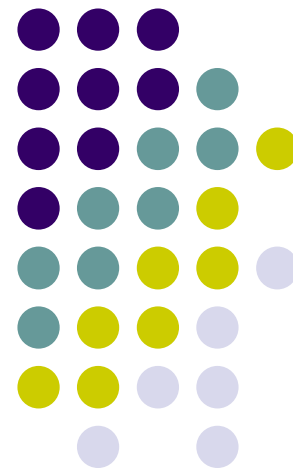


第十八章 檔案處理

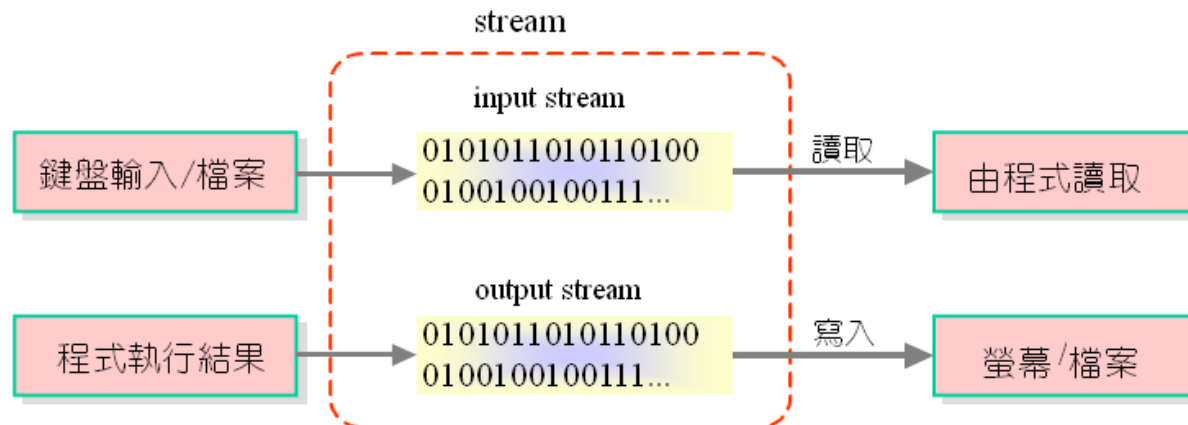
認識檔案的基本觀念
學習檔案的開啟與關閉
熟悉文字檔案的處理
處理二進位檔案





串流

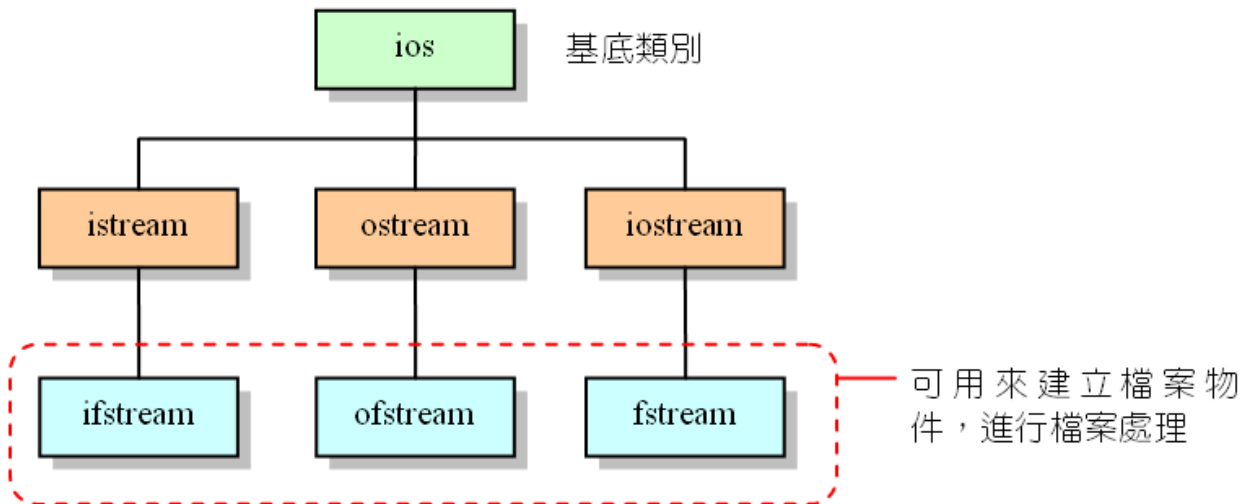
- 串流可分為
 - 「輸入串流」 (input stream)
 - 「輸出串流」 (output stream)
- 下圖說明串流如何做為檔案處理的橋樑

