

第1章、系統軟體

作者：陳鍾誠

旗標出版社



第1章、系統軟體

- 1.1 何謂系統程式？
- 1.2 系統程式與系統軟體
- 1.3 本書的章節架構
- 1.4 實務案例 (一)：Dev C++開發環境
- 1.5 實務案例 (二)：GNU 開發工具

1.1 何謂系統程式？

- System Software (系統軟體)
- System Programming (系統程式設計)

System Software (系統軟體)

- 何謂系統軟體？
 - 設計給程式設計師使用的軟體, 就被稱為系統軟體
 - 設計給一般大眾使用的軟體, 則稱為應用軟體。

系統軟體 v.s 應用軟體

- 系統軟體

- 組譯器
- 載入器
- 連結器
- 巨集處理器
- 編譯器
- 直譯器
- 虛擬機

- 應用軟體

- 試算表
 - Excel
- 排版軟體
 - Word
- 瀏覽器
 - Internet Explorer
 - Firefox
 - Chrome

系統程式設計 (System Programming)

- 定義：
 - 系統相關的程式設計技術，其中的系統指的是作業系統或電腦系統。
- 作業系統層次的程式設計
 - Linux 系統程式
 - Windows 系統程式

作業系統相關的程式設計主題

- 行程管理
- 執行緒
- 行程通訊
- 並行控制
- 記憶體管理
- 檔案輸出入
- 驅動程式

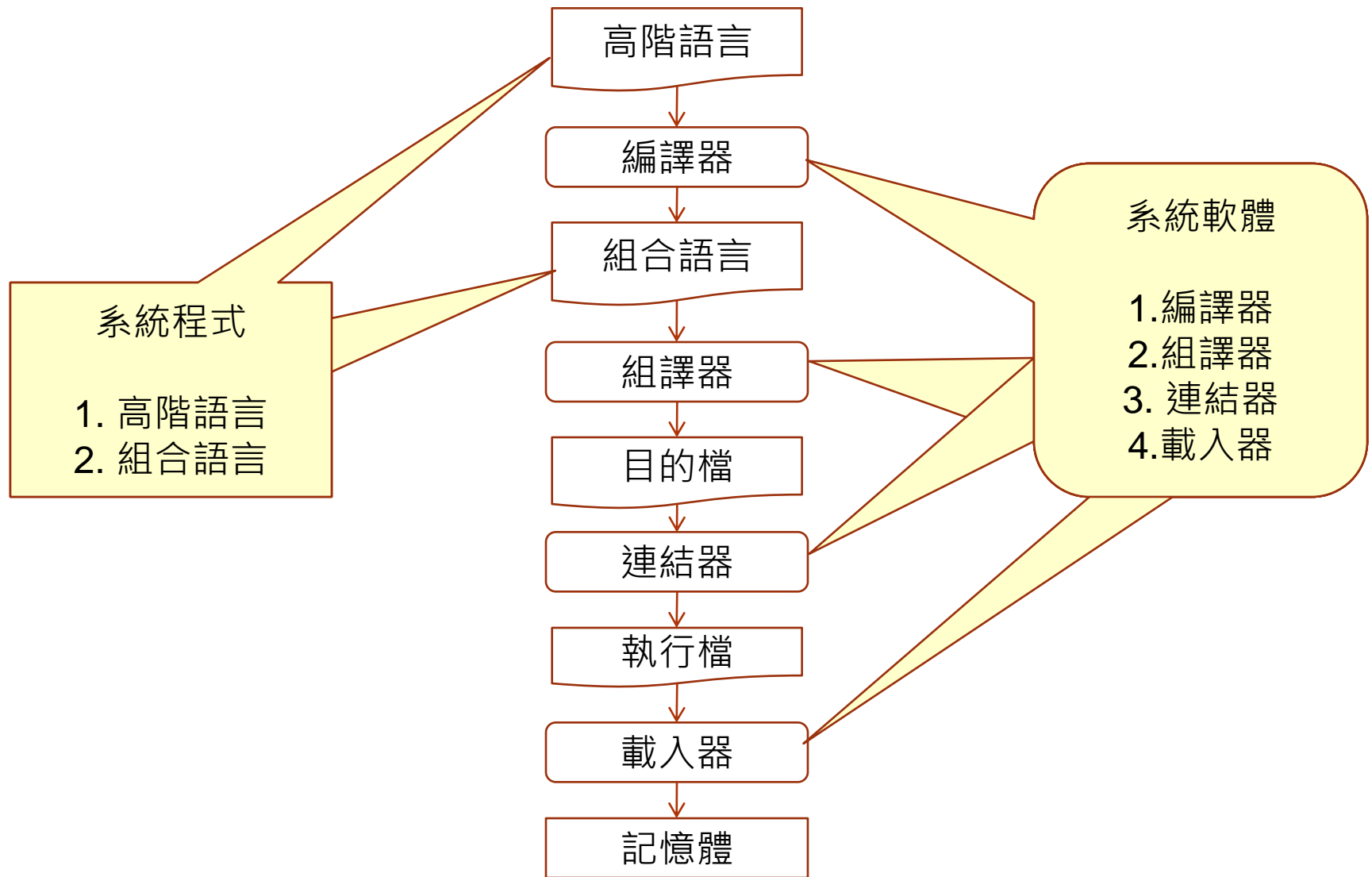
電腦系統相關的程式設計主題

- 組合語言
- C 語言
- 嵌入式系統

1.2 系統程式與系統軟體

- 高階語言 → 編譯器 → 組合語言
- 組合語言 → 組譯器 → 目的碼
- 目的碼 → 連結器 → 可執行檔
- 可執行檔 → 載入器 → 記憶體，開始執行

圖 1.1、程式的編譯、組譯、連結、載入之流程



從機器語言到組合語言

範例 1 : 2 進位串列形式的指令

```
00010011000100100011000000000000
```

範例 2 : 2 進位分組形式的指令

```
0001 0011 0001 0010 0011 0000 0000 0000
```

範例 3 : 16 進位分組形式的指令

