

認識C語言

C語言概述

1

- C語言由Dennis Ritchie博士，於1972年在貝爾實驗室(Bell laboratory) 開發
- 由B語言而來，用來開發 DEC PDP-II電腦 的系統程式，與Unix作業系統有密不可分的關係
- 許多作業系統及驅動程式由C語言撰寫
- 許多高階語言的編譯器或直譯器亦是C的傑作
- 發展
 - Unix作業系統
 - C++與Java為C根基，再加上物件導向功能

C語言的特色

■ C特性

- 高效率的編譯式語言：程式 → 機器碼(machine code) → 執行
 - 直譯器(Interpreter)
 - 編譯器(Compiler)
- 介於高階與低階之間的語言
- 靈活的程式控制流程
- 可攜性佳
- 為程式設計師所設計的語言
 - 直接依記憶體的字址來存取變數，執行效率佳
- C語法嚴謹簡潔，尤其是指標pointer的應用

直譯器 VS 編譯器

➤ 直譯器

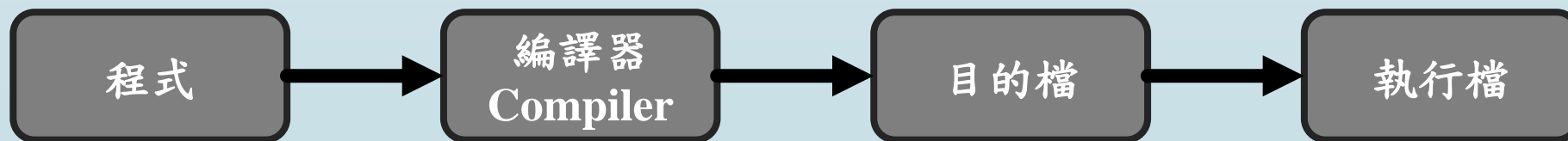
- 逐行檢查程式語法
- 如果沒有錯誤則執行該行程式
- 遇到錯誤則會立即中斷
- 程式執行完畢
- 佔用記憶體較少，僅需要存取原始程式
- 編譯時間會延遲執行時間，執行速度較慢，效率低
- Javascript，PHP等



