

# 程式設計 (**Programming**)

真理大學 資訊工程系 吳汶涓老師

CH01

電腦、網際網路與  
全球資訊網簡介



# 資工畢業出路

職務名稱	職務說明
財務工程師	運用資訊網路、資料庫、經濟能力從事金融業金融分析工作
系統分析師	將客戶需求內化為技術元件，並與程式設計師溝通，進而完成專案需求
程式設計師	資訊產品研發與撰寫，主要擔任程式Design與Coding之角色
網路管理工程師	主要擔任硬體設備之架設，與網站伺服器之管理
資訊安全工程師	公司網路資訊安全之維護、管理與技術支援
網頁設計師	規劃執行與設計網頁，使網頁更能親近使用者介面
多媒體設計師	因應多媒體的多元發展，主要設計具互動性以及豐富多元的傳達效果
網站行銷企劃	結合企業整體資源於網站，執行網際網路上的行銷活動
IC測試工程師	測試完工之IC是否正確符合產品規格之需求
IC設計工程師	將企劃之IC產品規格轉換成技術功能規格之工程師
業務工程師	負責做企業、產品簡介，以及開發客戶與產品報價、談判等工作

- 徵求軟體工程師
- 工作條件:大專以上
  - 一、二年系統設計經驗
  - 精通C語言

# 何謂電腦？



## ■ 電腦

- 能以人類數十億倍的速度，執行計算和邏輯判斷的裝置
- 受到**電腦程式 (computer programs)** 的一組指令控制來處理資料 (data)

## ■ Hardware

- 電腦是由各種裝置所組成的：鍵盤、螢幕、滑鼠、磁碟機、記憶體、CD-ROM以及處理器等等

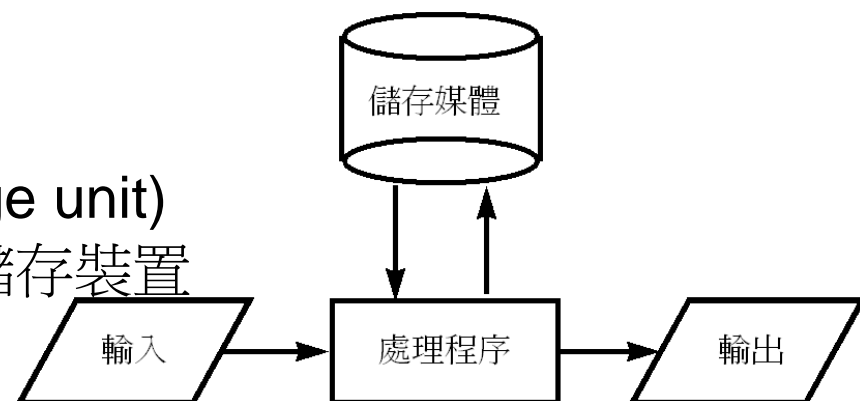
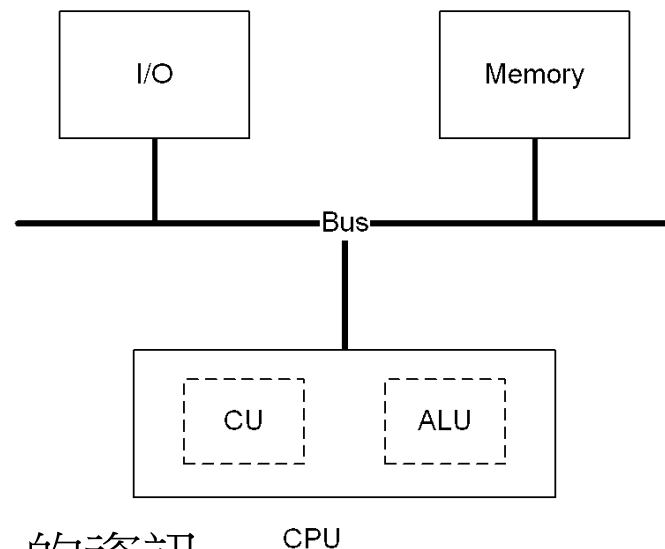
## ■ Software

- 在電腦上執行的**程式 (applications)**

# 電腦的架構

## ■ 六個邏輯單元

1. 輸入 (input unit)
2. 輸出 (output unit)
3. 記憶 (memory unit)
  - 快速存取、容量較低、儲存輸入的資訊
4. 算術和邏輯 (ALU)
  - 執行算術運算以及邏輯判斷
5. 中央處理 (CPU)
  - 負責監督和協調的作業
  - 多核心處理器
6. 輔助儲存 (secondary storage unit)
  - 便宜、長期、高容量的儲存裝置

