



SoftTip 尖端電腦

榮國電腦軟體研究開發部

NO.61820

資料結構——以C實作

(上冊)

榮國R&D劉裕祥等編著
陳振南老師校閱

版
權
所
有



翻
印
必
究

製 作 : 螢 圃 電 腦
發 行 者 : 周魏筱玉
發 行 所 : 台北市和平西路一段84號10樓(螢圃大廈)
信 件 請 寄 : 台北郵政 6-52 號信箱
郵政劃撥帳號: 07923480 戶名: 魏筱玉
FAX : (02)365-6584

BIC 6182

中華民國八十年十月出版
出版登記證: 局版臺業字第四二八九號

TRADEMARKS:

Microsoft, MS-DOS, OS/2, Multiplan, Softcard, XENIX, Microsoft Press, Microsoft Windows, QUICK C, QUICK BASIC, NASM are registered trademarks of Microsoft Corporation. FRED, Runtime+, dBASE III PLUS, dBASE CTOOLS, MicroNaze, Framework, TimeFrame, FrameLock, and Framework II are trademarks of Ashton-Tate. Apple is a registered trademark of Apple Computer, Inc. MacBASIC is a trademark of Apple Computer, Inc. Macintosh is a trademark of Macintosh Laboratory, Inc. IBM, Personal Edit II, PC/XT, PC AT, PS/2, are registered trademarks of International Business Machines Co. Commodore 64 is a trademark of Commodore. UNIX is a trademark of Bell Laboratories. Tandy is a trademark of Tandy Corporation. Microsoft is a registered trademark of Microsoft, Inc. Lattice is a trademark of Lattice, Inc. Aztec is a trademark of Hanx software. Epson is a registered trademark of Epson. Hercules is a trademark of Hercules Computer Technology. Easy 3D is a trademark of Enabling Technologies, Inc. Wordstar is a registered trademark of MicroPro Int'l Corp. Lotus 1-2-3 is a registered trademark of Lotus Development Corp. R:base is a trademark of Microrim. Turbo, Turbo Basic, Turbo Pascal, Turbo C, TURBO C++, Microcalc, SideKick, Superkey, BORLAND are trademarks of Borland, Inc.

● 本書資料曾參考下列書籍 ●

ADVANCE C FRIDIRNER ++
ADVANCED C
ADVANCED C PROGRAMMING
ADVANCED GRAPHIC IN C
ADVANCED TURBO C
BASIC TO C
C BY DISSECTION
C DATA BASE DEVELOPMENT
C FOR PROGRAMMER'S
C LANGUAGE FOR PROGRAMMER'S
C PRIMER PLUS
C THE COMPLETE REFERENCE
C TOOLBOX
C TRILOGY
CRAFTING C TOOLS
FROM BASIC TO C
GOING FROM BASIC TO C
IBM PC THE COMPLETE C
TURBO C++ PROGRAMMER'S GUIDE
TURBO C++ GETTING STARTED
USING TURBO C++

INTERACTIVE C
LATTICE TOP VIEW TOOLS BASKET
LEARNING TO PROGRAMMING IN C
MASTERING IN C
NSC 5.X QUICK REFERENCE
PASCAL TO C
PROFICIENT C
SUPERCHARGING C WITH ASSEMBLY
SUREFIRE PROGRAMMING IN C
THE C LIBRARY
THE C PROGRAMMER'S HANDBOOK
THE ILLUSTRATED C PROGRAMMING
THE SPIRIT OF C
TURBO C DOS UTILITIES
TURBO DEBUGGER USER'S GUIDE
USING TURBO C
VARIATION IN C
TURBO C++ USER'S GUIDE
TURBO C++ LIBRARY REFERENCE
GRAPHICS PROGRAMMING IN TURBO C

親愛的讀者：

一本書的好壞和成敗，與規劃者、作者群、出版社、編輯、校對，到印刷...，以及讀者的批評、指教等，均息息相關。

「瑩圃電腦」經過數年的辛苦播種、耕耘，所出版的書籍，愈來愈受外界好評與肯定；其中最大的功勞，還是屬於花錢購買「瑩圃」書籍的讀者、電腦教師以及榮譽會員們。

由於他們不吝惜花費寶貴的時間，熱忱地來信批評、指正書中的缺點，諸如提供最新資訊加入書籍中、同類書籍優缺點比較、錯誤勘正...等，及其構想與建議；甚至還有讀者主動提供其研讀中所寫的範例，做為再版書籍之參考；而「瑩圃」的書籍就是在這些幕後功臣的呵護、鞭策下日益進步！

