Computer Programming and Practice (I)
計算機程式設計與實習(一)

110年上學期
國立臺南大學 電機工程系
梁家銘

1 電腦、網際網路與全球資訊網簡介

1.1	簡介
1.2	硬體與軟體
1.3	資料架構
1.4	機器語言、組合語言、高階語言
1.5	C程式語言
1.6	C標準函式庫
1.7	C++與其他以C為基礎的語言
1.8	物件技術
1.9	典型的C開發環境
1.10	應用於Windows、Linux和Mac OS X的試行
1.11	作業系統
1.12	網際網路與全球資訊網
1.13	一些軟體開發術語
1.14	用資訊科技持續跟上時代

1.1 簡介

- 歡迎來到C與C++！
- C語言是一種簡潔但功能強大的程式語言
  - 適合有部分程式設計經驗(或完全沒有程式經驗)的技術人員
  - 也適合資深程式設計師，建立大量的軟體系統(具有豐富函式庫、進階用法)。
  - 對這些人來說，C程式是適合入門的學習工具。

- 電腦能執行許多特別的工作 (如：應用、服務、遊戲)。
  - 此門課，可以學會如何命令電腦來執行這些工作。
  - 電腦（通常指硬體，hardware)，其由軟體(software)所控制（軟體就是人撰寫的指令，其可讓電腦執行動作[action]並做判斷[decision]）。

1.2 硬體與軟體

- 電腦透過**電腦程式(computer programs)**的一組指令控制來處理**資料(data)**。
  - 這些電腦程式命令電腦去執行一連串、有次序的動作
  - 這些動作皆由電腦程式設計者(computer programmers)預先設計並指定好的。

- 電腦由各種硬體(hardware)裝置所組成。
  - 軟硬體技術快速發展，電腦硬體價格急速地下降。
  - 幾十年前，電腦體積龐大到塞滿整個大房間，費用高達數百萬美金的電腦
  - 現今竟可以放在一個比指甲還小的矽晶片表面上，單價只要幾塊美金。
  - Note: 軟體的發展性，將逐漸比硬體更有價值(跨領域人材非常搶手)。

1.2.1 摩爾定律(Moore’s Law)

- 過去數十年，硬體規格一再地快速上升。
  - 每隔一兩年，電腦運算能力會增加近一倍，且仍廉價。
  - 這個趨勢稱作**摩爾定律**（由Intel的共同創辦人Gordon Moore來命名）。
- 摩爾定律在某些事物上特別地明顯
  - 如：電腦記憶體容量、儲存裝置容量，及處理器速度。
  - 同樣現象也發生在通訊領域，硬體價格快速下降，通訊頻寬急速上升。

(如：以前CPU 600MHz， 現在CPU 4GHz/5Ghz或更高)

提外話：未來，NVIDIA 創始黃仁勳（Jensen Huang）名字命名的定律——「黃氏定律（Huang's Law）」對AI 性能的提升作出預測，預測GPU 將推動AI 性能實現逐年翻倍

1.2.2 電腦的架構

- 每部電腦可分成不同**邏輯單元(logical units)**

邏輯單元	描述
輸入單元 (input unit)	用於接收資訊：從輸入裝置（input devices）取得資訊（資料與電腦程式），並將此資訊放在其它裝置上以進行處理。(如：鍵盤)
輸出單元 (output unit)	用於輸出資訊：將處理過的資訊，送到不同的輸出裝置（output devices），讓這些資訊能夠在電腦之外使用。(如：螢幕)

邏輯單元	描述
記憶單元 (memory unit)	電腦中的一個存取快速、容量較低的「暫存」區域。 它將接收到的資訊暫時保存起來，需要時可馬上運用。記憶體中的資訊是「揮發的」。記憶體單元又稱為記憶體（memory）、、主記憶體（primary memory）或RAM（隨機存取記憶體，Random Access memory）。
算術和邏輯單元 (arithmetic and logic unit，ALU)	這是「計算、生產、製造」的部分，它負責執行如加、減、乘、除等計算。它所包含的判斷機制能讓電腦做運算。

