

# 導論

## 本章大綱

- ▶ 1-1 程式語言基本概念、功能及應用
- ▶ 1-2 C 語言開發環境介紹
- ▶ 1-3 高中生解題系統
- ▶ 1-4 APCS

## 1-1 程式語言基本概念、功能及應用

### 程式與電腦的關係

人與人之間溝通的工具稱為語言，世界上因有許多民族，因其發源地不同，所以就有許多語言。例如，華語、英語及德語等。其次，人與電腦溝通的工具，則稱為電腦程式語言。那為什麼沒有電視語言、冰箱語言或冷氣語言呢？那是因為這些機器的功能較為簡單，只要幾個按鈕就能發揮其功能，但是電腦的功能就非常多，多到連用整個鍵盤的所有按鍵都無法表現其功能，所以必須使用一些類似單字所組成的片語與敘述來發揮其所有功能，這些單字與片語的集合就稱為電腦程式語言，簡稱程式語言。就如同人類也無法用 26 個字母表達所有感受與思維，必須藉助這些字母的排列組合，先組成單字，再由單字組合成片語與

句字，才能充分表達其思維。目前流行的程式語言有 Java、C、C++、Visual Basic、C#、Python 等。

## 電腦程式設計

串連一些程式語言中的指令與資料，使其完成一件工作，就稱為電腦程式設計，簡稱程式設計。

## 運算思維

既然是電腦程式設計，那就要以電腦的思維方式去寫程式，而不是以人的思考方式寫程式。電腦的運算思維要從資料結構與演算法兩個運算思維開始。那什麼是資料結構與演算法，分別說明如下：

- 資料結構

什麼是資料結構呢？以解一元二次方程式  $ax^2+bx+c=0$  為例。此時  $a, b, c$  要以單一變數或陣列儲存呢？若選單一變數，那要選整數、長整數或實數儲存呢？此即為資料結構。此一例子比較單純，只要依使用者的需求即可決定哪種資料結構。但有時候就要學習電腦的特性了，以時間為例，人類是分別使用年、月、日、時、分、秒代表一個時間點，那就需要 6 個變數儲存，其次，兩個時間點的差額是分鐘不夠時，去借 1 小時來減，日數不夠時，去借 1 個月來減。但電腦可不用這樣想，電腦可是有很強的資料結構，電腦可直接用一個 10 位數的精確長整數『秒』，記憶與代表（又稱儲存）年、月、日、時、分、秒，其次，

		3 7 2
1	法則 3	1 3 8 3 8 4 9
2	$3 \times 20 + 7 = 67$ $67 \times 7 = 469$	3 8 4 4 6 9
3	$37 \times 20 + 2 = 742$ $742 \times 2 = 1484$	1 4 8 4 1 4 8 4
4		0

所有的時間的長度與前後運算也都可用『秒』完成，請看範例 5\_3e。又例如，單複選題答案 ABCDE 要如何儲存較省事呢？要以 A、BCD 或 AE 表示答案，還是直接用一個整數 0 到 31 來表示，以上即為資料的表示與儲存方式，稱為資料結構。電腦最基本的資料結構有單一變數、一維或二維陣列，這都會在本書探討。其它進階資料結構有結構、堆疊、佇列、鏈串列等，這將會在高中進階程式設計討論。

- 演算法

演算法是什麼呢？以人類解開根號為例，是套用國中數學的  $(a+b)^2=a^2+2ab+b^2$ ，如上頁圖，此稱為貪婪演算法，請看範例 7\_5d，因為每一階次都可以得到最佳成果，但電腦就不用這麼麻煩，還可直接用循序猜值法（範例 6\_1b）或二分猜值法（範例 6\_4c）求解。其次，電腦基本演算法還有分而治之演算法，本書將會陸續介紹這些基本演算法。

## 程式設計功能

程式設計可將一連串重複的工作，使用程式語言，寫一段程式、儲存，然後就可無限次數重複使用，這樣可節省很多時間。例如，範例 4\_3c 的解一元二次方程式為例，使用者必須不斷的計算  $d=\sqrt{(b^2-4ac)}$ ， $x1=(-b+d)/2a$ ， $x2=(-b-d)/2a$ ，才能解出方程式，那如果寫成程式，每次要求解時，只要輸入係數 a,b,c 即可求出方程式的解。

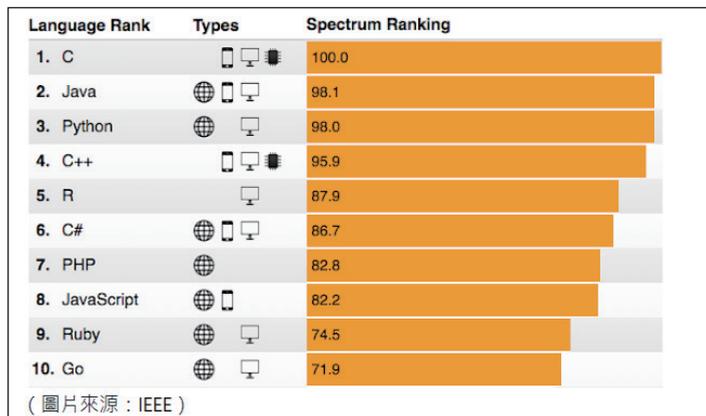
## 應用

程式設計的應用已經與我們生活習習相關，以前只是每個家庭有一部電腦，現在幾乎人手一支手機，手機內即有千變萬化的通訊軟體、遊戲。以前只有科學家、銀行有運用電腦計算，但是