```
13 12 30 00
```

範例 4 : 組合語言

```
ADD R1, R2, R3
```

範例 5 : 組合語言與機器指令的對應關係

```
ADD R1, R2, R3  
13 1 2 3 000
```

目的碼
(機器語言)



組譯器



組合語言

各種組合語言的指令

指令 1 : 加法指令

```
ADD R1, R2, R3
```

指令 2 : 減法指令

```
SUB R1, R2, R3
```

指令 3 : 目標後置的指令寫法

```
ADD R2, R3, R1
```

指令 4 : 以記憶體位址當參數

```
LD R1, [300]
```

指令 5 : 直接存取記憶體

```
ADD R1, R2, [300]
```

指令 6 : 使用變數

```
ADD R1, R2, COUNT
```

指令 7 : 減少參數個數

```
ADD R1, COUNT
```

指令 8 : 縮短指令長度

```
ADD COUNT
```

最近 (2000)
(精簡指令CPU)



中期 (1990)
(複雜指令CPU)

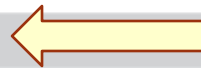


早期 (1980)
(8,16 位元 CPU)

從組合語言到高階語言

C 語言

```
x = a + 3 * b - c*d;
```



組合語言

```
LDI R1, 3  
LD R2, b  
MUL R3, R1, R2  
LD R1, c  
LD R2, d  
MUL R4, R1, R2  
LD R1, a  
ADD R2, R1, R3  
SUB R2, R2, R4  
ST R2, x
```

範例 1.1 C 語言與組合語言的對應關係

C 語言

- 1972 年貝爾實驗室所發展出來的 C 語言
 - Ken Thompson 與 Dennis Ritchie (合稱 K & R)
 - K &R 用 C 語言設計出 UNIX 作業系統
 - C 語言特別適合用來撰寫作業系統等低階程式，因此是當今系統程式的主要語言。
 - C 語言可以進行記憶體映射輸出入，這讓 C 語言特別適合用來撰寫嵌入式的系統程式