邏輯單元	描述
中央處理單元 (central unit，CPU)	電腦「執行管控」的區域，負責協調監督其它區域的作業。需要將資訊讀入記憶單元時，CPU會通知輸入單元；需要將記憶單元的資訊進行計算處理時，CPU就會通知ALU加以處理；需要將記憶單元的資訊傳送到某個輸出裝置時，CPU就會通知輸出單元。
輔助儲存單元 (secondary unit)	電腦長期、高容量的「儲存倉庫」。尚未被其它單元使用的程式或資料，通常都放在輔助儲存單元。因此，儲存在輔助儲存裝置中的資料為「永續的（persistent）」，當電腦的電源關閉時，這些資料還是會保留著。輔助儲存單元的存取速度比主要儲存單元慢，但儲存成本低。

1.3 資料架構

- 電腦處理的資料單元形成了**資料架構(data hierarchy)**，資料架構隨著從位元至字元、欄位等方式，在結構上會變成越來越大且複雜。

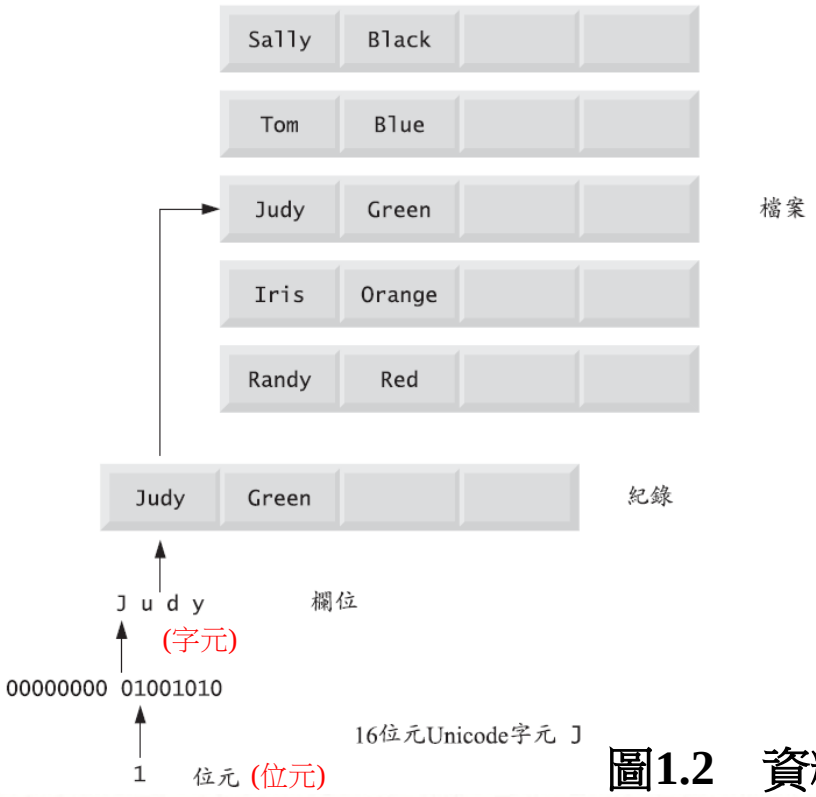


圖1.2 資料架構階層描述

Unit	Bytes	Which is approximately
1 kilobyte (KB)	1024 bytes	10^3 (1024 bytes exactly)
1 megabyte (MB)	1024 kilobytes	10^6 (1,000,000 bytes)
1 gigabyte (GB)	1024 megabytes	10^9 (1,000,000,000 bytes)
1 terabyte (TB)	1024 gigabytes	10^{12} (1,000,000,000,000 bytes)
1 petabyte (PB)	1024 terabytes	10^{15} (1,000,000,000,000,000 bytes)
1 exabyte (EB)	1024 petabytes	10^{18} (1,000,000,000,000,000,000 bytes)
1 zettabyte (ZB)	1024 exabytes	10^{21} (1,000,000,000,000,000,000,000 bytes)