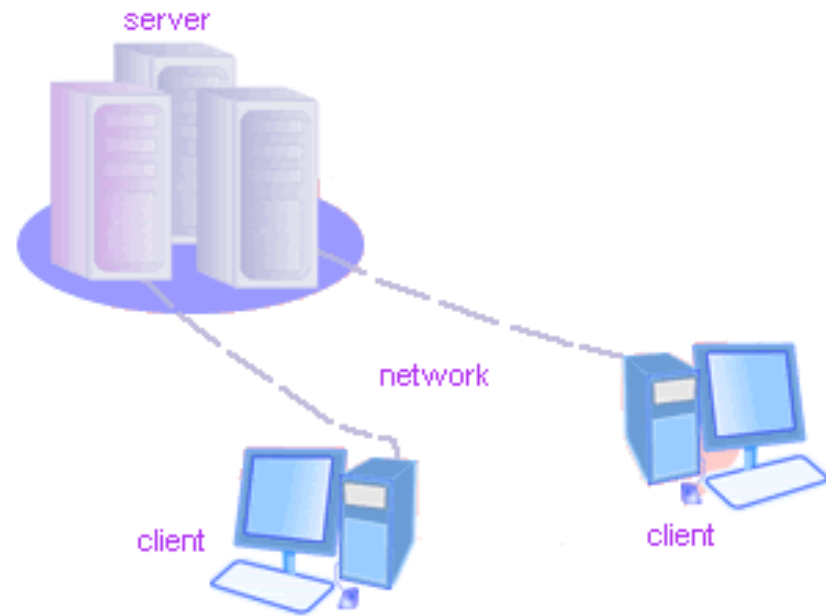


# 早期的電腦作業系統

- 批次處理 (batch processing)
  - 一次只能執行一項工作
- 作業系統 (operating systems)
  - 提升電腦使用的便利性
  - 管理工作之間的轉換
  - 增加總產量
- 多工 (multitasking)
  - 讓多個工作分享電腦的資源
- 分時
  - 執行某個使用者的一小部分工作，然後就移去服務下一位使用者

# 個人、分散式及用戶／伺服器環境

- Personal Computing
  - 價格較低，適合個人使用
- Distributed Computing
  - 將電腦計算分散到網路上
- Client / Server
  - 在檔案伺服器和用戶之間分享資訊
  - **FTP: file transfer protocol**  
(**FileZilla** 軟體)



# 電腦軟體

- 電腦軟體是提供電腦指令的程式，分系統軟體與應用軟體兩大部分
  - 系統軟體: 用來開發或執行其他的軟體 (ex.: 作業系統)
  - **應用軟體**: 為了解決特殊問題或提供特別服務的程式，如上網的IE、繪圖的AutoCAD等等
- 程式
  - 一系列**指令(Instructions)**
  - 特定的字彙、語法來規範
    - 語法(Syntax)
    - 語意(Semantics)



# 機器語言、組合語言以及高階語言

## 三種類型的程式語言

### 1. Machine Language

- 一連串數字組成機器特有指令 (電腦的母語)
- 範例: +1300042774  
+1400593419  
+1200274027

### 2. Assembly Language

- 類似英文的縮寫字代表基本操作 (經由「組譯器」轉譯)
- 範例: LOAD BASEPAY  
ADD OVERPAY  
STORE GROSSPAY

### 3. High-Level Language

如: C, C++, .NET, Java...

- 似於日常英文的程式語言 和一些數學符號 (經由「編譯器」轉譯)
- 範例: grossPay = basePay + overTimePay

# C的沿革

## ■ C

- Dennis Ritchie所開發，起源是B和BCPL這兩種語言
- 用來開發UNIX作業系統
- 與電腦硬體無關 (可攜性, portable)
- ANSI C (1989年建立標準)

## ■ C標準函式庫 (standard library)

- 可以利用C函式庫所提供的函式 (Ex.: **stdio.h**, **stdlib.h**, ...)
- 程式設計師可以自行建立函式 (function)
- 避免重寫軟體 (**software reusability**)

# 其他語言

## ■ C++

- ❑ C++根源於C語言，由貝爾實驗室的Bjarne Stroustrup所發展出來
- ❑ 增強了C語言，並提供物件導向能力 (元件, Component)
- ❑ 工業界和學術界使用的主要程式語言

## ■ Java

- ❑ Sun所開發
- ❑ 產生具有動態且互動式內容的網頁
- ❑ 目前用來開發大型的商用軟體
- ❑ 提供消費性電子商品的應用程式 (如行動電話，呼叫器和個人數位助理)

