



C++教學手冊單數題習題參考答案

博碩文化出版 洪維恩編著

親愛的讀者/您好，

感謝您選購 "C++教學手冊" (pg20153) - 書。

為了服務讀者，我們把本書的單數題習題參考答案置於博碩的網頁裡。如果您發現網路上所提供的解答有誤，或是本書有任何錯誤，請您與我聯繫，唯有您寶貴的意見，本書才得以臻完善。

此外，如果您需要本書完整的習題解答，請您與我連繫。

謝謝您！

洪維恩

wienhong@ms1.ydu.edu.tw

這份文件是於 5/12/2003 所提供



第一頁 參考/解答

1. 請參閱 1.2 節
3. 請參閱 1.4.2 節
5. 請參閱 1.4.2 節
7. 請參閱 1.3 節



3 習題解答



第二章 參考解答

2.1 簡單的例子

1. 請參考 hw2_1.cpp

2.2 C++ 程式解析

3. 請參閱 2.2.4 節
5. 請參考 hw2_5.cpp
7. 請參閱 2.2.3 節
9. 請參考下列宣示

```
int a1,a2,a3;  
char a4;
```

2.3 識別字及關鍵字

11. 請參閱 2.3 節
13. 請參閱 2.3.2 節

2.4 偵錯

15. 請參閱 2.4 節
17. 請參閱 2.5 節



第三頁 參考解答

3.1 變數與常數

1. 下列的幾個常數是不合法的：

- (a) 2B
- (b) 1A
- (d) a5
- (e) \$53
- (g) 2I
- (h) 6.34K

這是因為整數常數不能有數字以外的其它字元，詳細的說明請參閱第三頁的 3-2 與 3-3 頁。

3.2 基本資料型態

3. 請參閱表 3.2.1

- 5. (a) -96.43 --> -9.643E2
- (b) 1974.56 --> 1.97456E3
- (c) 0.01234 --> 1.234E-2
- (d) 0.000432 --> 4.32E-4

7. 請參閱表 3.2.1

9. 請參閱 hw3_9.cpp

因為 100 已設值給 ch 變數，而 ch 變數在宣告時設定為 char，所以在 ASCII code 裡的 100 為 d，即輸出為 d

11. 請參閱 hw3_11.cpp

- 13. (1) unsigned short int
- (2) unsigned int
- (3) double
- (4) float
- (5) float
- (6) unsigned short int
- (7) double
- (8) unsigned short int



5 習題解答



(9) bool

(10) long

3.3 資料型態的轉換

15. 請參閱 3.3.2 節

17. 變數型態在轉換過程中，若 int 型態變數值在 short 型態值可表示範圍內，則不會有錯誤發生，反之則會。

19. 請參閱 hw3_19.cpp

21. 請參閱 hw3_21.cpp

23. 請參閱 hw3_23.cpp

25. 請參閱 hw3_25.cpp



第四章 參考解答

4.1 運算式與運算子

1. (a) 輸出結果：a=9 a=9
(b) 輸出結果：a=0 b=3
(c) 輸出結果：a=0 b=15
3. 請參閱 4.1 節
5. 請參閱 表 4.1.3 節
7. 請參閱 4.1.4 節
9. 請參閱 4.1.6 節

4.2 運算子的優先順序

11. 請參閱下表的執行結果：

題號	敘述	結果
(1)	$6+4<9+12$	true
(2)	$16+7*6+9$	67
(3)	$(13-6)/7+8$	9
(4)	$7>0 \ \&\& \ 6<6 \ \&\& \ 12<13$	false
(5)	$8>0 \ \ 12<7$	true
(6)	$8<=8$	true
(7)	$7+7>15$	false
(8)	$19+34-6>4$	true
(9)	$12+7>0 \ \ 13_5>6$	true
(10)	$3>=5$	false