圖1.3 位元組的度量衡

測驗/解答

1.4 機器語言、組合語言、高階語言

機器語言(machine language)

- 任何電腦僅能直接瞭解其自身的**機器語言(machine language)**。機器語言一般來說是以數字組成(最後簡化成1與0)
- 機器語言與機器相依。

組合語言與組譯器

- 編寫機器語言對程式設計師非常不直覺。
- 因此，進而使用類似英語縮寫，來表達基本運算，即為**組合語言（assembly languages）**的基礎
- **組譯器(assembler)**是轉譯程式(translator program)可將組合語言轉成機器語言，讓電腦看得懂。

測驗/解答

高階語言與編譯器

- 為了加速程式設計開發，於是發展了**高階語言(high-level languages)** (Ex: X = Y+ Z;)
- 只需單一敘述(statement)，就能完成許多任務。
- **編譯器(compiler)**為轉譯程式(translator program)，可將高階語言程式轉成機器語言。

直譯器

- 由於將高階語言編譯成機器語言可能要花不少時間。
- **直譯器**(Interpreter)可直接執行高階語言程式，省去編譯的時間，但執行速度比編譯完成的程式來得慢。

測驗/解答

1.5 C程式語言

(比如：英文來自於拉丁語系)

- C語言是由B語言和BCPL(括弧語言)這兩種語言進化而成。
- C語言是在1972年由貝爾實驗室的Dennis Ritchie由B語言所發展出來的。
- C語言最初以開發 UNIX 作業系統而聞名

- 目前重要作業系統，皆以C和C++為基礎撰寫而成，如：Linux、部分微軟視窗、Google Android及Apple的OS X，皆以C作為基礎發展。
- C語言應用方面，還包含：嵌入式系統、即時系統、通訊系統 (ex: WIFI AP)。

測驗/解答

1.6 C標準函式庫

(功能/工具)

- C程式由許多**函式**(**functions**)的模組或是片段程式所構成。
- 大多數的C程式設計師，都會利用一套現成的函式，稱為**C標準函式庫**。

(現成工具/功能)

- **軟體再用：**以區塊導向編寫程式，避免重新編寫。
- 當使用C語言撰寫程式，將會使用到下列構件：
  - C標準函式庫、自己撰寫的函式、其他人撰寫的函式。

自行建立函式

(Ex: 比較x、y大小)

- 優點：清楚瞭解其實際運作方式。
- 缺點：須花時間來設計、發展、偵錯新函式，並對新函式進行效能調整。

測驗/解答

1.7 C++與其他以C為基礎的語言

(發展新技術)

- C++是由貝爾實驗室的Bjarne Stroustrup所發展出來的。
- C++根源於C語言，而且增加了許多功能，使得C語言變得更好。

(比如：建構一台汽車，由車身、輪子、方向盤等組成)

- 最重要的是它提供了**物件導向程式設計(object-oriented programming)** 的功能。
- 目前，C語言為基礎的熱門程式語言
- 包含： Objective-C、 Java、C#、Python、Swift

(有興趣的同學，可以進一步的去查查看)

測驗/解答

1.9 典型的C開發環境

- C系統通常由幾個部分組成：
  1. 程式開發環境 (如：IDE (Integrated development environment))
  2. 程式語言
  3. C標準函式庫