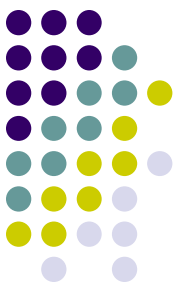




檔案處理類別的繼承圖

- 下圖列出與檔案相關類別的繼承圖





建立檔案物件

- 在開啟檔案之前，須先建立一個檔案物件（file object）
- 檔案物件可分下列三種：
 - 可供寫入資料的檔案物件
 - 可供讀取資料的檔案物件
 - 可供寫入與讀取資料的檔案物件
- 建立檔案物件的語法

`ifstream 物件名稱; // 建立可供讀取資料的檔案物件`

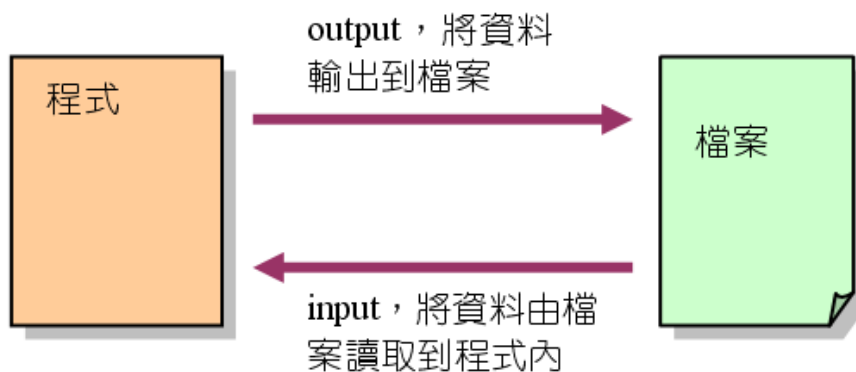
`ofstream 物件名稱; // 建立可供寫入資料的檔案物件`

`fstream 物件名稱; // 建立可供寫入資料與讀取資料的檔案物件`



對照關係

- 下圖是input/output與程式和檔案之間的對照關係





開啟檔案的語法 (1/3)

- 利用open() 函數開啟檔案的語法格式

```
檔案物件.open("檔案名稱",ios::開啟模式);
```

- 利用建構元開啟檔案的語法

```
檔案類別名稱 檔案物件("檔案名稱",ios::開啟模式);
```



開啟檔案的語法 (2/3)

- 「開啟模式」可供選擇的參數如下

ios::開啟模式	說 明
ios::app	開啟可供附加資料的檔案
ios::binary	開啟二進位的輸入/輸出檔案
ios::in	開啟可供讀取資料的檔案
ios::out	開啟可供輸入資料的檔案
ios::trunc	若開啟的檔案已存在，則先刪除它，再開啟檔案

- 範例

```
ifstream inf; // 建立可供讀取資料的檔案物件 inf
inf.open("c:\\test.txt",ios::in); // 開啟可供讀取資料的檔案 test.txt

ifstream inf("c:\\test.txt",ios::in); // 建立物件 inf，並開啟檔案
```



開啟檔案的語法 (3/3)

- 下面的語法可用來測試檔案是否開啟成功

```
ifstream inf("c:\\test.txt",ios::in);    // 建立物件 inf，並開啟檔案
if(inf.is open())                        // 判別檔案是否開啟成功
{
    ...                                // 檔案開啟成功時所做的動作
}
else
{
    ...                                // 檔案開啟不成功時所做的動作
}
```

- 關閉檔案

```
inf.close();    // 關閉以檔案物件 inf 所開啟的檔案
```




將資料寫入文字檔 (1/2)

