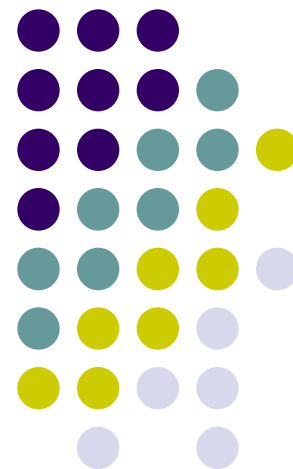
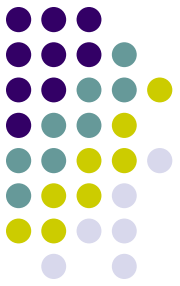


第一章

認識C++

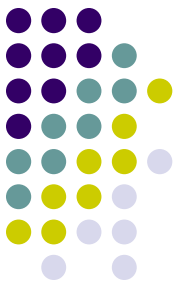
認識C++的歷史
瞭解程式的規劃與實作
撰寫第一個C++程式
學習程式碼的編譯與執行





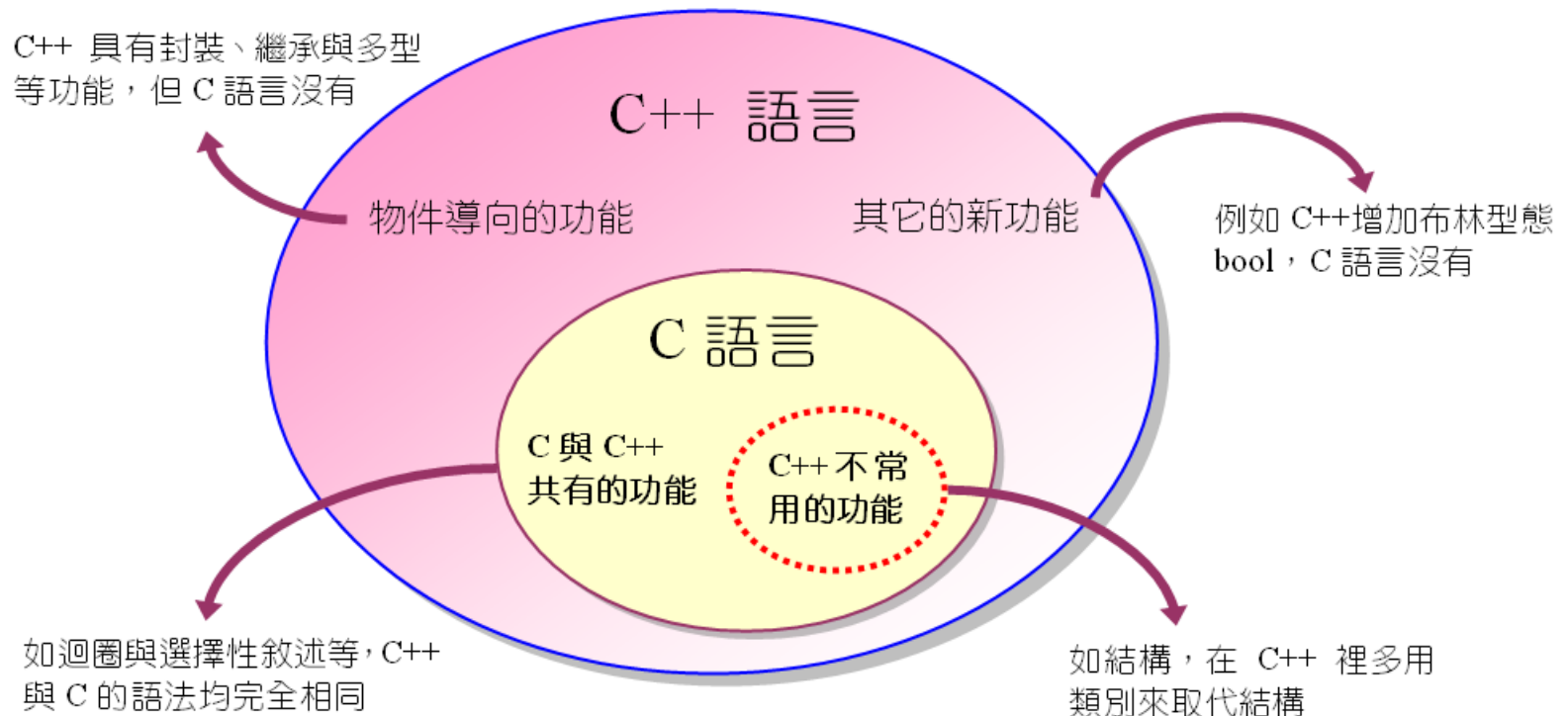
緣起 C/C++

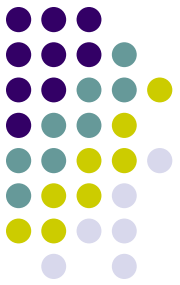
- C語言是於1972年所發展出來的語言
- C++ 於1980年開始發展
 - 將物件導向 (object oriented) 加入C語言中
 - 稱為「C with classes」程式語言



C++ 的特色 (1/3)

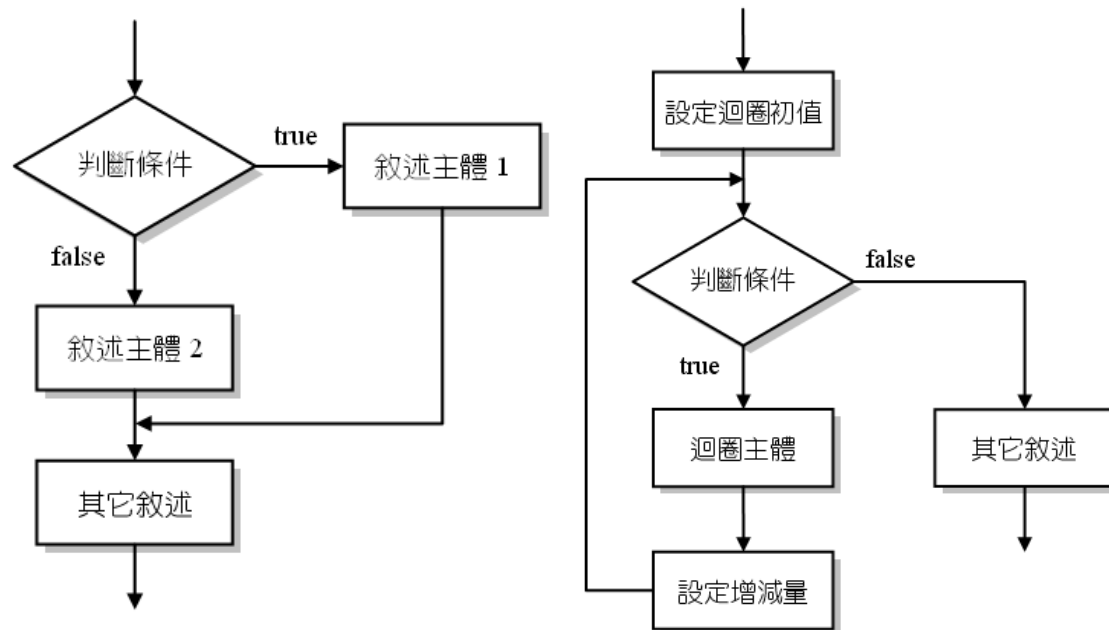
- C++具有下列的幾項特色：
 - 向下相容C語言（C++是由C語言延伸而來，因此包含C的功能）