1.3 本書的章節架構

- 第 1 章 系統軟體
 - 第 2 章 電腦的硬體結構
 - 第 3 章 組合語言
 - 第 4 章 組譯器
 - 第 5 章 連結與載入
 - 第 6 章 巨集處理器
 - 第 7 章 高階語言
 - 第 8 章 編譯器
 - 第 9 章 虛擬機器
 - 第 10 章 作業系統
 - 第 11 章 嵌入式系統
 - 第 12 章 系統軟體實作
-
- 導論
- 組合語言
- 高階語言
- 執行平台
- 實作『組譯器、編譯器、虛擬機器』
- Detailed description: The diagram uses orange brackets to group chapters. A single bracket groups Chapter 1 as '導論'. A larger bracket groups Chapters 2-5 as '組合語言'. Another bracket groups Chapters 6-8 as '高階語言'. A fourth bracket groups Chapters 9-11 as '執行平台'. A final bracket groups Chapter 12 as '實作『組譯器、編譯器、虛擬機器』'.

本書的章節導引圖

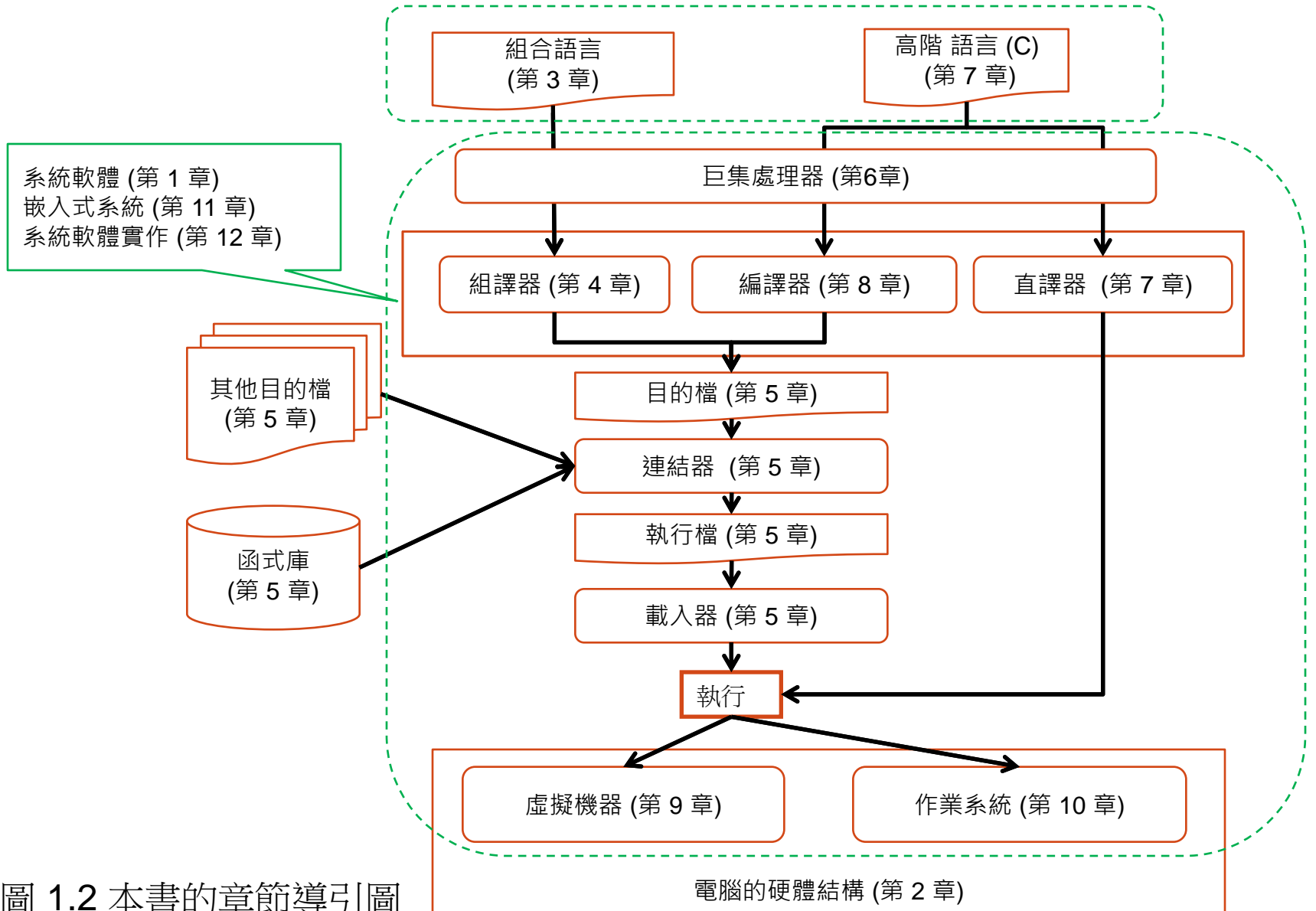
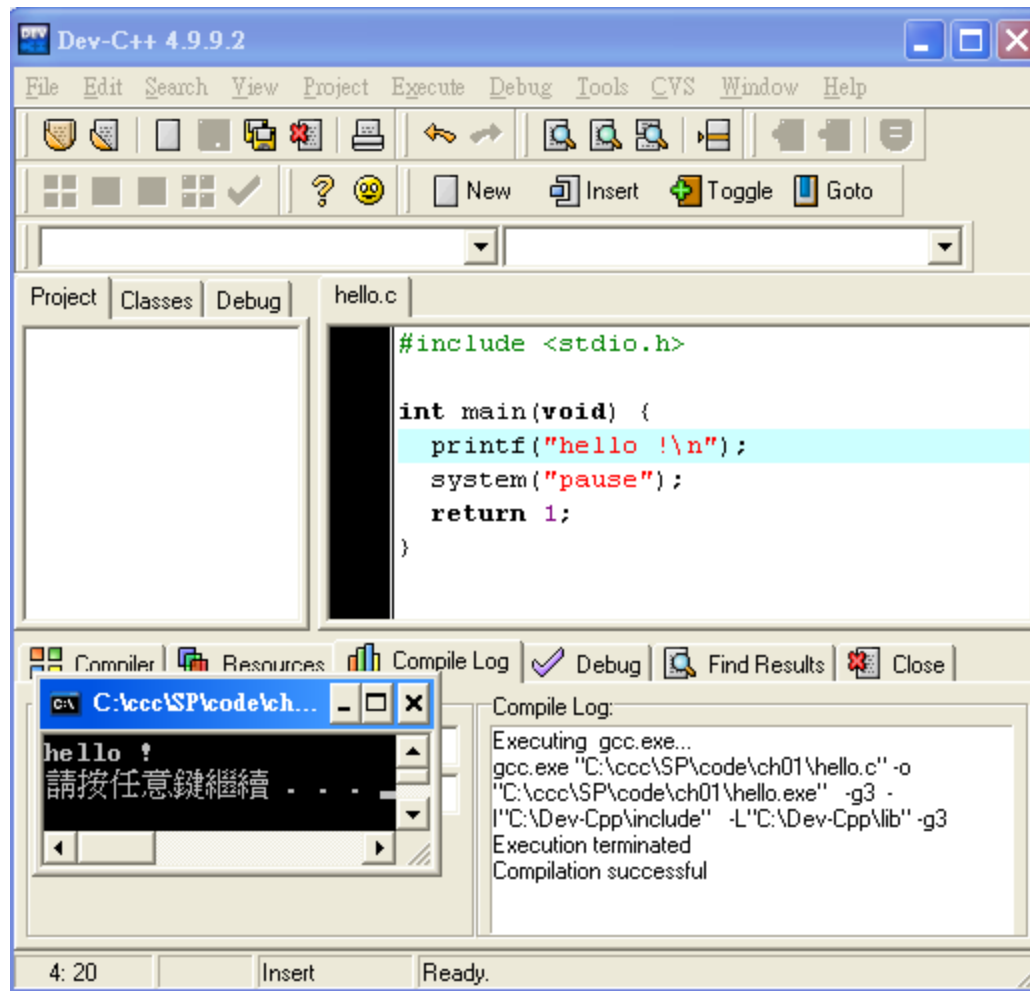