- 文字檔是由ASCII碼或是純文字所組成的檔案
- prog18_1說明如何將資料寫入文字檔

```
01 //prog18_1, 將資料寫入文字檔
02 #include <fstream> // 載入 fstream 標頭檔
03 #include <iostream>
04 #include <cstdlib>
05 using namespace std;
06 int main(void)
07 {
08     ofstream ofile("c:\\donkey.txt", ios::out); // 建立 ofile 物件
09
10     if(ofile.is open()) // 測試檔案是否被開啟
11     {
12         ofile << "我有一隻小毛驢" << endl; // 將字串寫入檔案
13         ofile << "我從來也不騎" << endl; // 將字串寫入檔案
14         cout << "已將字串寫入檔案..." << endl;
15     }
```

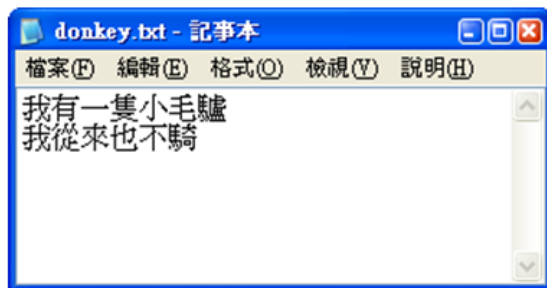


將資料寫入文字檔 (2/2)

```
16     else
17         cout << "檔案開啟失敗..." << endl;
18
19     ofile.close(); // 關閉檔案
20
21     system("pause");
22     return 0;
23 }
```

/* prog18_1 OUTPUT---
已將字串寫入檔案...
-----*/

- donkey.txt的檔案內容：





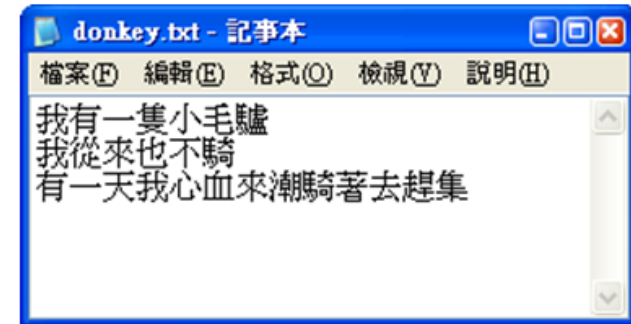
將資料附加到已存在的文字檔

- 將字串附加到已存在的檔案內

```

01 //prog18_2, 將資料附加到已存在的文字檔
02 #include <fstream>
03 #include <iostream>
04 #include <cstdlib>
05 using namespace std;
06 int main(void)
07 {
08     ofstream afile("c:\\donkey.txt",ios::app);    // 建立 afile 物件
09
10     if(afile.is open())                          // 測試檔案是否被開啟
11     {
12         afile << "有一天我心血來潮騎著去趕集";    // 將字串寫入檔案
13
14         cout << "已將字串附加到檔案了..." << endl;
15     }
16     else
17         cout << "檔案開啟失敗..." << endl;
18
19     afile.close();                                // 關閉檔案
20
21     system("pause");
22     return 0;
23 }

```



/* prog18_2 OUTPUT---

已將字串附加到檔案了...

-----*/



從檔案讀入資料

- 下面的範例可以練習從檔案讀入資料

```
01 //prog18_3, 從檔案讀入資料
02 #include <fstream>
03 #include <iostream>
04 #include <cstdlib>
05 using namespace std;
05 int main(void)
06 {
07     char txt[40]; // 建立字元陣列，用來接收字串
08     ifstream ifile("c:\\donkey.txt", ios::in);
09
10     while(!ifile.eof()) // 判別是否讀到檔案的尾端
11     {
12         ifile >> txt; // 將檔案內容寫入字元陣列
13         cout << txt << endl;
14     }
15
16     ifile.close(); // 關閉檔案
17     system("pause");
18     return 0;
19 }
```

/* prog18_3 OUTPUT-----

我有一隻小毛驢
我從來也不騎
有一天我心血來潮騎著去趕集

-----*/

- `get()`、`put()` 與 `getline()` 函數使用的語法

```
檔案物件.put(ch);           // 將 ch 字元變數的值寫入檔案內
```