現在的自駕車已經可以上路，鴻海郭台銘董事長也說五年後他的工廠不用開燈、不用人、不用供餐、不用住宿，使用機器人就可 24 小時生產，所以現在的程式設計可說是融入我們的生活了。其次，現在人口越來越少，會凸顯人口老化問題，居家生活照顧就要仰賴機器人了，每個人眼睛一張開，就要學習與電腦機器人一起生活。還有更可怕的事，以前是每個家庭一支電話，現在是每個人一支手機，不久的將來就會身邊的所有東西都有一個通訊裝置，例如，冰箱、電視、冷氣、汽車都有一個通訊裝置，都可以控制與回報其狀態，那就是物聯網的時代，不僅人與人可以聯繫，就連人與物或物與物都可以聯繫，這麼複雜的工作，都是電腦程式的應用。

## 程式語言的種類

一份由電機電子工程學會發行的雜誌 IEEE(2018.10)，透過分析來自 10 種來源的 12 個指標，算出目前熱門程式語言排行榜如下圖，C 語言還是目前第一名。其次，大學端設計的『大學程式設計先修檢測 APCS』學科，也以 C 語言為程式設計語言，中學生學習 C 語言可完全無縫接軌高中與大學程式設計課程，所以本書選用 C 語言為中學生『資訊科技』程式設計的程式語言。



## 1-2 C語言開發環境介紹

目前 C 語言常用熱門的開發環境有 Dev-C++、CodeBlocks，分別說明如下：

### Dev-C++

Dev-C++ 是一套用於開發 C/C++ 的自由整合式開發環境 (IDE)，雖然目前沒有繼續開發新版本，但因為其所佔記憶體很小，不管電腦多麼舊，還是可以順暢執行，所以在此介紹此一軟體。以下是 Dev-C++ 官網首頁 (www.bloodshed.net)。



### 下載與安裝

於上圖點選左邊 Download，依照指示即可下載與安裝 Dev-C++。

### 調整中文介面

若您的功能表選項不是中文，可點選功能表的“Tools/Environment Options”，再於 Language 中點選“Chinese(TW)”，如下圖。



## 調整字型大小

點選功能表的『工具』/『編輯器選項』/『字型』，如下圖，即可調整字型大小。



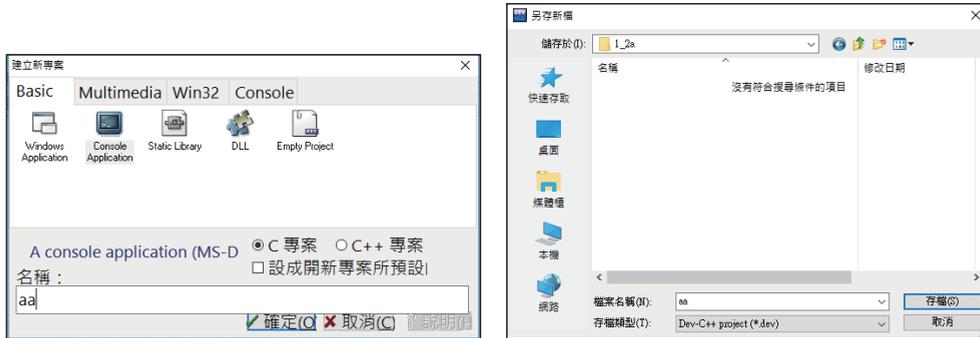
### 範例1-2a

請寫一程式，可以於螢幕輸出『Hello』。本例所需程式僅為『`printf("Hello");`』。

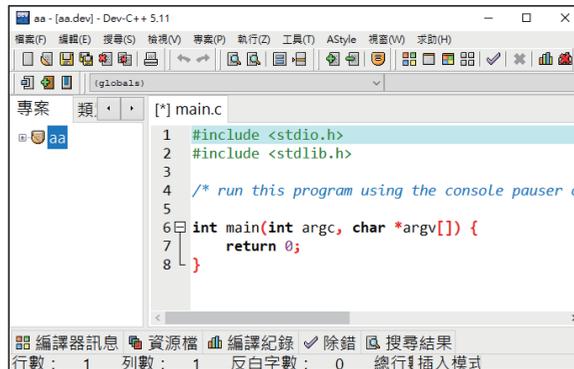
### 操作步驟

1. 開新專案。點選功能表的『檔案』/『開新檔案』/『專案』，畫面出現要求點選專案類型、存放專案的資料夾與專案名稱，如下圖左，本例專案類型點選『Console Application』與『C