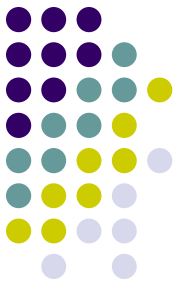




C++ 的特色 (2/3)

- 具物件導向技術的語言
 - 物件導向程式設計 (OOP) 具有的相關技術
 - 「封裝」 (十二章)
 - 「繼承」 (十二章)
 - 「多型」 (十二章)
- 靈活的程式流程控制





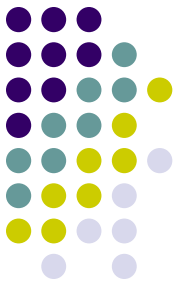
C++ 的特色 (3/3)

- **高效率且可攜性佳的語言**

- C++在執行前必須先經過編譯(Compile)，執行效率遠高於直譯式語言（ Basic 是透過直譯器 interpreter ）
- 可在修改少許程式碼的情況下，便能在不同的作業系統執行（可攜性佳的語言）

- **程式碼的再利用**

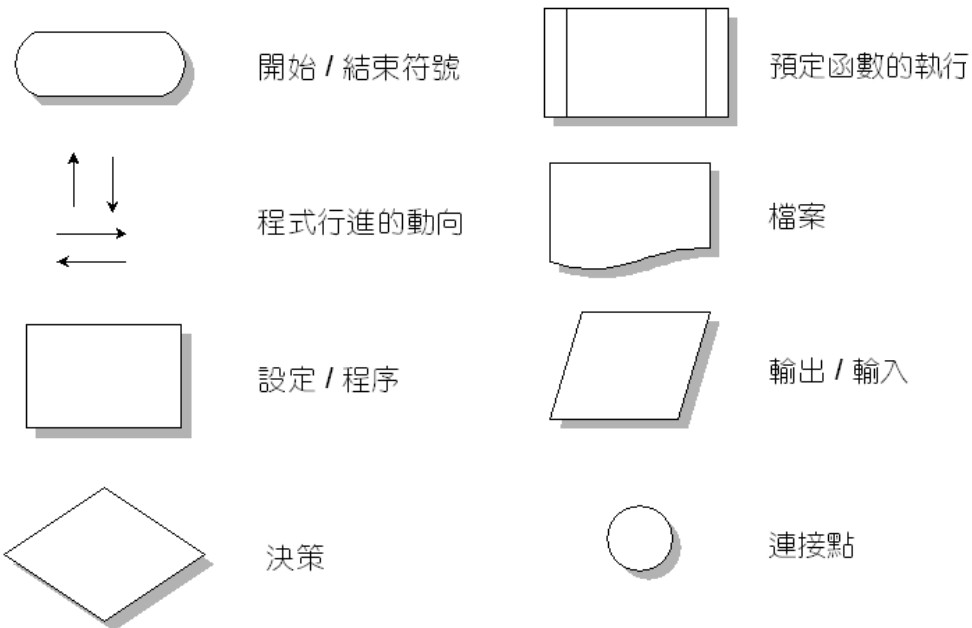
- 可依據原有類別所提供的功能，再加上新的需求，進而設計出新的類別，便是程式碼的「再利用」
- 程式碼的再利用可減少程式碼的開發時間與費用



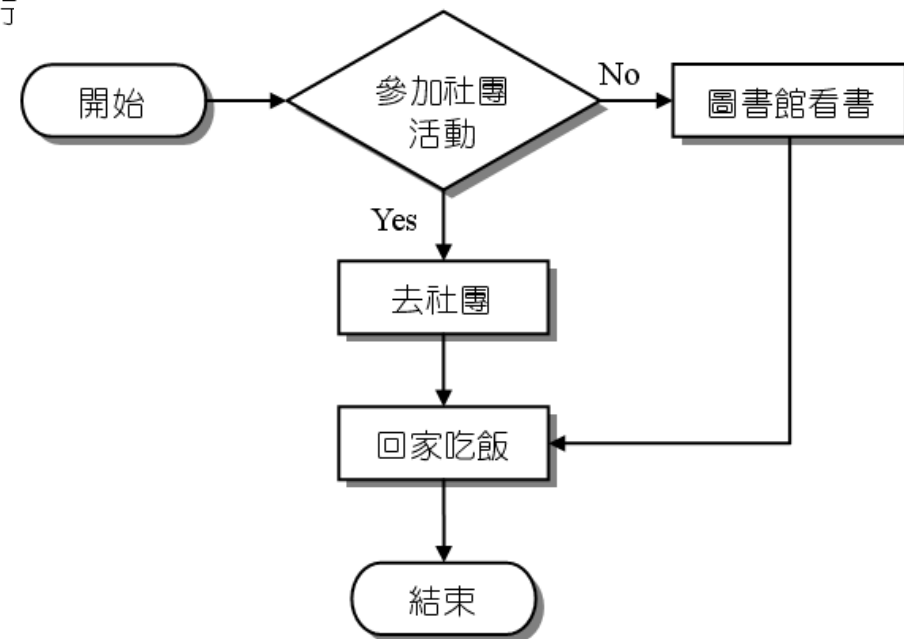
規劃與實作 (1/2)

1. 規劃程式

常用的流程圖符號



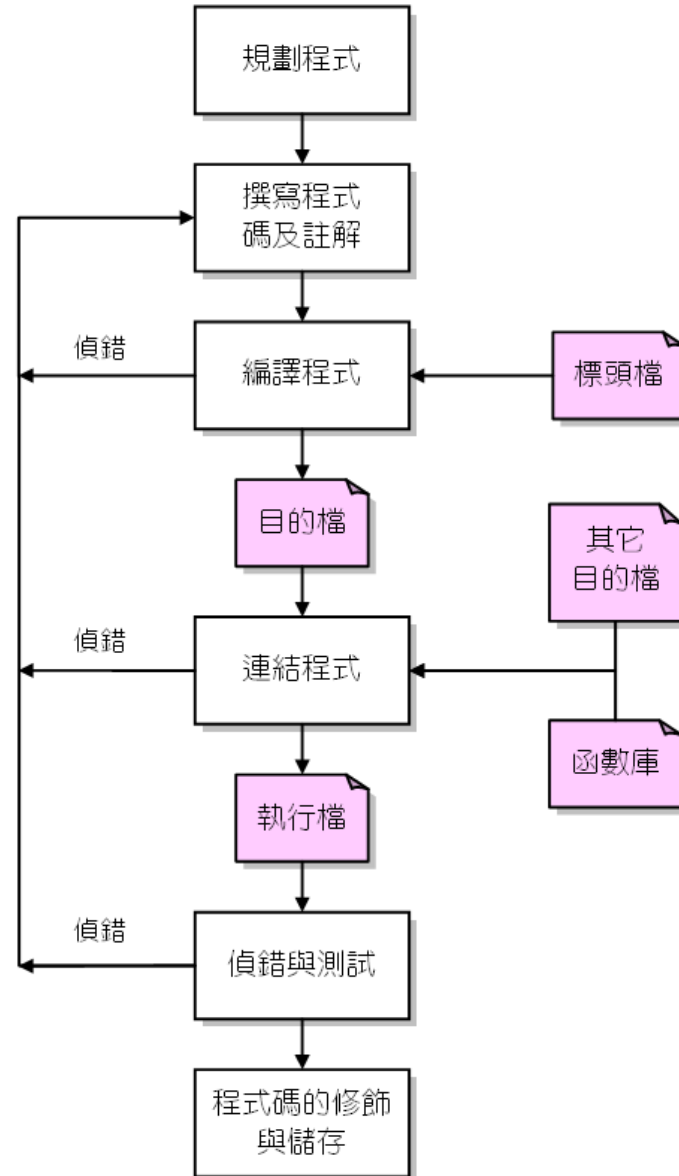
繪製流程圖的範例





規劃與實作 (2/2)