直譯器 VS 編譯器

➤ 編譯器

- 檢查所有程式是否語法錯誤
- 編譯成可執行檔，每次修改程式都必須重新編譯
- 執行時不需要再編譯，執行效率高
- C、Cobol、Pascal、Java等

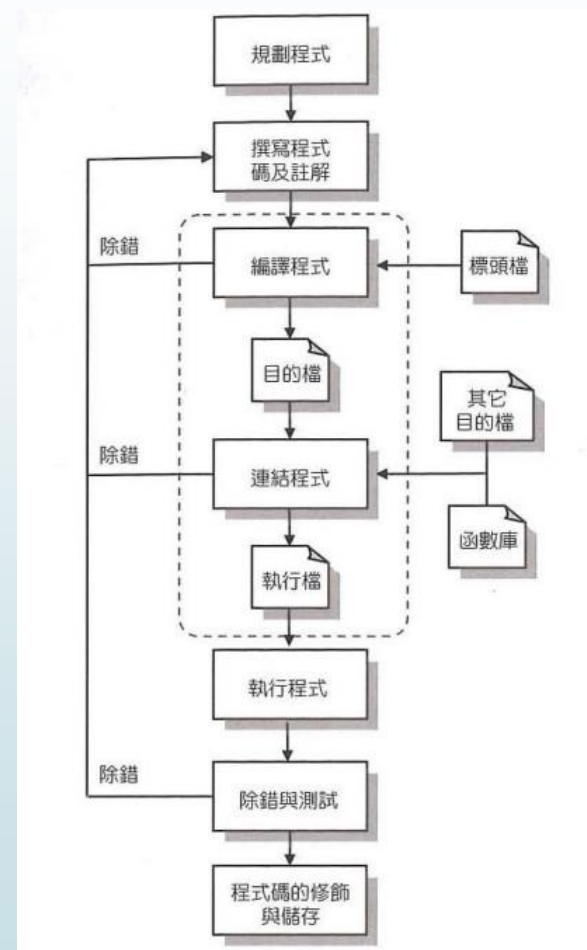


介於高階與低階之間的語言

- 程式語言概分兩大類：「低階語言」與「高階語言」
- 低階語言
 - 組合語言
 - 電腦中執行效率相當高，且對於硬體控制的程度高
 - 但對人類而言生澀難懂、編寫及閱讀與維護困難
- 高階語言
 - 敘述性語言，與人類慣用的語法較相近，較易閱讀、除錯
 - 對硬體控制能力較差，執行效率較差
- C語言
 - 結合低階語言(硬體控制高)和高階語言(易於閱讀、除錯)，固有人稱「中階語言」

程式的規劃與實作

- 規劃程式
 - 流程圖
- 撰寫程式碼及註解
- 編譯與連結程式
 - 利用編譯器轉換成電腦認識的東西
 - 產生目的檔
 - 透過連結器與其他函數模組連結產生可執行檔
- 執行程式
 - EXE檔案，即可發佈給其他人執行



程式的規劃與實作

- ➡ 除錯與測試
 - ➡ 語意錯誤(semantic error)
 - ➡ 語法錯誤(syntax error)
 - ➡ bug / debug
- ➡ 程式碼的修飾與儲存

bug由來

- 在1947年由第一名女程式員葛麗絲·霍普 (Grace Hopper) 發現電腦上的第一個bug。當她的團隊在Mark II電腦上工作的時候，電腦突然不能正常運作，團隊的電腦程式員都找不到原因。經過大家一番努力下，發現原來是有一隻飛蛾意外飛入了電腦的內部，從而引起故障。團隊在日記中記錄了這件事，於是電腦人開始用「Bug」這個字表達電腦中的錯誤，這本日記的遺稿現時仍收藏在華盛頓的美國國家歷史博物館中。

葛麗絲·穆雷·霍普
(Grace Murray Hopper)



葛麗絲·霍普 (Grace Hopper)



Relay #70 Panel F (moth) in relay

First actual case of bug being found.

撰寫第一個C程式

- 編輯器 (dev C++)
- 編譯與執行的過程
 - 產生目的檔(object file)
 - 將標頭檔(header file)讀進來，根據標頭檔內記載的函數原型(prototype)，檢查程式中所使用到的函數用法是否正確，若無錯誤的話就會產生目的檔(預設dev c++ 刪除)，意指「目的檔」代表一個已經編譯過且沒有錯誤的程式。
 - 連結程式(linker)
 - 將其他的目的檔及函數庫(library)連結在一起後，產生「.exe」可執行檔。
 - 可執行檔
 - 獨立執行的程式(還要看電腦環境是否符合)

- 檔案→開新檔案→原始碼
- 儲存檔案→存檔類型(C source file *.c)
- 執行→編譯並執行

C語言起手式

➡ `#include` <標頭檔.h>

➡ `stdio.h`

➡ 程式進入點函數名稱:`main`

```
int main(void){  
    return 0;  
}
```

➡ 每一行程式以半形「`;`」結尾

➡ 大小寫區分

➡ 重要函數

➡ `printf()`


➡ `\n` (換行)

➡ `system("pause")`

Hello World C

➡ 第一個程式

```
#include <stdio.h>
int main(void){
    printf("Hello World C");
    system("pause");
    return 0;
}
```