專案』，點選資料夾為『1\_2a』(請先新增資料夾『1\_2a』，讓一個專案使用一個資料夾)，專案名稱輸入『aa』，如下圖左，畫面會出現另存新檔畫面，如下圖右。



完成以上操作，畫面已新增程式樣版如下圖：



2. 輸入本例所需程式。新增專案時，已經新增樣版 main() 函式如下：

```
#include <stdio.h>
#include <stdlib.h>
/* run this program using the console pauser or add your own
getch, system("pause") or input loop */
int main(int argc, char *argv[]) {
    return 0;
}
```

3. main() 函式的參數列有點複雜，但初學者先不要理會，只要將本例所需程式

```
printf("Hello");
```

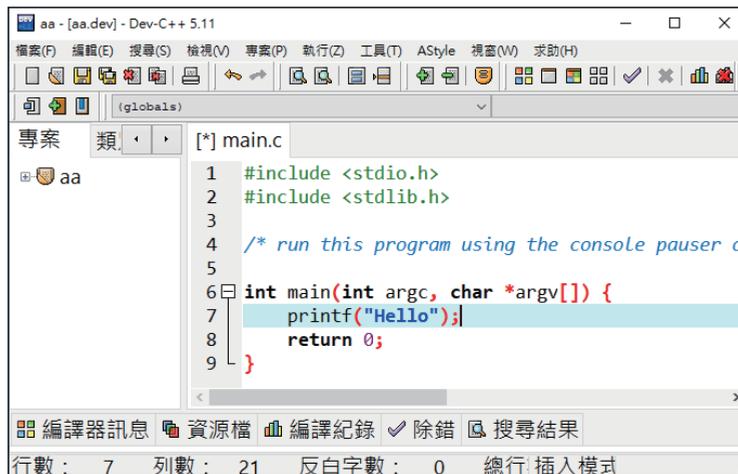
填入 main() 內即可，如以下程式：

```
int main(int argc, char *argv[]) {  
    printf("Hello");  
    return 0;  
}
```

其次，

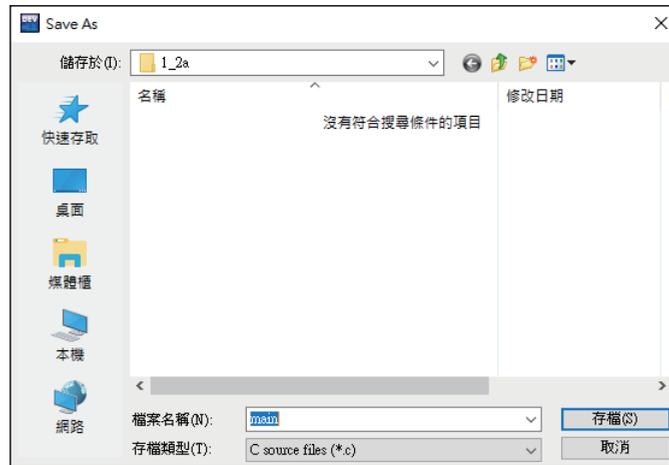
```
#include <stdio.h>  
#include <stdlib.h>
```

也先不要理會，它們是一些常用的函式標頭檔，讓您可以使用一些常用函式，關於函式的使用，請看第八章。還有，『/\*...\*/』是註解，請看 4-2 節，這裡也先不要理會。完成以上操作，畫面如下：



```
aa - [aa.dev] - Dev-C++ 5.11  
檔案(F) 編輯(E) 搜尋(S) 檢視(V) 專案(P) 執行(Z) 工具(T) AStyle 視窗(W) 求助(H)  
(globals)  
專案 類  
aa  
[*] main.c  
1 #include <stdio.h>  
2 #include <stdlib.h>  
3  
4 /* run this program using the console pauser o  
5  
6 int main(int argc, char *argv[]) {  
7     printf("Hello");  
8     return 0;  
9 }  
編譯器訊息 資源檔 編譯紀錄 除錯 搜尋結果  
行數： 7 列數： 21 反白字數： 0 總行： 插入模式
```

4. 執行程式。點選功能表的『執行』/『編譯並執行』，畫面出現如下，要求輸入主程式的檔名，系統預設『main』，本書均使用此預設值。



5. 執行結果。

```
D:\cb\1_2a\aa.exe
Hello
-----
Process exited after 0.08279 seconds with return value 0
請按任意鍵繼續 . . .
```