# 其他語言

- Fortran, COBOL, Pascal, Ada
- BASIC
  - 1960年代由Dartmouth學院的John Kemeny 和 Thomas Kurtz教授發展的，用來撰寫簡單的程式
- Visual Basic
  - 1990年由微軟公司採用來發展Microsoft Windows應用程式
- Visual Basic, Visual C++, and Visual C#
  - 微軟為新的. NET programming程式設計平台設計的

# 軟體工程趨勢：物件技術

## ■ 物件 (Object)

- 模擬真實世界事物的可重複使用軟體元件 (稱為**類別, class**)
- 有意義的軟體元件
  - 日期物件、時間物件、薪資物件、收據物件、聲音物件、影像物件、檔案物件、記錄物件等等
  - 任何名詞都可以表示成一個物件
- 可重複使用
- 比程序式程式設計更易於了解，維護、修改及偵錯
- 有助於模組化

# C開發環境

編譯器

連結器

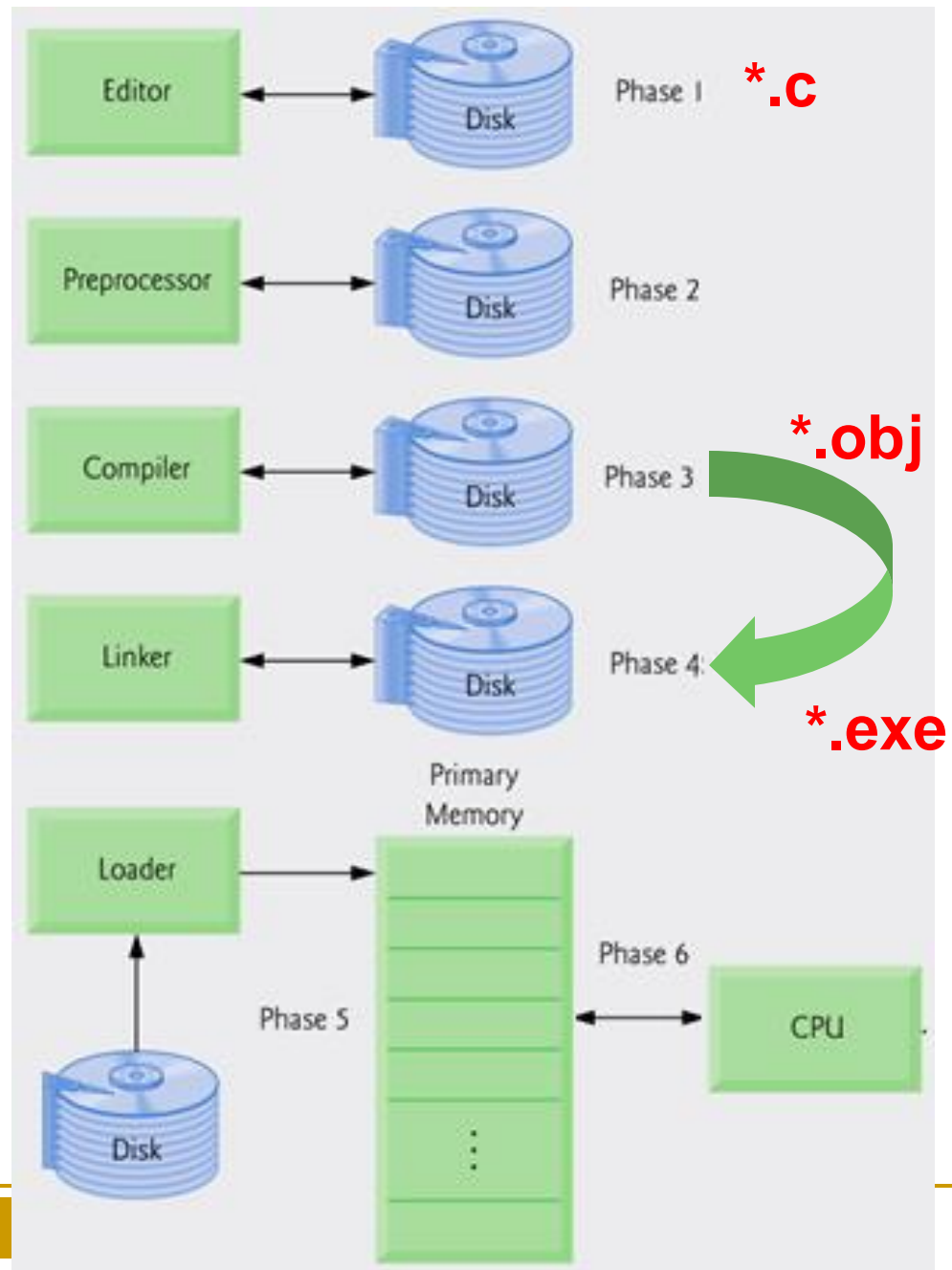


Fig. 1.1 典型的C開發環境

# C開發環境

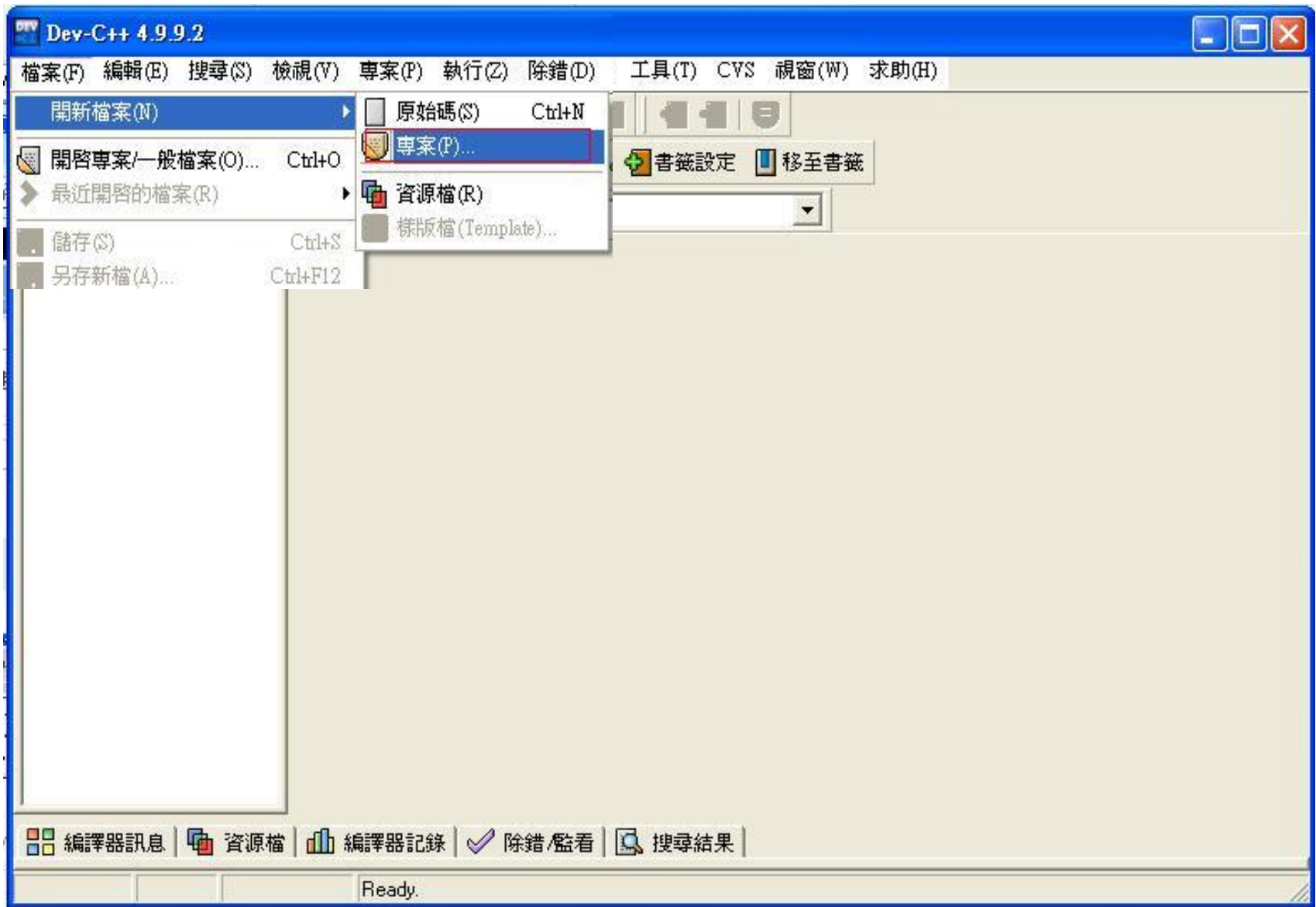
## ■ UNIX/Linux系統下

- gcc

## ■ IDE

- Turbo C / C++
- Borland C++ / Borland C++ Builder
- Visual C++
- **Dev C++**
- ...







# C語言程式結構

```
# include <stdio.h>
```

```
# include <stdlib.h>
```

```
int main(void)  /* 主程式*/
```

```
{
```

```
    return 0;
```

```
}
```