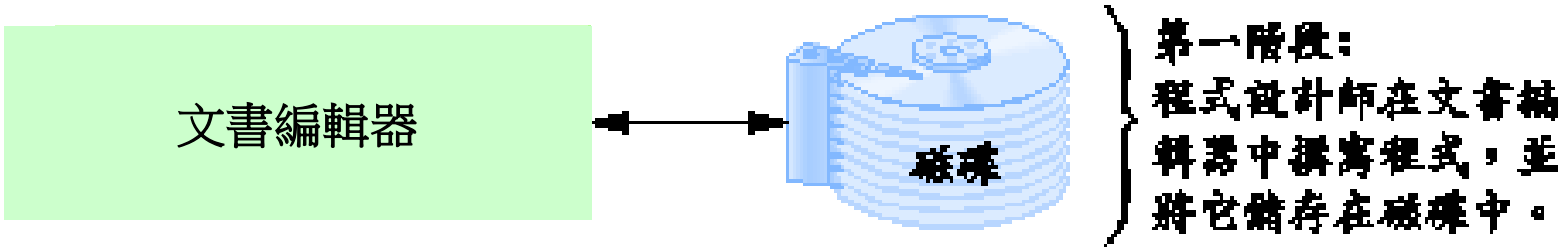
- C程式在執行前，通常必須經過6個階段：
  1. 編輯程式(edit)
  2. 前置處理(preprocess) (引進標頭檔)
  3. 編譯 (compile) (檢查語法)
  4. 連結 (link) (連結函式庫、物件)
  5. 載入(load)
  6. 執行 (execute)

(後面會詳細介紹)

測驗/解答

1.9.1 第一階段：建立程式

- 第一個階段是編輯檔案，可以利用**文書編輯器(editor program)**來完成。



1.9.2 下一階段：前置處理與編譯C++程式

- 編輯完程成，將進行**編譯(compile)**程式。
- 編譯器會將C程式碼轉譯成機器碼(也稱為**目的碼，object code**)。

- **前置處理(preprocessor)**程式會在編譯器轉譯階段之前執行。
- **C前置處理器(C preprocessor)**會按照一種叫作**「前置處理指令」(preprocessor directive)** 的特殊指令進行動作，該指令表示編譯前要對程式執行某些操作。

- 編譯錯誤(或編譯期錯誤)
- 編譯器轉譯C語言成為機器語言時，當無法辨識敘述時，即發生語法錯誤，也稱為編譯錯誤或編譯期錯誤。

測驗/解答

1.9.3 連結(linking)

- C程式中，有些參照(reference)會指到在別處定義的函式與資料。
  - **連結器(linker)**會將目的碼與尚未加入之函式連接起來，以產生**可執行的映像檔(executable image)**

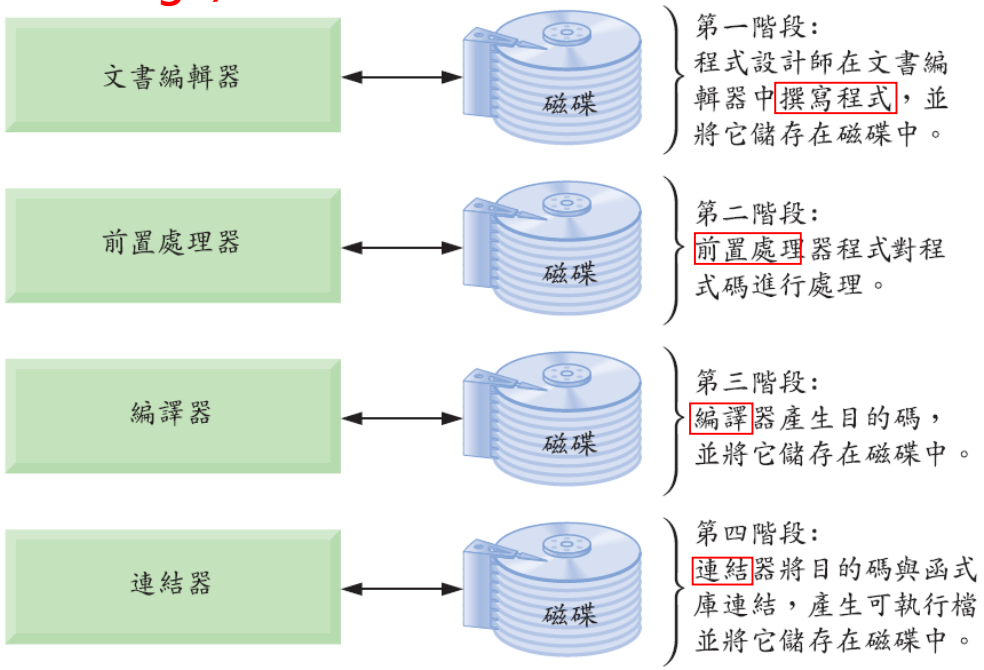


圖1.7 典型的C開發環境(1/3)