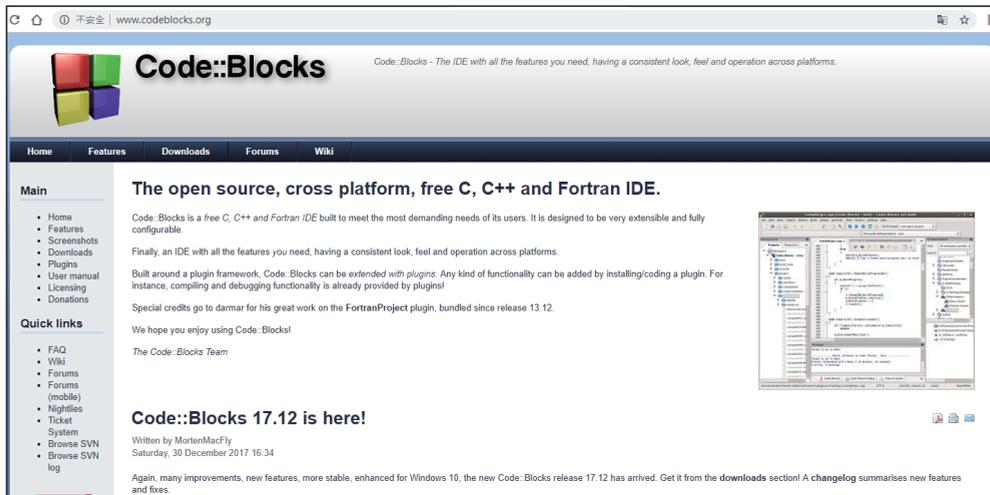
6. 開啓檔案總管，全部檔案如下：這也不要理會，反正一個專案就是要用一個資料夾，因為一個專案包含專案檔、應用程式、C 的原始檔、檔案或圖片等，這些檔案都要放在同一個資料夾，那當您要複製專案時，才不會遺漏這些檔案。



名稱	修改日期	類型	大小
aa	2018/11/23 上午 ...	Dev-C++ Project...	1 KB
aa	2018/11/23 上午 ...	應用程式	128 KB
main	2018/11/23 上午 ...	C Source File	1 KB
main.o	2018/11/23 上午 ...	O 檔案	1 KB
Makefile.win	2018/11/23 上午 ...	WIN 檔案	2 KB

## CodeBlocks

CodeBlocks 是目前最熱門的 C/C++ 自由整合式開發環境 (IDE)，APCS 考試也提供此 IDE，官網如下圖：



點選『DownLoad』，畫面如下圖：



點選『Download the binary release』，畫面如下圖：



File	Date	Download from
codeblocks-17.12-setup.exe	30 Dec 2017	Sourceforge.net
codeblocks-17.12-setup-nonadmin.exe	30 Dec 2017	Sourceforge.net
codeblocks-17.12-nosetup.zip	30 Dec 2017	Sourceforge.net
codeblocks-17.12mingw-setup.exe	30 Dec 2017	Sourceforge.net
codeblocks-17.12mingw-nosetup.zip	30 Dec 2017	Sourceforge.net
codeblocks-17.12mingw_fortran-setup.exe	30 Dec 2017	Sourceforge.net

NOTE: The codeblocks-17.12-setup.exe file includes Code::Blocks with all plugins. The codeblocks-17.12-setup-nonadmin.exe file is provided for convenience to users that do not have their machine(s).

NOTE: The codeblocks-17.12mingw-setup.exe file includes *additionally* the GCC/G++ compiler and GDB debugger from TDM-GCC (version 5.1.0, 32 bit, SJLJ). The codeblocks-17.12mingw-nosetup.zip file includes *additionally* to that the GFortran compiler (TDM-GCC).

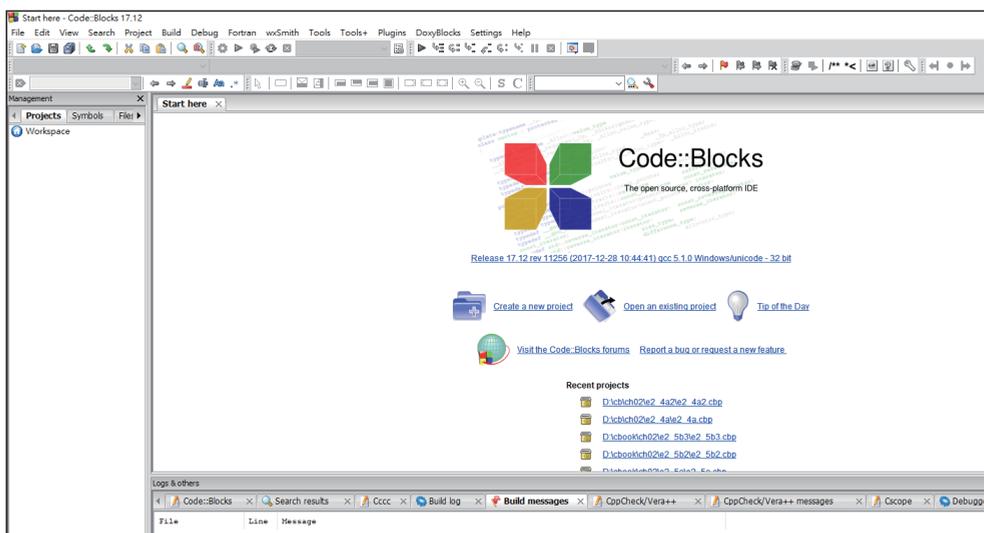
NOTE: The codeblocks-17.12(mingw)-nosetup.zip files are provided for convenience to users that are allergic against installers. However, it will not allow to select plugins / features to install (everything) and not create any menu shortcuts. For the "Installation" you are on your own.

