer-to-Peer Network，P2P	對等式網路	187
Phishing	釣魚網站	310
Physical Layer	實體層	198
Platform as a Service，PaaS	平台即服務	210
Pop	取出	009
Postorder Traversal	後序走訪方式	022
Predict	預測階段	252
Preorder Traversal	前序走訪方式	022
Presentation Layer	呈現層	198
Priority Scheduling，PS	優先權排程	175
Privacy	隱私權	207
Private Key	私鑰	312
Process	程序	172
Program Development Tool	程式發展工具	170
Pseudocode	虛擬碼	041
Public Domain Software	公共財軟體	297
Public Key	公鑰	312
Push	放入	009
Q		
Quality Health	健康樂活	328
Queue	佇列	009
R		
Random-access Memory，RAM	記憶體	076
Rapid Elasticity	迅速且彈性	211
Ready State	預備狀態	173
Rear	尾端	010
Recommender Systems	推薦系統	257
Recursion	遞迴	051
Registered	合法註冊完成	292
Regression	迴歸	251
Reinforcement Learning	強化式學習	253
Replacement	替換	177
Requirements Analysis and Planning	需求分析與計劃	263
Resource Pooling	資源集區	212
Responsible AI Practices	負責任的人工智慧實踐	206
Reward	獎勵	254
Rewards Crowdfunding	回報型群眾募資	319
Root	根節點	013

Round-Robin Scheduling，RRS	輪詢排程	176
Routing Information Protocol	RIP 路由協定	200
Routing Table	路由表	200
Running State	執行狀態	173
Runtime Error	執行階段錯誤	078
S		
Science Club for Girls	科學女孩俱樂部	327
Search	搜尋	016
Security	安全性	207
Segment	區段	196
Selection	選擇結構	092
Selection Sort	選擇排序法	148
Self-calling	自我呼叫	051
Satellite News Gathering，SNG	SNG車	191
Semantic Error	語意錯誤	078
Sequential	循序結構	092
Sequential Search	循序搜尋法	156
Session Layer	會議層	198
Share Alike	相同方式分享	294
Shareware	共享軟體	297
Shortest Job First，SJF	最短工作優先排程	175
Shortest Path	最短路徑	045
Shortest Path Algorithm	最短路徑演算法	067
Sibling	兄弟	015
Simple Mail Transfer Protocol，SMTP	簡易郵件傳輸協定	196
Six Degrees of Separation	六度分隔理論	033
Smart Living	智慧生活	328
Software as a Service，SaaS	軟體即服務	209
Sort	排序	016
Stack	堆疊	009
Starvation	飢餓現象	175
State	狀態	254
Supervised Learning	監督式學習	205
Support	支持度	245
Sustainable Environment	永續環境	328
Syntax Error	語法錯誤	078
System Software	系統軟體	170
T		
TELNET	虛擬終端機模擬協定	196
Terminated	執行結束	172
Throughput	工作產出量	174
Time Complexity	時間複雜度	061
Top	最上層；頂端	009
Training	訓練階段	252
Transmission Control Protocol，TCP	傳輸控制協定	196
Transport Layer	傳輸層	196
Trie	字典樹	012
Turnaround Time	完成時間	174
Two Factor Authentication，2FA	雙重驗證	311
U		
Un-supervised Learning	非監督式學習	205
User Datagram Protocol，UDP	使用者資料協定	196
Utility	公用軟體	170
Utilization	使用率	174
V		
Variability	易變性	229
Variety	多樣性	228
Velocity	快速性；時效性	228
Veracity	真實性	229
Version Control	版本控制	271
Vertex	頂點；節點	030
Volume	大量性	228
W		
Wait State	等待狀態	173
Waiting Time	等待時間	174
Web Crawler	網路爬蟲	230
Weight	權重	030
Wide Area Network，WAN	廣域網路	187
Wireless Sensor Networks，WSN	無線感測網路	195
World Community Grid，WCG	世界社群網格	181

普通型高級中等學校　科技領域

資訊科技 全一冊 Python

編　著 者：蔡介銘

執行編輯：黃國賜

出　版 者：育達文化事業股份有限公司

地　　址：407056臺中市西屯區何厝東二街81號

電　　話：(04) 2316-0117

傳　　眞：(04) 2316-6090

劃撥帳號：22605056 育達文化事業股份有限公司

書籍編號：882742A (112/1)

出版日期：中華民國112年8月　初版

◎ 版權所有・侵害必究

※若發現本書有缺頁、倒裝、漏印或嚴重汙損等情形，請接受本公司最誠摯的歉意，並請您將本書包妥寄回，本公司將盡速爲您處理，謝謝您！

讀者專線：(04) 2316-0117

或E-mail：yuida.yuida@msa.hinet.net

※本公司已盡力處理著作權事宜，如偶有疏漏之處，還請著作權持有者與本公司接洽，謝謝！