測驗/解答

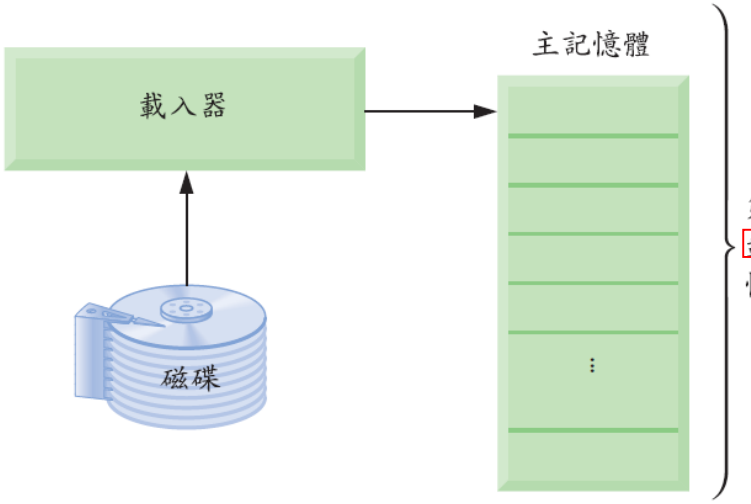


圖1.7　典型的C開發環境(2/3)

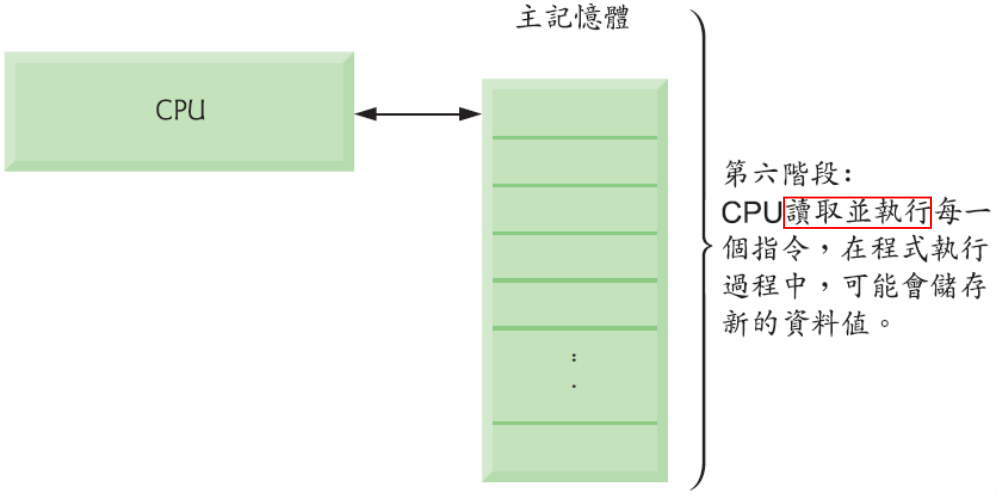


圖1.7　典型的C開發環境(3/3)

測驗/解答

1.9.4 載入 (loading)

- 程式執行前，必須先被放入記憶體中。
- 由**載入器**負責把可執行的映像檔從磁碟搬到記憶體中。
- 程式所用到的共享函式庫中其它元件，也須一併載入。

1.9.5 執行(executes)