*If unsure, please use codeblocks-17.12mingw-setup.exe!*

初學者請直接點選『codeblocks-17.12mingw-setup.exe』，因為它才有內含『C/C++』編譯軟體，不然要自備編譯軟體。

## 安裝軟體

下載的檔案就是『.exe』執行檔，請在『下載區』按兩下程式，就可自動執行。開啓『CodeBlocks』，畫面如下圖：

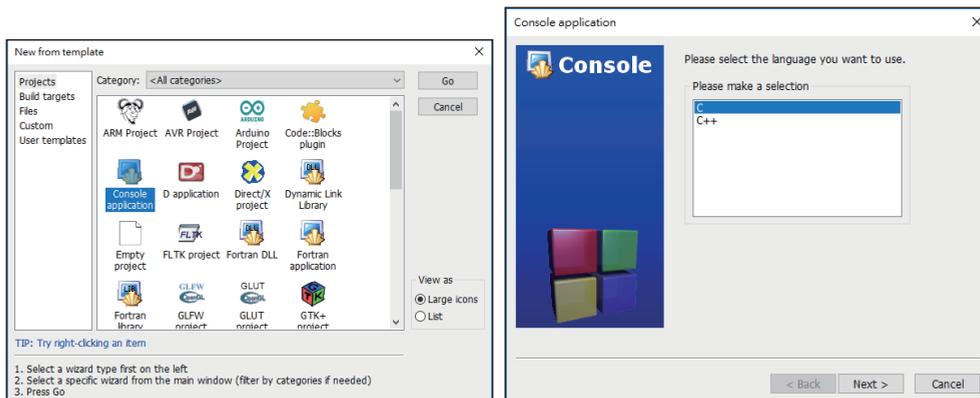


**範例1-2a**

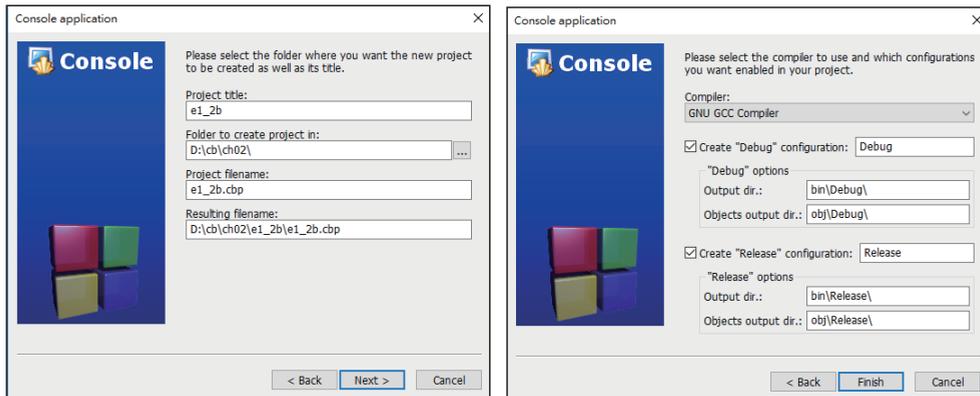
請寫一程式，可以於螢幕輸出『Hello』。本例所需程式僅為『`printf("Hello");`』。

**操作步驟**

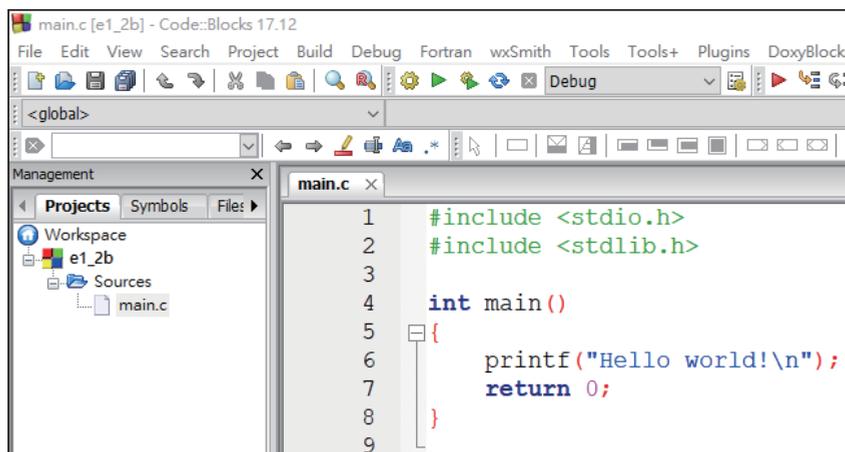
1. 開新專案。點選功能表的『File』/『New』/『Project』，畫面出現要求點選專案類型，如下圖左，本例專案類型點選『Console Application』，再點選『Go』，畫面如下圖右，可點選專案類型，本例點選『C』。



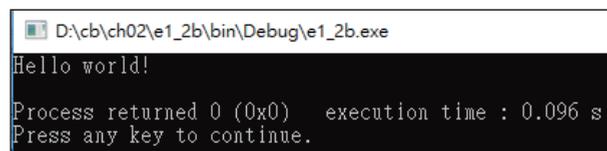
2. 點選資料夾與輸入專案名稱。  
本例點選資料夾為『d:\cb\ch02』，專案名稱輸入『e1\_2b』，如下圖左，電腦將會自動建立資料夾『e1\_2b』。