4.3 運算式

13. 請參閱 hw4_13.cpp
15. 請參閱 hw4_15.cpp
17. 請參閱 hw4_17.cpp



7 皆題解答



4.4 運算式的型態轉換

19. 請參閱 4.4 節



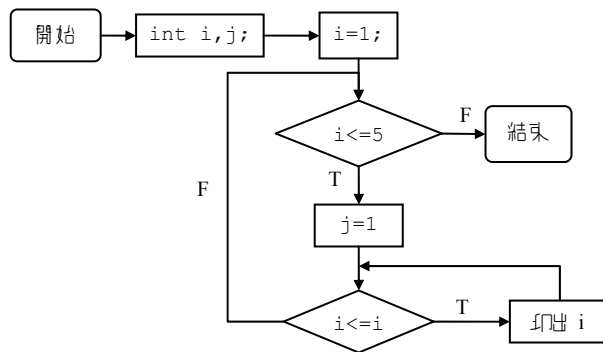
第 11 頁 參考/解答

5.1 程式的結構設計

1. 請參閱 5.1.1 節
3. 請參閱 5.1.3 節

5.2 迴圈

5. 請參閱 5.2.2 節
7. 請參考 hw5_7.cpp



9. 請參考 hw5_9.cpp

5.3 迴圈的跳躍

11. 請參閱 5.3.1 節
13. 請參考 hw5_13.cpp

5.4 我的程式會轉彎—選擇性敘述

15. 請參閱 5.4.2 節
17. 請參考 hw5_17.cpp
19. 請參考 hw5_19.cpp
21. 請參考 hw5_21.cpp



9 習題解答



23. 請參攷 hw5_23.cpp

25. 請參攷 hw5_25.cpp

27. 請參攷 hw5_27.cpp

29. 請參攷 hw5_29.cpp

31. 請參攷 hw5_31.cpp

5.5 夏好月的多重選擇—switch 敘述

33. 請參閱 5.5 節

35. 請參攷 hw5_35.cpp



第六章 參考解答

6.1 函數的基本架構

1. 請參考/ hw6_1.cpp
3. 請參考/ hw6_3.cpp
5. 請參考/ hw6_5.cpp
7. 請參考/ hw6_7.cpp
9. 請參考/ hw6_9.cpp
11. 請參考/ hw6_11.cpp

6.2 特殊的 inline 函數

13. 請參考/ hw6_13.cpp
15. 請參考/ hw6_15.cpp

6.3 變數的等級

17. 請參考/ hw6_17.cpp

6.4 同時使用多個函數

19. 請參考/ hw6_19.cpp
21. 請參考/ hw6_21.cpp
19. 請參考/ hw6_19.cpp
21. 請參考/ hw6_21.cpp



第七章 參考解答

7.1 參照與函數

1. 請參攷 hw7_1.cpp
3. 請參攷 hw7_3.cpp
5. 請參攷 hw7_5.cpp

7.2 函數的多載

7. 請參攷 hw7_7.cpp
9. 請參攷 hw7_9.cpp
11. 請參攷 hw7_11.cpp

7.3 參數的預設值

13. 請參攷 hw7_13.cpp
15. 請參攷 hw7_15.cpp

7.4 前置處理器__#define

17. 請參攷 hw7_17.cpp
19. 請參攷 hw7_19.cpp

7.5 前置處理器__#include

21. 請參攷 hw7_21.cpp

7.6 命令列引數的使用

23. 請參攷 hw7_23.cpp
25. 請參攷 hw7_25.cpp



第八章 參考解答

8.1 一維陣列

1. (d)

3. 修正如下：

(a) `int arr[]={1,2,3,4,5,6};`

(b) `int arr[6]={1,2,3,4,5,6};`

(c) `int arr[][6]={1,2,3,4,5,6};`

(d) `float arr[]={1,2.2,3,6.4,5.5};`

5. (a)

7. 變數型態為 `int`，陣列名稱為 `arr`，陣列大小為 20 (4*5)