 C:\C語言\01-Hello World.exe

Hello World C請按任意鍵繼續 . . . _

習題

13

➡ 請輸出以下結果：

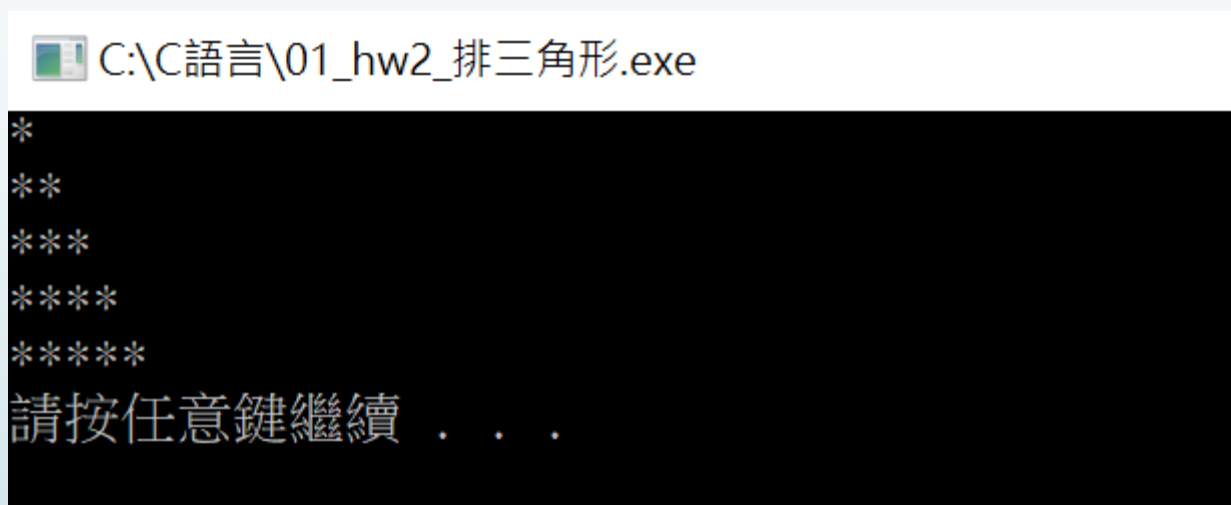


```
C:\C語言\01_hw1_換行.exe  
Hello  
World  
C  
請按任意鍵繼續 . . . _
```

習題

14

➡ 請輸出以下圖形(不用迴圈)

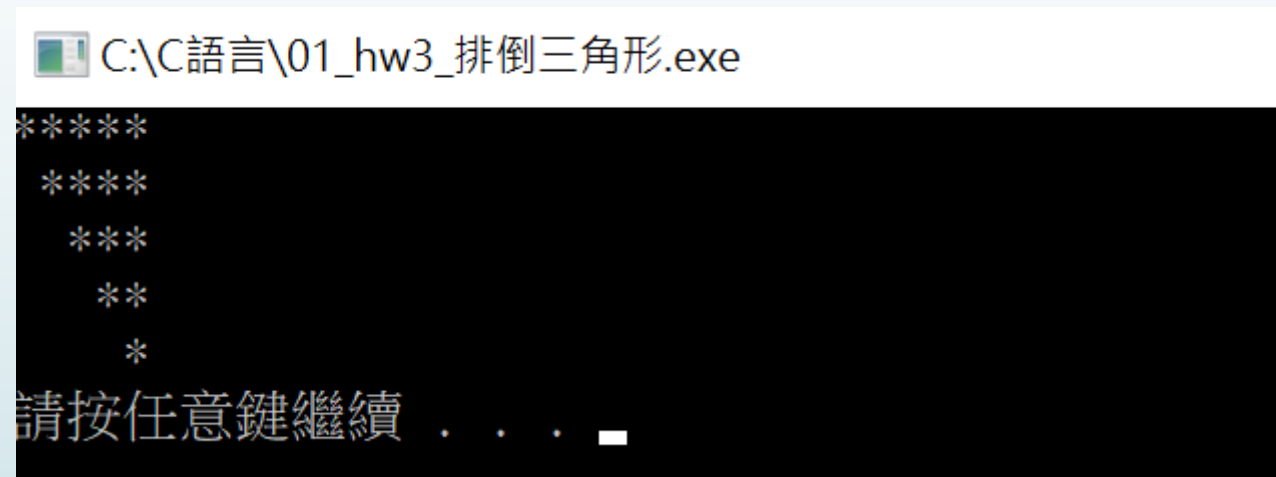


```
C:\C語言\01_hw2_排三角形.exe
*
**
***
****
*****
請按任意鍵繼續 . . .
```

習題

15

➡ 請輸出以下圖形(不用迴圈)



```
C:\C語言\01_hw3_排倒三角形.exe
*****
****
***
**
*
請按任意鍵繼續 . . .
```