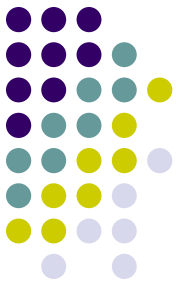
- 2. 撰寫程式碼及註解
- 3. 編譯程式
- 4. 連結程式
- 5. 偵錯與測試
 - 語意錯誤 (semantic error)
 - 語法錯誤 (syntax error)
- 6. 程式碼的修飾與儲存





編譯軟體

- 目前使用者最多的 C++ 整合開發環境軟體為 Dev-C++ 及 Visual C++ Express，不但功能強大，而且這兩套軟體都是免費的。
- Dev-C++ 的功能雖然比不上 Visual C++ 強大，但其系統檔案非常小，只有9M，佔用的資源很少，並且支援中文，操作方式簡易，各種整合功能一應俱全，很適合初學 C++ 語言者使用。
- 微軟公司提供各種語言的免費 Express 版本，其中包含 Visual C++ 2010 Express，使用者可在其網站上自由下載。



編譯軟體

- Dev-C++ 的下載網頁為：
<http://www.bloodshed.net/dev/devcpp.html>，在
下載的 <devcpp-4.9.9.2_setup.exe> 檔案（讀者
下載時版本可能不同）上按滑鼠左鍵兩下，即
可進行安裝。



撰寫 C++ 程式準備工作

- C++ 程式中有些部分是在每個程式都會使用到，可將其視為 C++ 程式的基本架構，包括：
 - C++ 程式執行是由 `main()` 函式開始，所以 C++ 程式一定要有 `main()` 函式。
 - 每個程式必定都有輸出或輸入的部分，要使用輸出、輸入指令必須引入部分標頭檔。
 - 顯示 C++ 程式的結果時，若不加上暫停的指令，將立刻關閉顯示結果的視窗而無法觀察到執行結果。
 - 設計者可先建立好這些程式碼，再複製到新程式檔內，而不必每次都重複輸入。



撰寫 C++ 程式準備工作

- Dev-C++ 工作環境

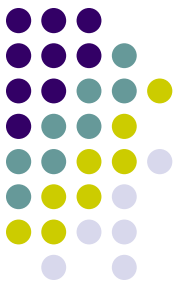




程式碼的編輯到執行 (1/2)

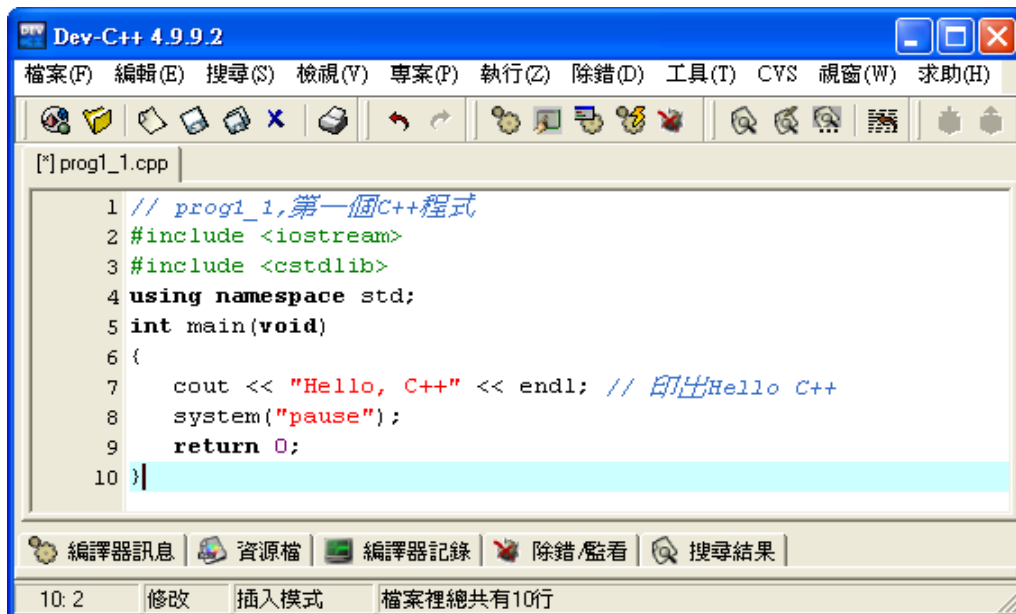
- 第一個C++程式

```
01 // prog1_1, 第一個 C++ 程式
02 #include <iostream>
03 #include <cstdlib>
04 using namespace std;
05 int main(void)
06 {
07     cout << "Hello, C++" << endl;    // 印出 Hello, C++
08     system("pause");
09     return 0;
10 }
```



程式碼的編輯到執行 (2/2)

- 下面的視窗為鍵入程式碼之後的情形：



```
1 // prog1_1, 第一個C++程式
2 #include <iostream>
3 #include <cstdlib>
4 using namespace std;
5 int main(void)
6 {
7     cout << "Hello, C++" << endl; // 印出Hello C++
8     system("pause");
9     return 0;
10 }
```