圖 1.2 本書的章節導引圖

1.4 實務案例 (一) : Dev C++開發環境

- Dev C++ :
 - Bloodshed Software 公司所設計的免費的開發工具
 - 下載點 : <http://www.bloodshed.net/devcpp.html>
 - Dev C++ 使用 gcc 作為預設的編譯器。

圖 1.3 單一程式檔的Dev C++ 編譯執行畫面



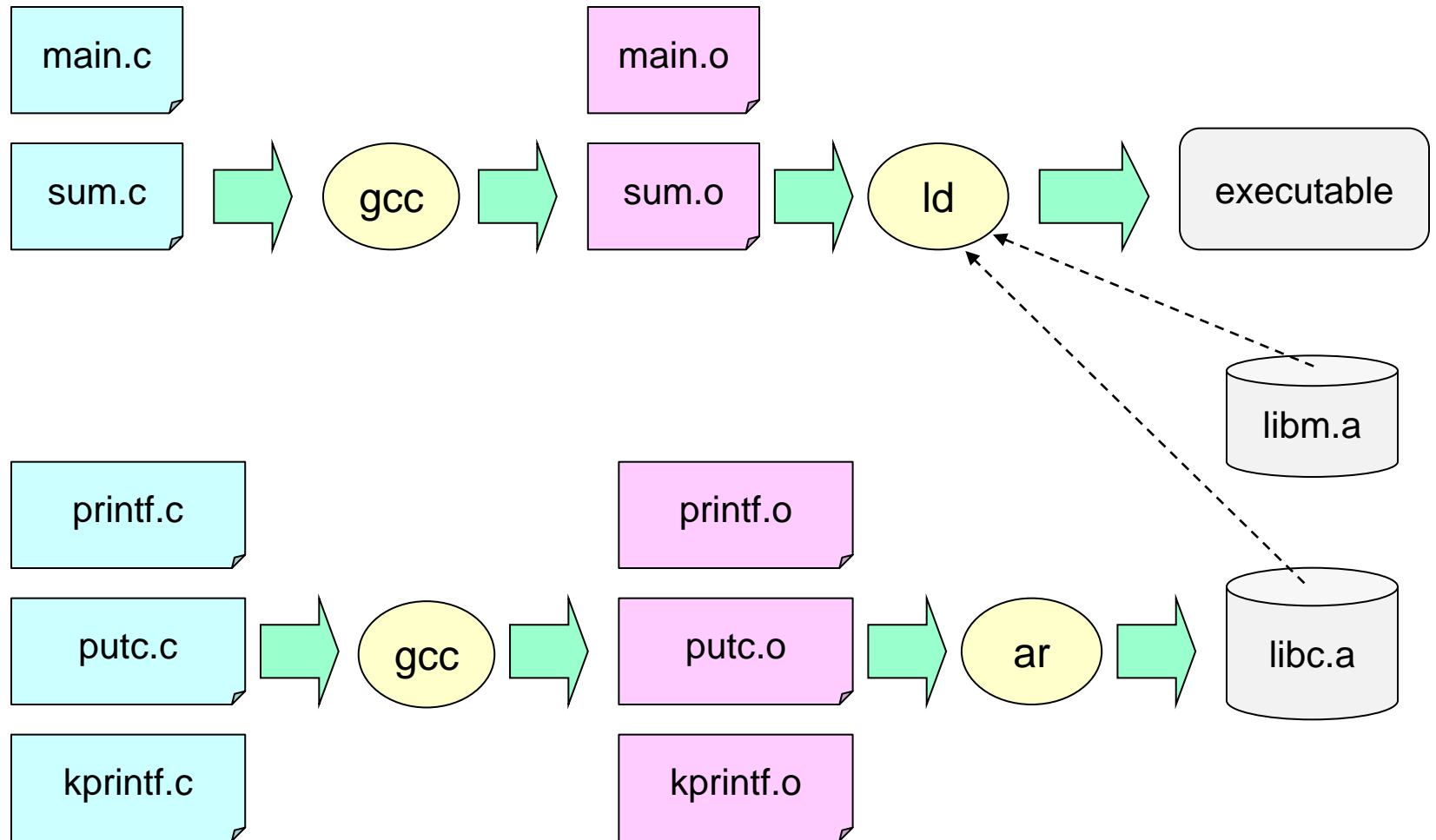
1.5 實務案例 (二) : GNU 開發工具

- GNU 開發工具
 - 由 GNU 組織所設計
 - 包含
 - gcc 編譯器
 - as 組譯器
 - ld 連結器
 - make 專案建置工具
 - Linux 作業系統也是用 GNU 工具所開發完成的

如何使用 GNU 工具

- 在 Linux 當中
 - 預設就包含了 GNU 工具
- 在 MS. Windows 中
 - 安裝 Dev C++ 後就會有 GNU 工具，請參考附錄 D 的安裝方法。
 - 安裝 Cygwin 也可以使用 GNU 工具，請參考附錄 E 的安裝方法。

圖 1.4 GNU工具使用的基本流程



C 語言程式範例

▶範例 1.2 程式 main.c 與 sum.c

C 語言主程式 (main.c)

```
#include <stdio.h>
int main(void) {
    int sum1 = sum(10);
    printf("sum=%d\n", sum1);
    system("pause");
    return 1;
}
```

C 語言函數 (sum.c)

```
int sum(int n) {
    int s=0;
    int i;
    for (i=1; i<=n;i++) {
        s = s + i;
    }
    return s;
}
```

使用 gcc 編譯 C 語言程式

圖 1.5 利用 gcc 同時編譯 main.c 與 sum.c 並輸出執行檔

```
C:\ch01>gcc sum.c main.c -o sum
```

```