- 電腦在CPU控制下，每次執行一個指令的方式開始執行程式。

1.9.6 執行時可能會發生的問題

- 程式在測試時不一定會成功。此時，電腦可能會顯示一段錯誤訊息。
- 若發生這樣的情形，就要回到編輯階段做些必要修正。

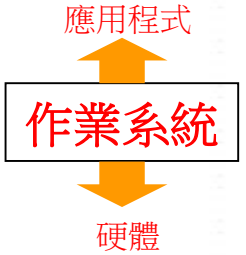
1.9.7 標準輸入、標準輸出和標準錯誤串流

- C中大部份程式都會輸入和/或輸出資料。
- 大部分C函式都從**stdin (標準輸入串流，standard input stream)** 輸入資料。
- 大部分C函式都從**stdout (標準輸出串流，standard output stream)** 輸出資料。

測驗/解答

1.11 作業系統 (Operating System)

- **作業系統**是一個提供使用者、應用程式開發者、系統管理者方便使用的環境之軟體系統。
- 作業系統提供應用程式更安全、更有效率及能與其他應用程式同時(平行)執行。
- 作業系統的主要元件部分，稱作為**核心(kernel)**。



- 目前主流的桌上型作業系統為：Linux、Windows和Mac OS X.
- 行動式作業系統，應用於智慧型手機、平板電腦為：Google Android、Apple iOS、Windows Phone和BlackBerry OS

測驗/解答

1.12 網際網路(Internet)與全球資訊網(www)

- 1960年代，ARPA（美國國防部的先進研究計畫署）資助大學院校與研究機構，建構電腦系統間的網路藍圖。
- 當大多數人透過電話線以110 bits/s連上網路時，這些電腦以50,000 bits/s速度與通信線路連接。
- ARPA持續地發展出**ARPANET**，最終演變成為現在的**網際網路 (Internet)**。
- 透過ARPANET進行通信的協定，稱為**傳輸控制協定（TCP）**。
- TCP確保訊息能正確地從發送端傳送到接收端，完好無損正確接收 。 (錯誤會有檢查、重傳機制)

測驗/解答

1.12.1 網際網路：網路的網路

- ARPANET的主要目標是讓多使用者，透過相同通訊路徑同時傳送與接收資訊。
- 此網路運作技術被稱為封包交換，其中以小型捆包傳送的數位資料稱為封包。 (編號、位址、驗證)
- 封包包含了位址、錯誤控制和序列資訊：
- 位址資訊 - 讓封包路由至其目的地
- 序列資訊 - 協助重組封包至原始順序重現給接收者。

1.12.2 全球資訊網：讓網際網路使用者友善

(ex: Facebook, Instagram, Twitter, etc.)

- 全球資訊網（World Wide Web，簡稱「網站（web）」）是與網際網路相關之硬體和軟體的集合，允許電腦使用者查找幾乎是各種主題、以多媒體為基礎的檔案，具有文本、圖形、動畫、聲音、視訊等各種組合的檔案。
- 基於「超連結(hyper link)」，形成超文件標記語言（HTML）
- 運用超文本傳輸協定（HTTP）為通訊協定，為World Wide Web系統的骨幹。

測驗/解答

1.12.3 常見網路服務

網路服務來源	使用方法
Google地圖	地圖服務
Twitter	微網誌
YouTube	視頻搜尋
Facebook	社交網路
Instagram	照片分享
Foursqiare	行動報到
LinkedIn	商務社交網路

網路服務來源	使用方法
Groupon	社交商務
Netflix	電影租借
eBay	網際網路商務
Wikiprdia	協作百科全書
PayPal	付款
Last.fm	網際網路收音機
Amazon eCommerce	書籍及其他商品購物

網路服務來源	使用方法
Salesforce.com	顧客關係管理
Skype	網際網路電話
Microsoft Bing	搜尋

網路服務來源	使用方法
Flickr	照片分享
Yahoo Search	搜尋
Weatherbug	天氣

測驗/解答

1.15　用資訊科技持續跟上時代

- 讀者可以參考下列網址找到越來越多的網路與網站相關資源中心 www.deitel.com/resourcecenters.html。

測驗/解答