- 程式碼經過編譯與執行後的執行結果

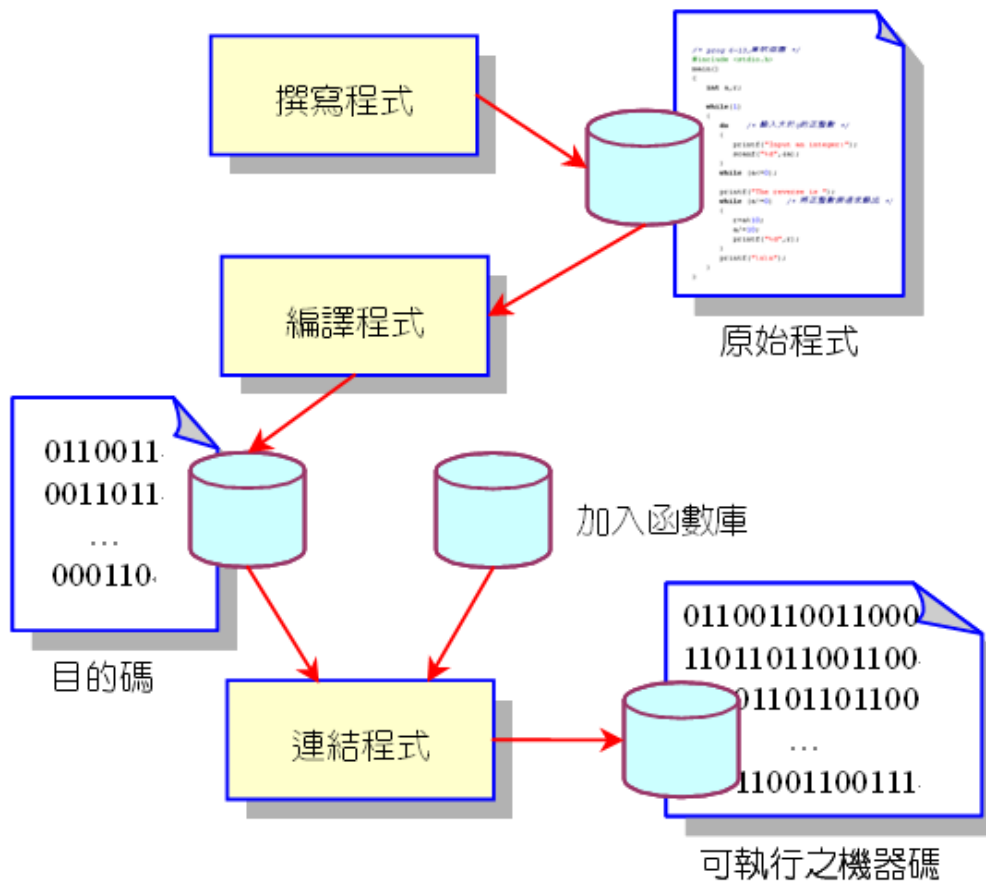


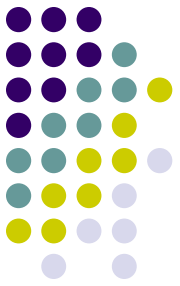
```
C:\Documents and Settings\Administrator\My Documents>
Hello, C++
請按任意鍵繼續 . . .
```



編譯與執行過程的解說

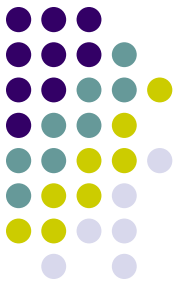
- 原始程式編譯及連結的過程





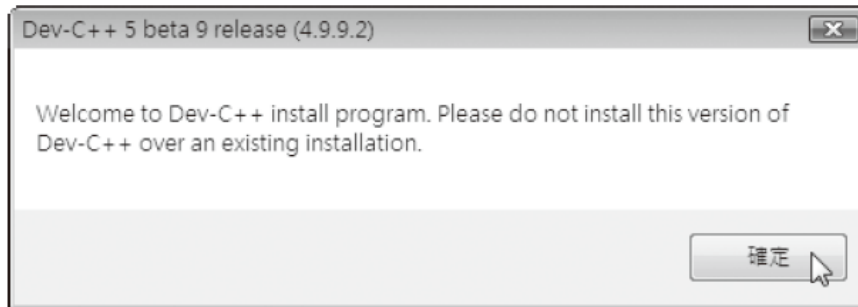
The End-

補充：C++ 的開發環境

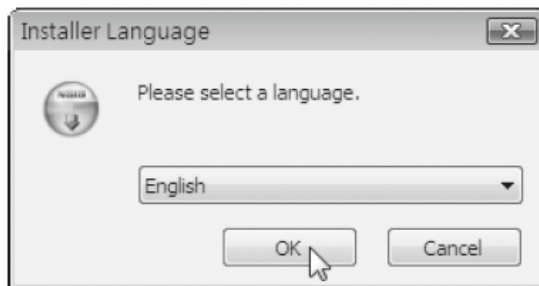


- Dev-C++ 的下載網頁為：

<http://www.bloodshed.net/dev/devcpp.html>，在下載的 <devcpp-4.9.9.2_setup.exe> 檔案（讀者下載時版本可能不同）上按滑鼠左鍵兩下，即可進行安裝：