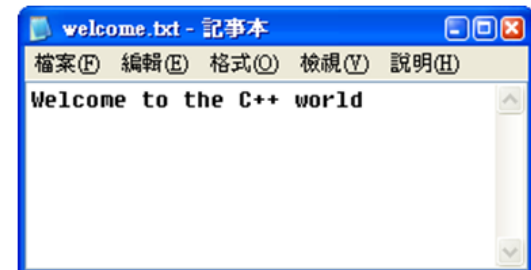


使用put()函數

- prog18_4是利用put() 函數將字串寫入檔案的範例

```
01 //prog18_4, 利用 put() 將字串寫入檔案
02 #include <fstream>
03 #include <iostream>
04 #include <cstdlib>
05 using namespace std;
06 int main(void)
07 {
08     char txt[]="Welcome to the C++ world" ;           // 建立字元陣列
09     int i=0;
10
11     ofstream ofile("c:\\\\welcome.txt",ios::out);
12
13     while(txt[i] != '\0')                             // 判別 txt[i] 字元是否為字串尾端
14     {
15         ofile.put(txt[i]);                             // 將字元 txt[i] 寫入檔案
16         i++;
17     }
18     cout << "字串寫入完成..." << endl;
19     ofile.close();
20
21     system("pause");
22     return 0;
23 }
```

/* prog18_4 OUTPUT---
字串寫入完成...
-----*/



拷貝文字檔案

18.3 文字檔的處理



- 拷貝文字檔案的範例

```
01 //prog18_5, 文字檔的拷貝與讀取
02 #include <fstream>
03 #include <iostream>
04 #include <cstdlib>
05 using namespace std;
06 int main(void)
07 {
08     char txt[80],ch;
09
10     ifstream ifile1("c:\\welcome.txt",ios::in);
11     ofstream ofile("c:\\welcome2.txt",ios::out);
12
13     while(ifile1.get(ch))                // 判別是否讀到檔案的尾端
14         ofile.put(ch);
15     cout << "拷貝完成..." << endl;
16     ifile1.close();
17     ofile.close();
18
19     ifstream ifile2("c:\\welcome2.txt",ios::in);
20     while(!ifile2.eof())                // 判別是否讀到檔案的尾端
21     {
22         ifile2.getline(txt,80,'\n');
23         cout << txt << endl;
24     }
25     ifile2.close();
26
27     system("pause");
28     return 0;
29 }
```

/* prog18_5 OUTPUT-----

拷貝完成...

Welcome to the C++ world

-----*/



二進位檔案

- write() 可將變數寫到二進位檔裡

將變數 var 寫入
二進位檔

變數位址

```
write( (char *) &var, sizeof(var) );
```

強制轉換為指向 char
的指標型態

變數所佔的位元組

- read() 函數可從二進位檔裡讀取資料

由二進位檔讀入資料，
並設給變數 var

變數位址

```
read( (char *) &var, sizeof(var) );
```

強制轉換為指向 char
的指標型態

變數所佔的位元組



二進位檔案的寫入

- 下面的範例是二進位檔寫入的練習

```

01 //prog18_6, 二進位檔寫入的練習
02 #include <fstream>
03 #include <iostream>
04 #include <cstdlib>
05 #include <cmath> // 載入數學函數庫 cmath
06 using namespace std;
07 int main(void)
08 {
09     double num;
10     ofstream ofile("c:\\binary.dat",ios::binary); // 開啟可供寫入的二進位檔
11
12     for(int i=1;i<=5;i++)
13     {
14         num=sqrt((double)i); // 將 i 轉成 double,再計算 sqrt(i)
15         ofile.write((char*)&num,sizeof(num)); // 將 num 寫入二進位檔
16     }
17     cout << "已將資料寫入二進位檔了..." << endl;
18
19     ofile.close(); // 關閉檔案
20
21     system("pause");
22     return 0;
23 }

```

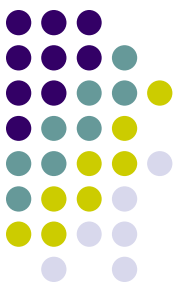


類型: DAT 檔案
修改日期: 2010/6/28 下午 05:54
大小: 40 個位元組

```

/* prog18_6 OUTPUT-----
已將資料寫入二進位檔了...
-----*/