3. 完成以上操作，畫面如下圖：（請點開『Source』前面『+』，再按二下『main.c』）



4. 執行程式。  
點選功能表的『Build』 / 『Build and Run』，即可執行程式，畫面如下圖：

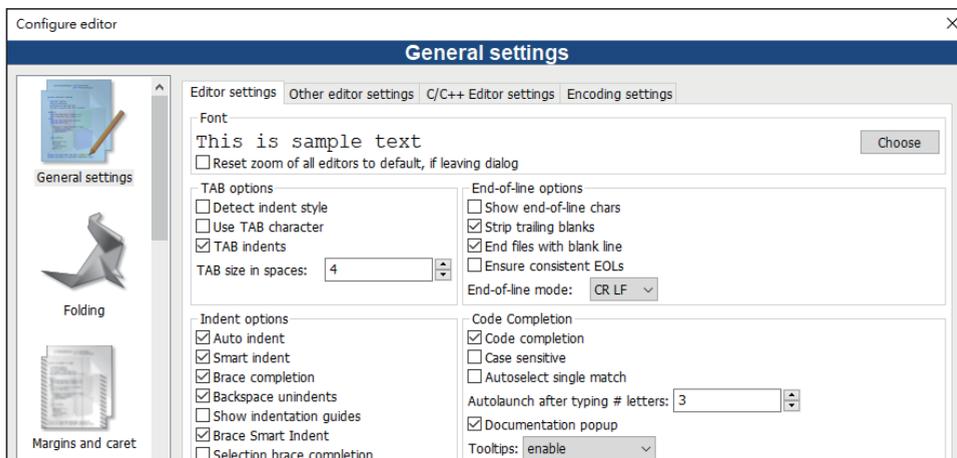


5. 以上 `printf()`，就是本例所執行的程式，往後您要寫任何的程式，只要將此敘述刪除，並在此位置寫下您的程式就可以。
6. 完成以上操作，資料夾『e1\_2b』，如下圖。已經有專案檔與 `main.c` 原始程式檔（APCS 考試請點選下圖的 `main.c` 再上傳）



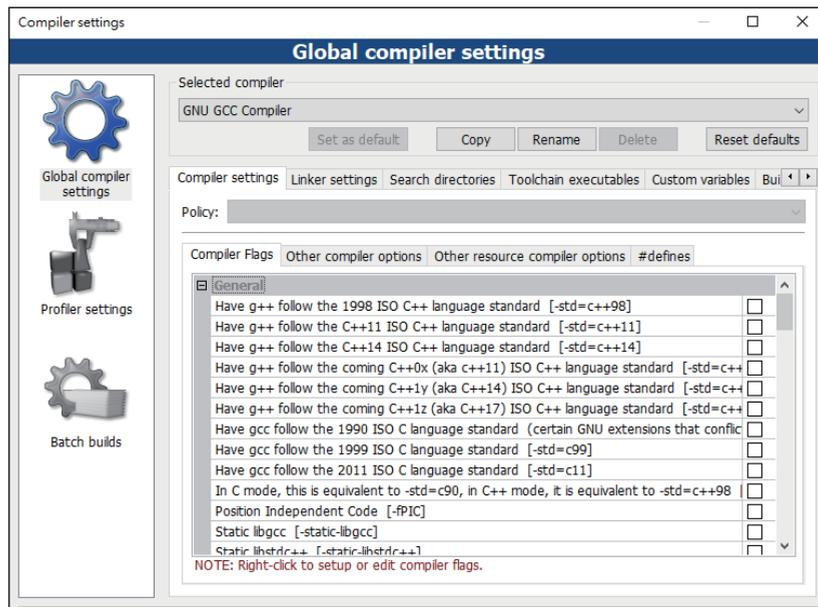
7. 如何調整字型大小。

請點選功能表『Settings』/『Editor』，畫面如下圖：

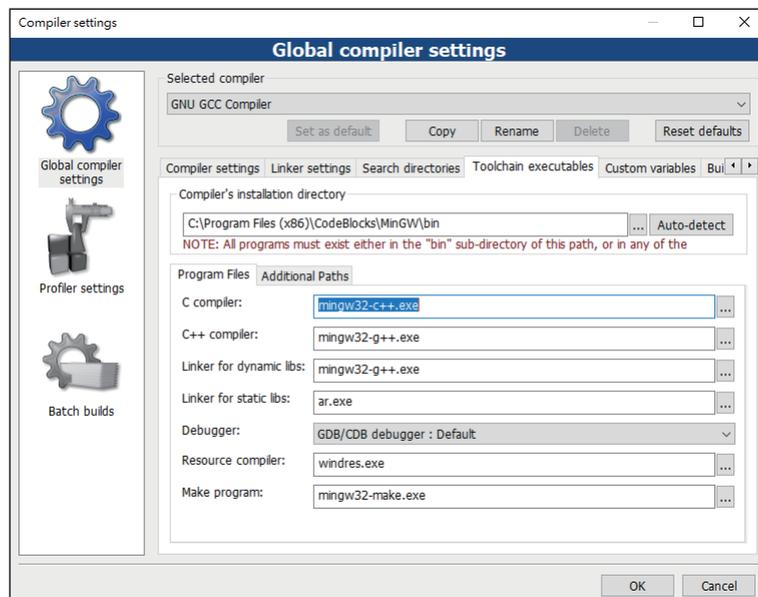


8. 若出現找不到編譯器？（若之前您有安裝其他 C 語言，有可能系統誤判而找不到編譯器）

(1) 請點選功能表『Setting』/『Compiler』，畫面如下，在這一畫面，您可以指定要使用哪一版本編譯器。（C 語言編譯器有三個版本，分別是 1990、1999 與 2011，往後線上查詢時就會明瞭）