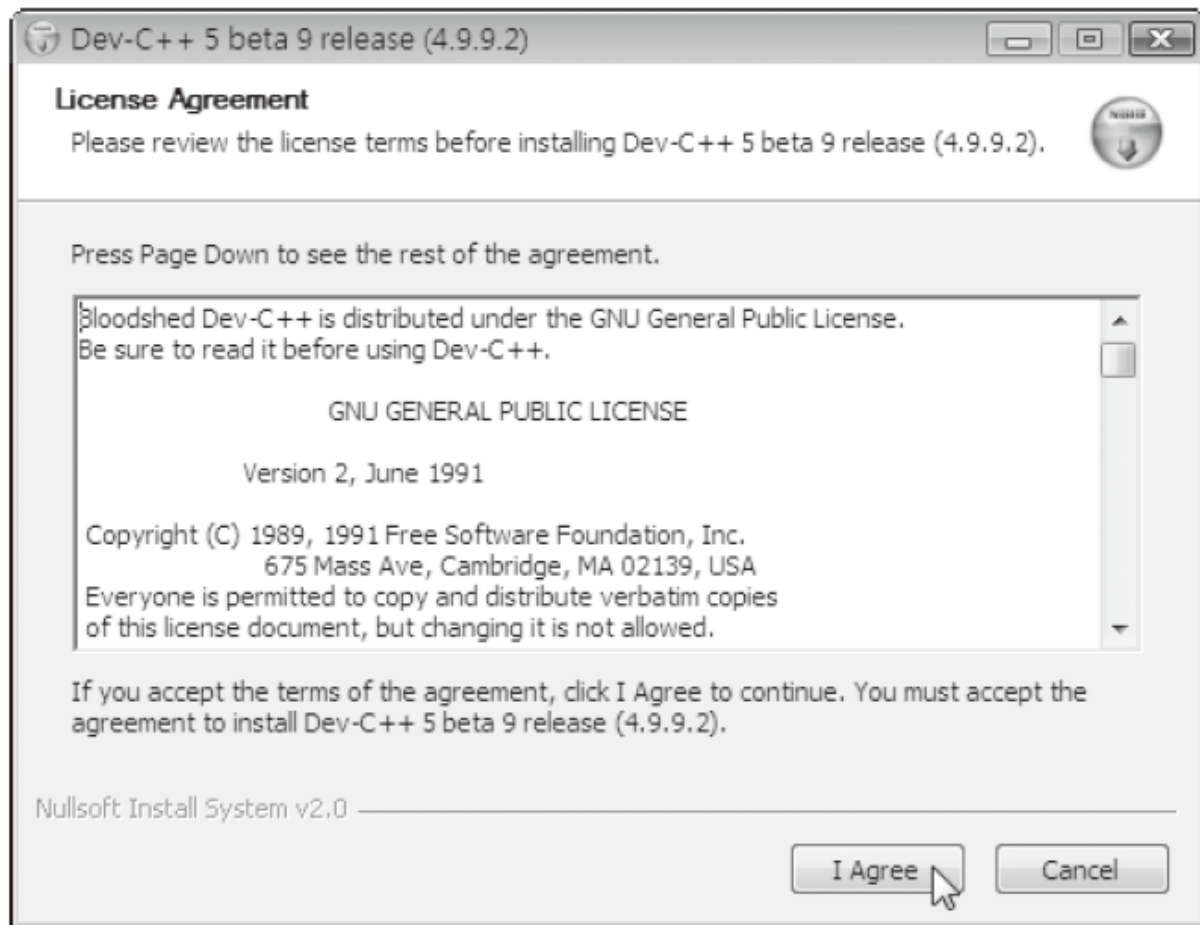


◀ 提醒如有舊版本的 Dev-C++ 存在將無法安裝，按 確定 鈕。



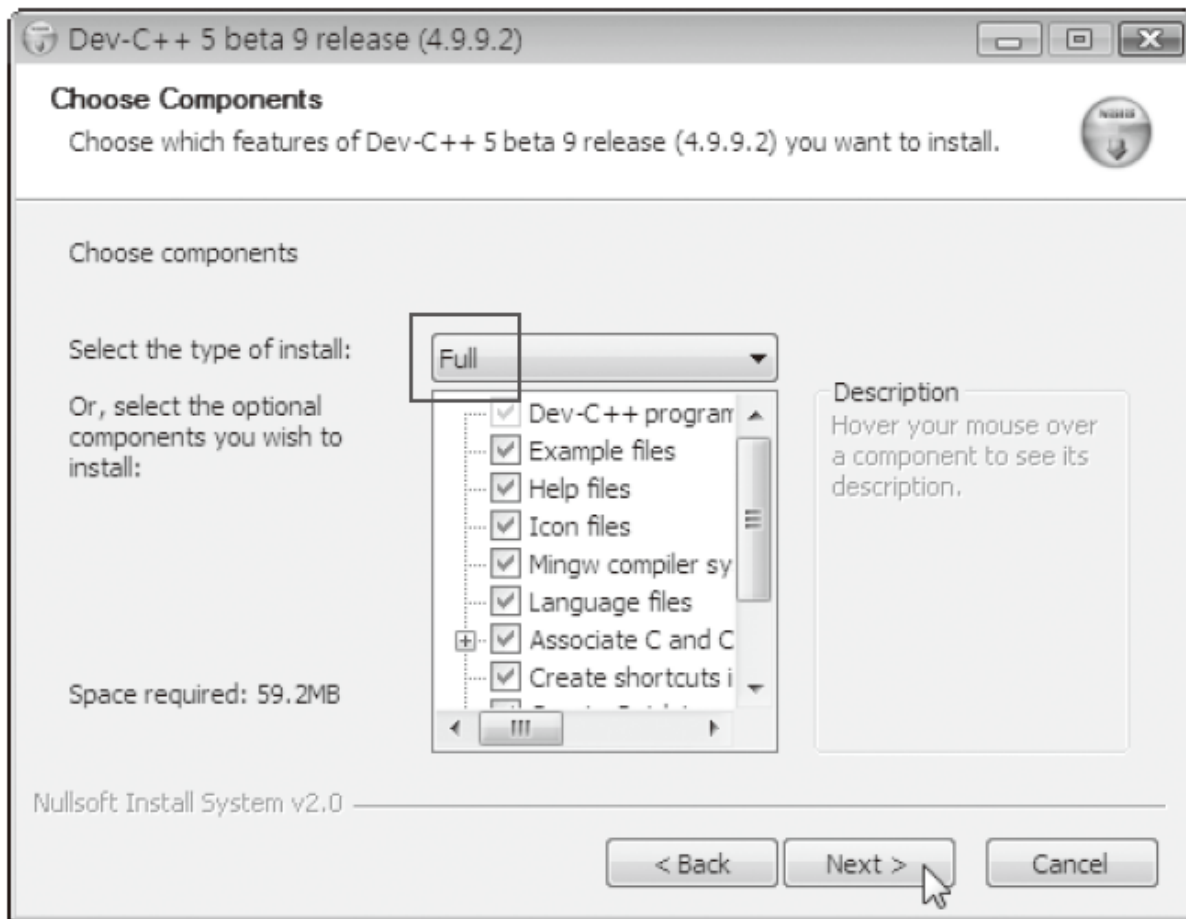
◀ 此處無中文可選擇，先以英文版本安裝，安裝完成後再改成中文。按 OK 鈕。

C++ 的開發環境



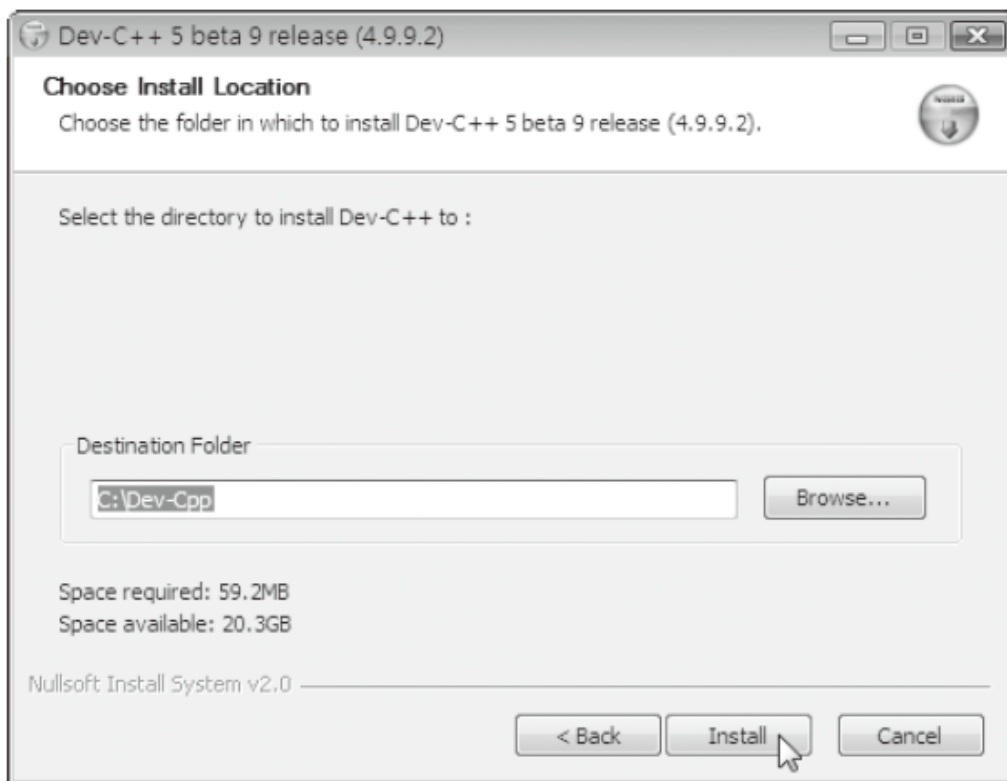
◀ 版權頁，按 **I Agree** 鈕表示同意版權條款。

C++ 的開發環境

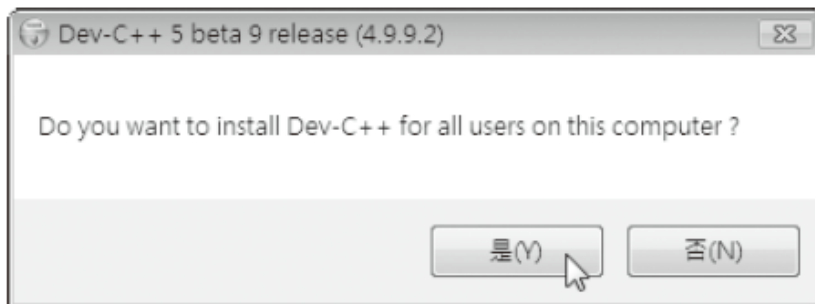


◀ 選擇安裝的元件：選 **Full** 表示安裝全部元件，按 **Next** 鈕。

C++ 的開發環境