```



讀取二進位檔

- 下面的例子是將資料從二進位檔中讀出

```

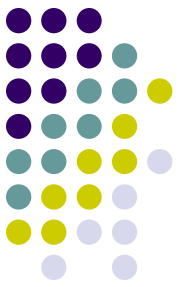
01  //prog18_7, 讀取二進位檔
02  #include <fstream>           // 載入 fstream 標頭檔
03  #include <iostream>
04  #include <cstdlib>
05  using namespace std;
06  int main(void)
07  {
08      ifstream ifile("c:\\binary.dat",ios::binary); // 開啟二進位檔
09      double num;
10
11      for(int i=1;i<=5;i++)
12      {
13          ifile.read((char*) &num,sizeof(num)); // 從二進位檔中讀取資料
14          cout << num << endl;                // 印出讀取的內容
15      }
16      cout << "二進位檔已被讀取了..." << endl;
17
18      ifile.close();                // 關閉檔案
19      system("pause");
20      return 0;
21  }

```

/* prog18_7 OUTPUT----

1
1.41421
1.73205
2
2.23607
二進位檔已被讀取了...

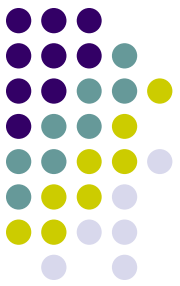
-----*/



將物件寫入二進位檔 (1/2)

- 下面的範例是將物件的內容寫入二進位檔中

```
01 //prog18_8, 將物件的內容寫入二進位檔
02 #include <fstream>
03 #include <iostream>
04 #include <cstdlib>
05 using namespace std;
06 class CStudent
07 {
08     protected:
09         char name[40];
10         int age;
11     public:
12         void get data(void)          // 成員函數，用來輸入物件的資料成員
13         {
14             cout << "Enter name: "; cin >> name;
15             cout << "Enter age: "; cin >> age;
16         }
17         void show data(void)         // 成員函數，用來顯示物件的資料成員
18         {
19             cout << "Name: " << name << endl;
20             cout << "Age: " << age << endl;
21         }
22     };
23
```



將物件寫入二進位檔 (2/2)

```
24  int main(void)
25  {
26      CStudent st;
27      st.get data();
28
29      ofstream ofile("c:\\student.dat",ios::binary);
30
31      ofile.write((char*) &st,sizeof(st));      // 將物件寫入二進位檔中
32      cout << "資料已寫入檔案中..." << endl;
33
34      ofile.close();                          // 關閉檔案
35      system("pause");
36      return 0;
37  }
```

/* prog18_8 OUTPUT---

Enter name: **tippi**

Enter age: **8**

資料已寫入檔案中...

-----*/



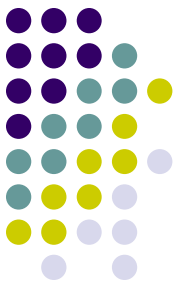
類型: DAT 檔案
修改日期: 2010/6/29 下午 02:31
大小: 44 個位元組



從二進位檔中讀取物件 (1/2)

```
01 //prog18_9, 從二進位檔裡讀取物件的資料
02 #include <fstream>
03 #include <iostream>
04 #include <cstdlib>
05 using namespace std;
06 class CStudent
07 {
08     protected:
09         char name[40];
10         int age;
11     public:
12         void get_data(void)          // 成員函數，用來輸入物件的資料成員
13         {
14             cout << "Enter name: "; cin >> name;
15             cout << "Enter age: "; cin >> age;
16         }
17         void show_data(void)         // 成員函數，用來顯示物件的資料成員
18         {
19             cout << "Name: " << name << endl;
20             cout << "Age: " << age << endl;
21         }
22     };
23
```

- 下面的練習是從二進位檔中讀取物件，顯示結果



從二進位檔中讀取物件 (2/2)

```
24  int main(void)
25  {
26      CStudent st;
27
28      ifstream ifile("c:\\student.dat",ios::binary);
29
30      ifile.read((char*) &st,sizeof(st));
31      st.show data();
32
33      ifile.close();           // 關閉檔案
34
35      system("pause");
36      return 0;
37  }
```

/* prog18_9 OUTPUT---

Name: tippi

Age: 8

-----*/



The End-