(2) 繼續點選『toolchain executables』，畫面如下圖，請留意您的編譯器路徑是否正確，若不正確請在此修改。



9. 若左邊的『Management』視窗不見。請點選『View/Manager』。

## ■ 補充說明

1. 鍵入程式應該要有層次感。例如，函式實作應該要使用 `tab` 鍵內縮（初學者常用空白鍵內縮，那會很吃力，且無法完全整齊）。例如，以下的 `printf("Hello");`，請先鍵入 `tab` 鍵，再鍵入程式。

```
int main(int argc, char *argv[]) {  
    printf("Hello");  
    return 0;  
}
```

還有遇到 `for`、`do`、`while` 迴圈或 `if` 等，也都要先用 `tab` 鍵內縮，請看以下各程式。

## 常犯的錯誤

以下是一些鍵入程式時常犯的錯誤，整理如下，方便初學者除錯。

1. 忘記在敘述行尾端打上分號（；），例如：

```
printf("Hello")
```

或是不應該出現分號的地方，多打了分號。例如：

```
#include <stdio.h>;  
if (a>b);
```

且分號『;』是英數下的半形，不是中文狀態下的全形。

2. 大小寫未區分。C 語言規定大小寫的字母是不同的字元，應注意字母的輸入。例如，

```
printf("Hello");  
Printf("Hello"); //請留意P的大小寫
```

就不一樣了。

3. 拼錯字。例如將

```
printf("Hello");
```

打成

```
printf("Hello"); //請留意m與n或l與!
```

那指令就錯了，將會出現錯誤信息。

4. 括弧未配成對。有一個左大括弧『{』，務必要配一個右大括弧『}』；同樣地，左小括弧『(』務必要配一個右小括弧『)』；左中括弧『[』務必要配右中括弧『]』。初學者經常會忘記打上右括弧，導致語法錯誤。
5. 引號未成對。不論雙引號『"』(用於字串)，或是單引號『'』(用於字元)的使用，都應該成雙成對出現。
6. 不該出現的空白符號。例如，把

```
printf("Hello");
```

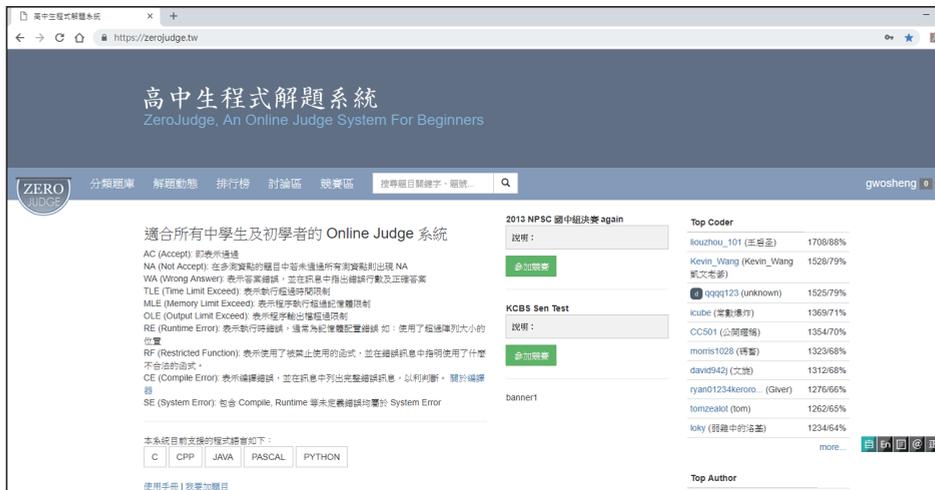
打成

```
print f("Hello");
```

也不行。

## 1-3 高中生程式解題系統

高中生解題系統（<https://zerojudge.tw/>）畫面如下圖：



高中生程式解題系統 (Online Judge)，它是一個適合所有中學生及初學者的程式設計題庫系統，內有基礎題庫與競賽題庫，每一個題目都有詳盡說明與測試資料，學生還可送出程式，由系統協助檢查是否正確，其次，這個系統不僅能評判是否正確，若有錯誤，還能提示錯誤的地方，真是一個很好的學習資源。下圖是基礎題庫：