C:\ch01>gcc sum.c main.c -o sum
```

```
C:\ch01>dir *.exe
```

磁碟區 C 中的磁碟沒有標籤。

磁碟區序號： 70AE-6E8A

C:\ch01 的目錄

2010/03/12	上午 09:01	16,019	sum.exe
	1 個檔案	16,019	位元組
	0 個目錄	11,429,384,192	位元組可用

```
C:\ch01>sum
```

```
sum=55
```

```
請按任意鍵繼續 . . .
```

使用 gcc 產生組合語言

- 使用 **-S** 參數可要求 gcc 產生組合語言

```
gcc -S sum.c -o sum.s  
gcc -S main.c -o main.s
```


將 gcc 當成組譯器使用

圖 1.6 將 gcc 當成組譯器使用

```
C:\ch01>gcc -S sum.c -o sum.s

C:\ch01>gcc -S main.c -o main.s

C:\ch01>gcc main.s sum.s -o sum2

C:\ch01>sum2
sum= 55
請按任意鍵繼續 . . .
```

同時組譯並連結

圖 1.7 利用 gcc 編譯 C 語言 main.c 同時組譯組合語言 sum.s

```
C:\ch01>gcc main.c sum.s -o sum3
```

```
C:\ch01>sum3
```

```
sum=55
```

```
請按任意鍵繼續 . . .
```

習題

1. 請說明何謂系統軟體？
2. 請列出你所知道的系統軟體。
3. 請說明系統軟體與系統程式兩者有何區別。
4. 請說明組合語言在系統軟體學習上的角色。
5. 請說明 C 語言在系統程式上的用途。
6. 請列出您所經常使用的程式語言，並說明其相關的系統軟體之用法。
7. 請從網路下載 Dev C++ 軟體，並參照附錄 D 的說明，安裝並使用 Dev C++ 撰寫 C 語言程式，並學習該軟體的用法。
8. 請找出 Dev C++ 當中的 GNU 工具，並在設定好 PATH 環境變數後，試用 gcc 指令編譯任意一個 C 語言程式 (設定方法請參考本書附錄 D)。