9. 修正如下：

(a) `float num[30];` // 陣列大小，應用大括號括起來

(b) `int A[7]={5};` // 陣列名稱不能使用關鍵字及設值 應用大括號括起來

(c) `int array[6];` // 陣列大小，應用大括號括起來

11. 請參考 `hw8_11.cpp`

13. 請參考 `hw8_13.cpp`

8.2 二維陣列以及多維陣列

15. 請參考 `hw8_15.cpp`

17. 請參考 `hw8_17.cpp`

8.3 傳遞陣列給函數

19. 請參考 `hw8_19.cpp`

21. 請參考 `hw8_21.cpp`

8.4 字串

23. `'a'` 表字元，`"a"` 表字串

25. 請參考 `hw8_25.cpp`



8.5 字串的輸出與輸入

27. 請參考 hw8_27.cpp

29. 請參考 hw8_29.cpp

8.6 常用的字串處理函數

31. 請參考 hw8_31.cpp

33. 請參考 hw8_33.cpp

35. 請參考 hw8_35.cpp

37. 請參考 hw8_37.cpp

8.7 字串陣列

39. 請參考 hw8_39.cpp

41. 請參考 hw8_41.cpp



第 9 章 參考解答

9.1 指標概述

1. 請參考 hw9_1.cpp

9.2 指標變數

3. `int *ptr;` 應修改為 `double *ptr;`，因為指標所指向的型態，必須和之前宣告的型態相同 (或是將 `double f;` 修改為 `int f;`，只要指標與所指向變數的型態相同即可)

`ptr=f;` 應修改為 `ptr=&f;`。也就是說，如要將 `f` 的地址設給指標 `ptr` 存放，則在變數 `f` 前要加上 `&`

5. (3)

9.3 指標運算子

7. (1) `*ptr`，取出 `ptr` 所指向的值，即 `i` 的值
(2) `&ptr`，取出指標變數 `ptr` 所在記憶體的位置
(3) `&i`，取出變數 `i` 所在記憶體的位置
(4) `ptr`，指標變數

9.4 指標的運算

9. `double a, *p1, *p2;`

`p1=p2=&a;`

11. (3)

13. 請參閱 9.4.3 節

9.5 指標與函數

15. 請參考 hw9_15.cpp

17. 請參考 hw9_17.cpp

19. 請參考 hw9_19.cpp

9.6 指標與陣列



15 習題解答



21. 請參攷 hw9_21.cpp

23. 請參攷 hw9_23.cpp

25. 請參攷 hw9_25.cpp

27. 請參攷 hw9_27.cpp

9.7 指向指標的指標

29. 請參攷 hw9_29.cpp

9.8 動態記憶體配置

31. 請參攷 hw9_31.cpp

9.9 指標與參照

33. 請參閱表 9.9.1 節



第十章 參考解答

10.1 結構

1. 請參考 hw10_1.cpp
3. 請參考 hw10_3.cpp

10.2 以結構為引數傳遞到函數

5. 請參考 hw10_5.cpp
7. 請參考 hw10_7.cpp

10.3 共用空間

9. 請參考 hw10_9.cpp

10.4 列舉型態

11. 請參考 hw10_11.cpp
13. 請參考 hw10_13.cpp
15. 請參考 hw10_15.cpp

10.5 使用自定的型態—typedef

17. 請參考 hw10_17.cpp



第十一章 參考解答

11.1 識別類別

1. 請參閱 11.1 節
3. 請參閱 11.1.2 節
5. 請參閱 hw11_5.cpp

11.2 撰寫函數成員

7. 請參閱 11.2 節
9. 請參閱 hw11_9.cpp
11. 請參閱 hw11_11.cpp
13. radius 設定為 double 型態，但傳入 set 函數卻為 int 型態，所以無法將 r 值設定給 radius

11.3 函數成員引數的傳遞

15. 請參閱 hw11_15.cpp
17. 請參閱 hw11_17.cpp

11.4 公有成員與私有成員

19. 請參閱 hw11_19.cpp
21. 此題犯了「語意」的錯誤，要攔截此錯誤，請參閱 hw11_21.cpp。

11.5 友函數

23. 請參閱 hw11_23.cpp。將 area() 改為友函數的好處並不大，因為 prog11_10 中，所有的資料成員皆為 public，因此不必是友函數也可以存取到它們。