基礎題庫	競賽題庫	IOI/APCS	Uva 題庫	原創/不分類題庫
第 1 2 3 4 5 6 7 8 9 10 11 12 頁				
序號	題目			標籤
1	a001. 哈囉 – Brian Kernighan	91% / 40805 人		基本輸出輸入
2	a002. 簡易加法	91% / 37590 人		基本輸出輸入
3	a003. 兩光法師占卜術	96% / 30603 人		曆法
4	a004. 文文的求婚	94% / 27915 人		曆法
5	a005. Eva 的回家作業 – POJ	94% / 23082 人		圖論
6	a006. 一元二次方程式	94% / 18751 人		數學
7	a009. 解碼器 – ACM 458	96% / 13609 人		字元處理
8	a010. 因數分解	90% / 9540 人		數論
9	a013. 羅馬數字 – NPSC 模擬試題	83% / 3358 人		進位制
10	a015. 矩陣的翻轉	96% / 11441 人		陣列

只要學完本書第二章，就可開始從基礎解題。其中第一題還有參考解答，您可以複製此程式，練習此系統的使用。下圖是目前各校的解題數排行榜。

排名	學校	統計
1	新北市立板橋高級中學 (AVG: 52.4)	AC: 57325次 / 1094人
2	臺北市立中山女子高級中學 (AVG: 21.2)	AC: 47374次 / 2238人
3	臺北市私立延平高級中學 (AVG: 36.4)	AC: 46722次 / 1282人
4	臺北市立第一女子高級中學 (AVG: 26.5)	AC: 43081次 / 1627人
5	臺北市立成功高級中學 (AVG: 24)	AC: 39033次 / 1623人
6	國立嘉義高級中學 (AVG: 58.4)	AC: 30425次 / 521人
7	高雄市立高雄高級中學 (AVG: 15.9)	AC: 28917次 / 1819人
8	國立臺中高級工業職業學校 (AVG: 27.3)	AC: 27218次 / 996人
9	國立臺灣師範大學附屬高級中學 (AVG: 15.5)	AC: 23732次 / 1532人
10	國立彰化高級中學 (AVG: 15.4)	AC: 22200次 / 1439人

本人覺得裡面很多題目，非常適合中學生動腦實作，所以在此大力推薦，希望能勾起大家的興趣，從中好好大顯身手，那程式設計功力一定可以大大提升。請自己註冊後，就可以開始。本書 2.4 節還會繼續介紹。

## 1-4 APCS

APCS 為 Advanced Placement Computer Science 的英文縮寫，是指「大學程式設計先修檢測」，官網是（<https://apcs.csie.ntnu.edu.tw>），開啓後畫面如下圖：



主辦單位期望舉辦具公信力之「程式設計檢測」，檢驗具備程式設計能力之高中職學生的學習成果，提供大學作為選才的參考依據。並藉由本檢測之推動，除了讓高中職重視資訊科學課程的學習外，亦讓大學酌訂抵免程式設計學分的相關措施。108 學年度已經有 31 院校提供專屬有 APCS 成績的推甄名額，請自行開啓官網瀏覽。其次，APCS 每年提供三次考試機會，每次皆有選擇觀念題與實作題，由所提供的考古題與筆者親自報名應試結果，只要多考幾次，不僅可以知道哪裡不足，有些題目也有類似的發展，所以成績一定可以提升，而且報考沒有年齡限制，也不用報名費，成績又可保留，所以建議大家，若對資工、資管相關科系有興趣的同學，國中就可開始報考，多考幾次成績當然會進步。目前官網提供的歷屆考古試題如下，本書將於各章節介紹其相關試題。



The screenshot shows a web interface with a navigation bar at the top containing links for HOME, 關於APCS, 檢測內容, 試題資訊, and 應測生專區. Below the navigation bar is a breadcrumb trail: 你目前位置: 首頁 > 試題資訊 > 歷次試題. A red notice states: 除以下已公開資訊供題型範例參考, 本檢測不公告試題、評分方式或解答。 Below this is a table listing seven exam questions with their dates, types, and file sizes.

日期	試題類型	檔案大小
2017-10-28_實作題_試題下載	檔案類型: pdf	檔案大小: 162KB
2017-03-04_實作題_試題下載	檔案類型: pdf	檔案大小: 467KB
2016-10-29_實作題_試題下載	檔案類型: pdf	檔案大小: 510KB
2016-03-05_實作題_試題下載	檔案類型: pdf	檔案大小: 638KB
2017-03-04_觀念題_試題下載	檔案類型: pdf	檔案大小: 773KB
2016-10-29_觀念題_試題下載	檔案類型: pdf	檔案大小: 695KB
2016-03-05_觀念題_試題下載	檔案類型: pdf	檔案大小: 599KB

以下是 APCS 考試範圍：(108 年 2 月修訂)

- 程式設計基本觀念 (Basic programming concepts)
- 資料型態 (Data types)、常數 (Constants)、變數 (Variables)、視域 (Scope)；全域 (Global) / 區域 (Local)
- 控制結構 (Control structures)
- 迴路結構 (Loop structures)
- 函式 (Functions)
- 遞迴 (Recursion)
- 陣列與結構 (Arrays and structures)
- 基礎資料結構 (Basic data structures)，包括：佇列 (Queues) 和堆疊 (Stacks)
- 基礎演算法 (Basic algorithms)，包括：排序 (Sorting) 和搜尋 (Searching)

**MEMO**