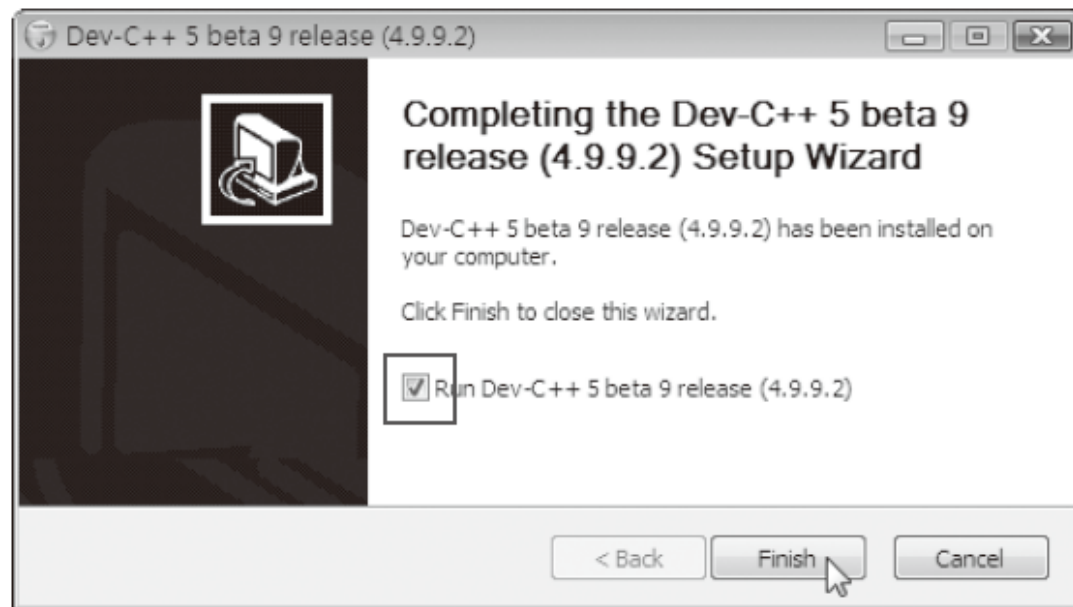


◀ 選擇軟體安裝的路徑：預設為 C 磁碟機的 <Dev-Cpp> 資料夾，按 **Install** 鈕開始安裝。

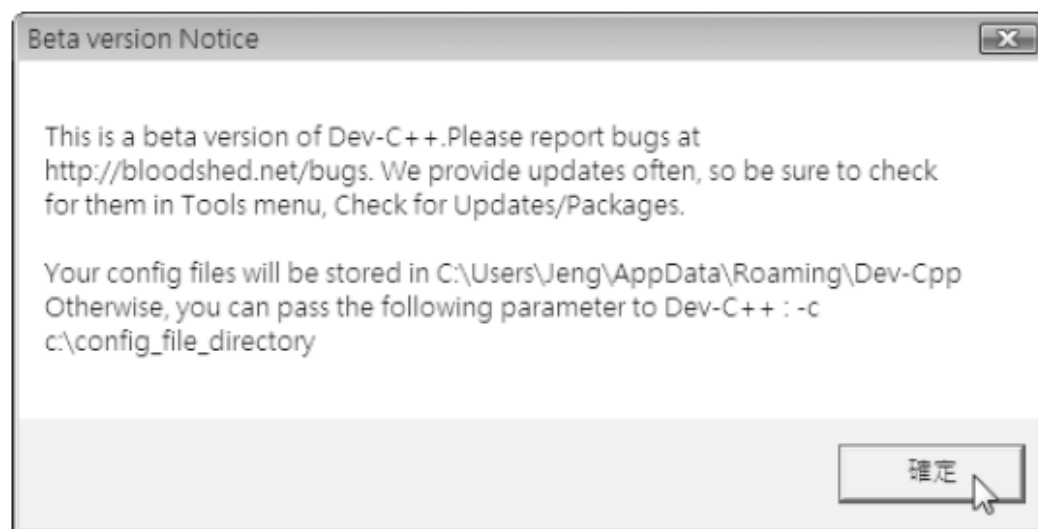


◀ 詢問是否要讓所有使用者都能使用 Dev-C++，按 **是** 鈕。

C++ 的開發環境

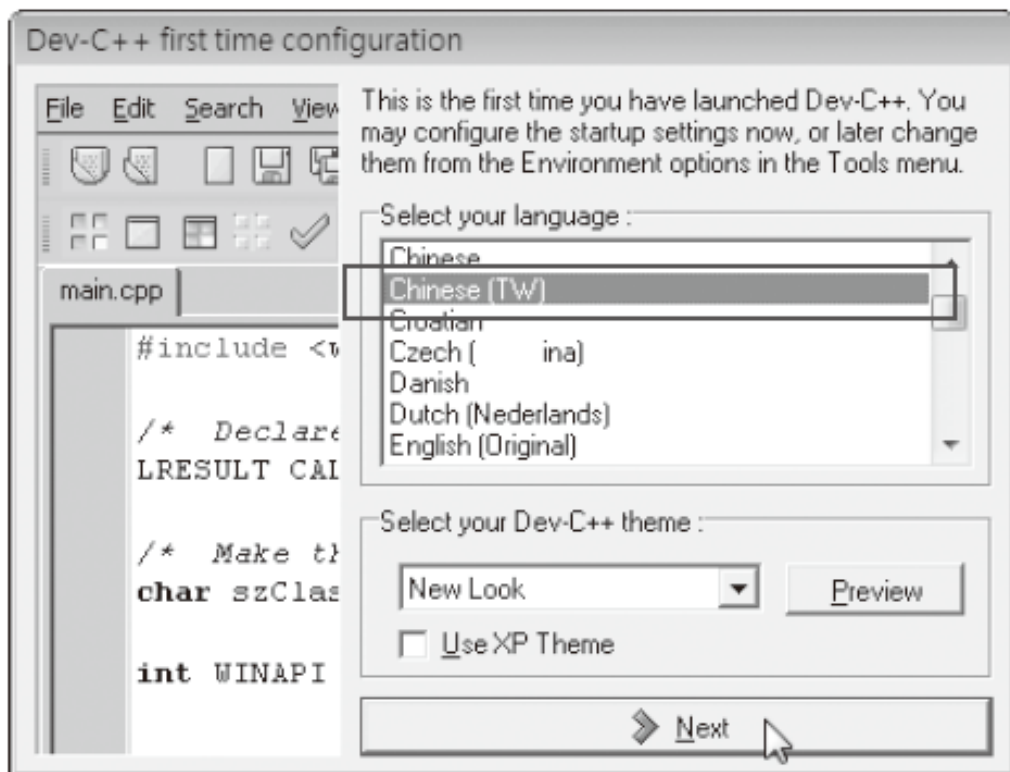


◀ 安裝完成，由於還有部分選項需在第一次執行 Dev-C++ 後設定，故核選 **Run Dev-C++ 5 beta 9 release (4.9.9.2)**，按 **Finish** 鈕，以便立刻執行 Dev-C++。