第十二章 參引解答

12.1 建構元

1. 請參閱 12.1 節
3. 加V 沒有引數的建構元即可，請參引 hw12_3.cpp

12.2 建構元引數的預設值

5. 建構元的使用，要避免模稜兩可的情況發生，即如沒有引數的建構元去掉即可。請參引 hw12_5.cpp

12.3 物件的進階處理

7. 請參引 hw12_7.cpp

12.4 類別裡的靜態成員

9. 請參閱 12.4 節
11. 請參閱 12.4 節
13. 請參引 hw12_13.cpp
15. 請參引 hw12_15.cpp

12.5 指向物件的指標與參照

17. 請參引 hw12_17.cpp
19. 請參引 hw12_19.cpp



第十三章 參考解答

13.1 解構元

1. 請參閱 13.1 節
3. 請參閱 hw13_3.cpp

13.2 動態記憶體配置與解構元

5. 請參閱 hw13_5.cpp
7. 請參閱 hw13_7.cpp

13.3 拷貝建構元

9. (1) 第 17 行
(2) 可以正確執行，因為類別裡並沒有使用到動態記憶體配置，也沒有把物件當成引數傳入函數中，因而本例可以正確執行。
(3) 請參閱 hw13_9.cpp



第十四章 參考解答

14.1 認識運算子的多載

1. 請參考/ hw14_1.cpp
3. 請參考/ hw14_3.cpp
5. 請參考/ hw14_5.cpp

14.2 加號「+」的多載

7. 請參考/ hw14_7.cpp
9. 請參考/ hw14_9.cpp
11. 請參考/ hw14_11.cpp
13. 請參考/ hw14_13.cpp

14.3 設定運算子「=」的多載

15. 「=」運算子的多載只能以類別的成員函數來撰寫，不能以友元函數，或是一般的函數撰寫。
17. 請參考/ hw14_17.cpp
19. 請參考/ hw14_19.cpp



第十章 參考解答

15.1 繼承的基本概念

1. 請參閱 15.1 節
3. 請參閱 hw15_3.cpp
5. 請參閱 hw15_5.cpp
7. 請參閱 hw15_7.cpp
9. 請參閱 hw15_9.cpp

15.2 在類別存取類別的成員

11. 請參閱 hw15_11.cpp
13. 請參閱 hw15_13.cpp
15. 請參閱 hw15_15.cpp
17. 請參閱 hw15_17.cpp
19. 可以，因為 `get_id()` 為 `public` 成員，繼承到 `CTextWin` 後還是 `public`，所以可以在 `main()` 裡呼叫到它。

15.3 改寫

21. 請參閱 hw15_21.cpp

15.4 在類別中使用拷貝建構子

23. 執行時會發生錯誤，因為在執行子類別的拷貝建構元之前，會先執行父類別的拷貝建構元。



第十六章 參引解答

16.1 虛擬函數

1. 請參引 hw16_1.cpp

16.2 指向基址類別的指標

3. 請參引 hw16_3.cpp

16.3 抽象類別與虛擬函數

5. 請參引 hw16_5.cpp
7. 此錯誤是因為 CShape 為抽象類別，我們不能以抽象類別來建立物件之故。
9. 請參引 hw16_9.cpp

16.4 抽象類別於多層繼承的應用

11. 請參引 hw16_11.cpp

16.5 虛擬解構元

13. 請參閱 16.5 節
15. 因為在 prog16_9 裡，所有類別裡的資料成員都沒有以動態記憶體的方式來配置，因而不需在解構元裡撰寫銷毀記憶體的動作。



第十七章 參考解答

17.1 基本概念

1. 請參閱 17.1 節

17.2 檔案的開啟與關閉

3. 請參閱 17.2 節

17.3 V 字檔的處理

5. 請參閱 hw17_5.cpp
7. 請參閱 hw17_7.cpp
9. 請參閱 hw17_9.cpp

17.4 二進位檔的處理

11. (a) 請參閱 hw17_11a.cpp
(b) 請參閱 hw17_11b.cpp

綜合練習



第十八章 參引解答

18.1 名稱空間

1. 請參閱 18.1 節
3. 請參引 hw18_3.cpp
5. (a) 主程式 main() 裡有和名稱空間相同的變數名稱時，會以 main() 的變數優先使用，否則使用名稱空間裡的變數，必須加上名稱空間的名稱
(b) 不會有影響，理相同
(c) 請參引 hw18_5(c).cpp

18.2 例外處理

7. 請參閱 18.2.1 節
9. 請參引 hw18_9.cpp

18.3 樣板

11. 請參引 hw18_11.cpp
13. 請參引 hw18_13.cpp
15. 請參引 hw18_15.cpp

18.4 大型程式的變長與條件式編譯

17. 請參引 hw18_17.cpp
19. 請參引 hw18_19.cpp
21. 請參引 hw18_21.cpp