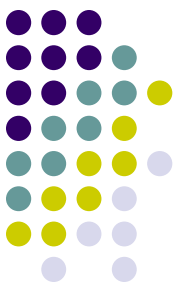


◀ 提醒可在功能表的 **Tool** 項目中搜尋新版本，按 **確定** 鈕。

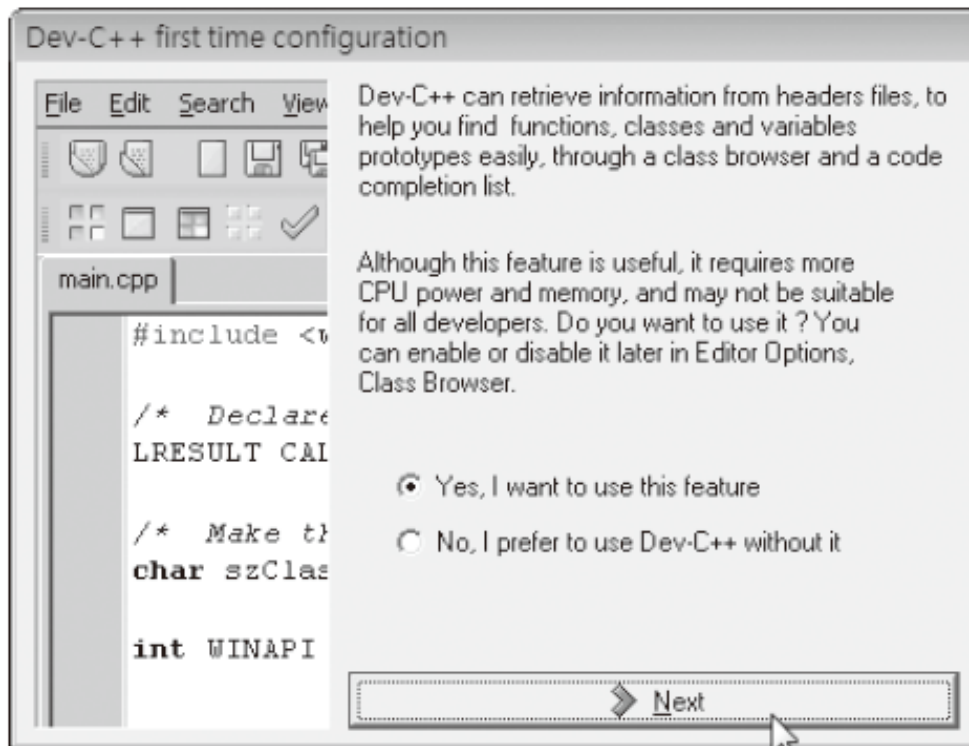
C++ 的開發環境



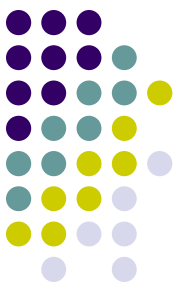
◀ 選擇使用的語言：選 **Chinese[TW]**
表示繁體中文，按 **Next** 鈕。



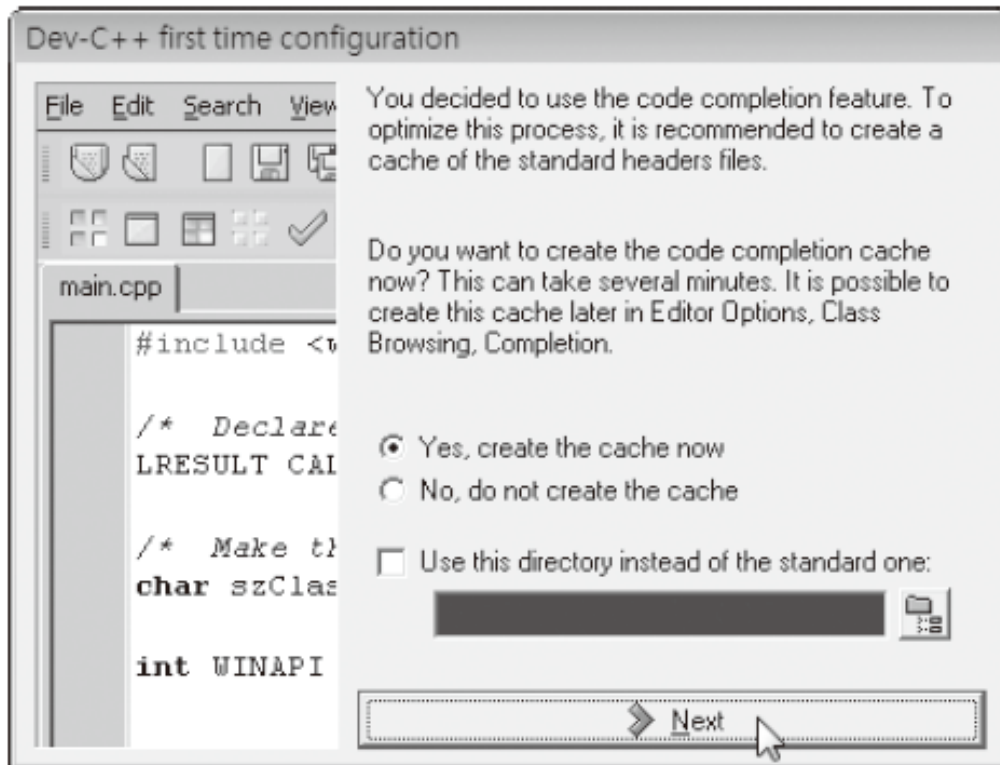
C++ 的開發環境



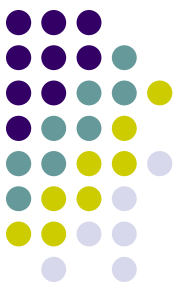
◀ 是否搜尋標頭檔：標頭檔中會包含自建的函式或類別，使用預設的 **Yes, I want to use this feature.**，按 **Next** 鈕。



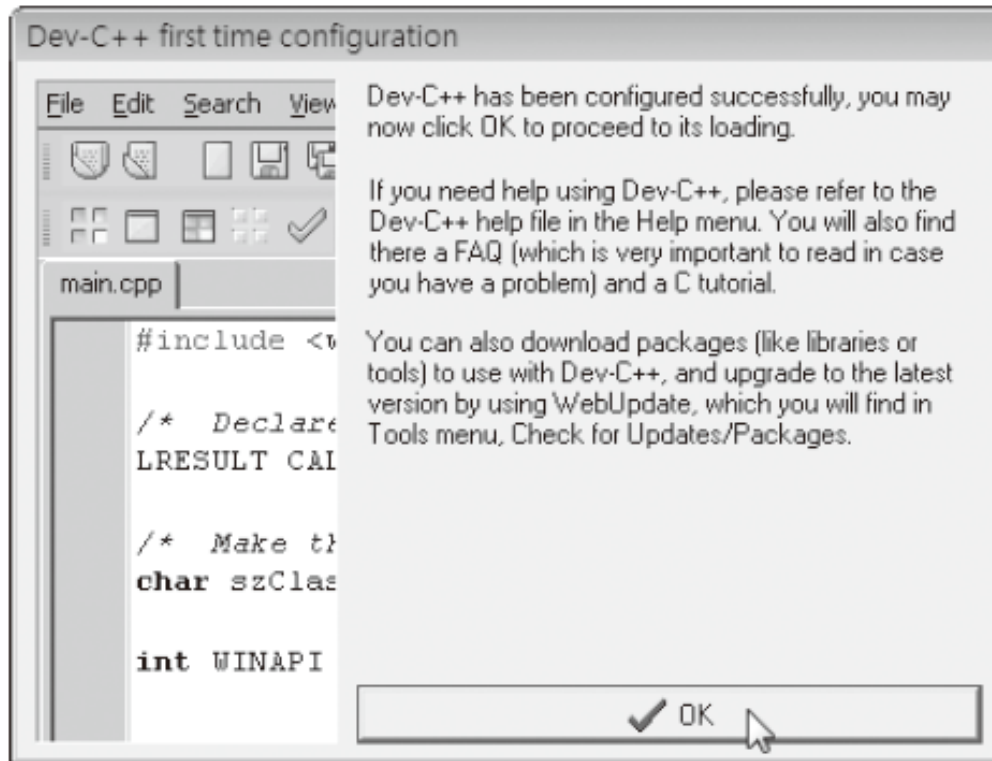
C++ 的開發環境



◀ 是否新增快取：新增快取可增加執行效率，使用預設的 **Yes, create the cache now.**，按 **Next** 鈕。



C++ 的開發環境



◀ 第一次開啓 Dev-C++ 設定完成，按 **OK** 鈕後就可以開始使用 Dev-C++ 了！



C++ 的開發環境

- 微軟公司提供各種語言的免費Express版本，其中包含 Visual C++ 2010Express，使用者可在其網站上自由下載

<http://www.microsoft.com/express/download/#webInstall>)，在下載的 <vcsetup.exe> 檔案上按滑鼠左鍵兩下，即可進行安裝：



C++ 的開發環境



◀ 如果願意將安裝經驗傳送給微軟公司，就核選是，傳送我的安裝經驗相關資訊給 Microsoft Corporation，按下一步鈕。



C++ 的開發環境



◀ 核選 我已閱讀並且接受授權合約中的條款，按下一步 鈕。



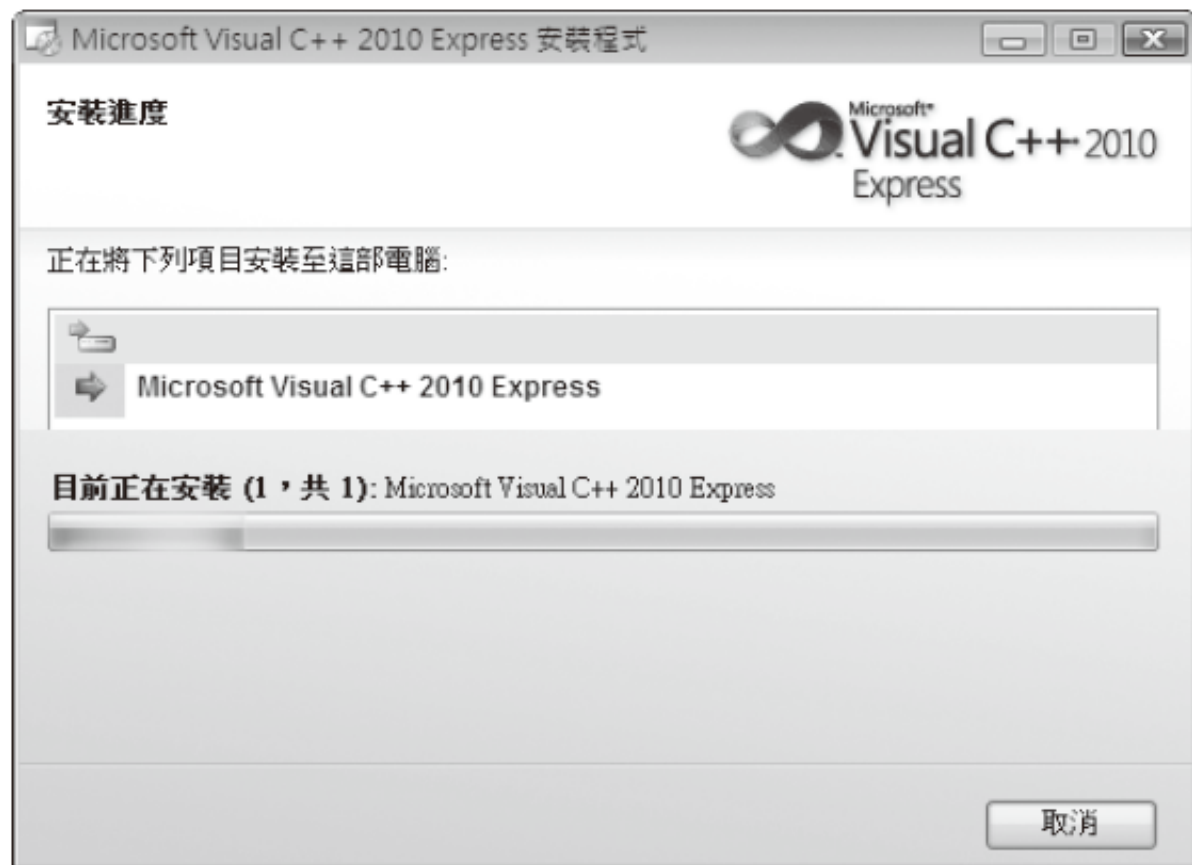
C++ 的開發環境



◀ 出現安裝項目視窗，系統會檢查原先已存在的元件，僅安裝尚未安裝者，您顯示的安裝元件可能與此不同。核選後按 **安裝** 鈕。



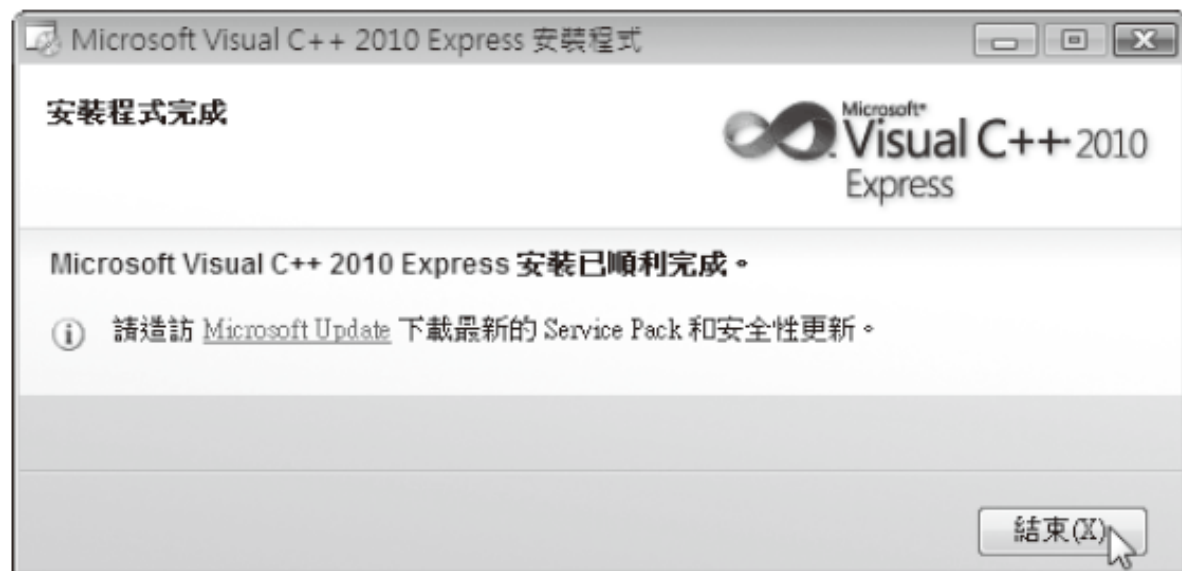
C++ 的開發環境



◀ 安裝過程非常冗長，視安裝者的電腦運作速度及安裝元件數量而定，約需 40 分鐘至 1 小時。



C++ 的開發環境



◀ 按 結束 鈕即安裝完畢，必須重新啓動電腦才能使用 Visual C++ 2010。