在讀者的來信中，我們也發現，其中有許多寶貴的意見與資料，使得若干書籍脫胎換骨，而重獲新生。

其實，書也像人一樣，「生」固然重要，但「教」與「養」，更是重要！過去，「瑩圃」一直不曾向那些默默愛護、照顧「瑩圃」書籍的人公開道謝，這是一項錯誤！

所以，現在「瑩圃」要公開感謝所有曾經為「教」、「養」「瑩圃」書籍而付出心血的讀者與老師們，除致上薄酬外，「瑩圃」並與他們共享每本書的成就。

「瑩圃」將於每本新書中特闢一頁感謝函，向該書之重要功臣公開致意，感謝其付出的辛勞；同時並寄贈該書十本，予其指定之親友。

祝您每天都有更大的成就
DAILY BONANZA TO YOU !

瑩圃電腦 總經理

周錦文 敬啓

BALA CHOW

感謝函

本書書名為資料結構--以C 實作(上冊)是本公司精心為讀者製作的書籍，在本書中完整的將資料結構作一次全盤而詳盡的編排不但內容充實健全，在排版上本公司更引進最新的排版技巧不但力求版面精美，字跡端正清晰，而且累積全部工作人員心血通力製作，歷時良久，相信必能對各位讀者發揮莫大的幫助。

凡此種種，均非「瑩圃」之功勞；相信每一位閱讀了本書的人，都會與「瑩圃」一樣，下列的每一位；因著他們的大方，願意與其他人分享其智慧和能力，讀者們才得以大飽眼福，滿足求知慾。

在此我們感謝瑩圃 R&D劉裕祥先生細心地撰寫這本書的內容及程式。更特別感激銘傳商專陳振南老師由本書之架構規劃、要點提示，到內容之審閱指正等等，數個月來一直本著教育家的熱忱為使莘莘學子能有更好的書可讀，甚至不計酬勞代價，更讓我們由衷欽佩，特此銘謝。

祝您每天都有更大的成就

DAILY BONANZA TO YOU !

瑩圃電腦 總經理

周錦文 敬啓

BALA CHOW

本書簡介

書號：68120

開數：16 開 頁數：560 頁

適合讀者：大學、專科高職學生略具 C 語言基礎者

所需硬體：IBM PC AT 或 PC XT

所需軟體：C Compiler

書附磁片：壹片（含本書中之 C 原始程式檔）

本書特點：

1. 書內範例程式均為可執行之完整程式
 2. 書內範例程式均附在磁片上，易於教學。
 3. 本書收集目前有多多的資料結構，取其精華，編輯而成。
 4. 範例及圖解最多
 5. 範例程式均含流程圖以方便讀者學習，了解程式。
 6. 每章均含重點整理及習題作業。
-

內容概要：

第 1 章 導論

何謂資料結構/C 與資料結構/演算法與演算法分析/Big-Oh 函數/書中程式使用說明/重點整理/

第 2 章 C 之陣列與結構

陣列/陣列之應用/兩矩陣相加/轉置矩陣/特殊矩陣表示法/稀疏矩陣/上三角形與下三角形表示法/對稱矩陣/陣列之應用—多項式相加/C 之結構與組合/結構/typedef/結構指標/組合/結論/專題討論
——製作處理集合功能之函式/重點整理/

第 3 章 堆疊與佇列

堆疊之定義與操作/使用陣列與結構來製作堆疊/另一種方式的堆疊運算/多個堆疊/堆疊之應用/前序、中序及後序表示法/將中序式轉成後序式之演算法/將中序式轉成後序式之程式/由後序式求值/八

個皇后問題 (Eight Queen Problem) / 迷宮問題 (Maze Problem) /
設定與儲存一個迷宮:/尋找之方向:/ 找尋路徑:/佇列/使用陣列與
結構來製作佇列/單純之佇列操作/ 環狀佇列之操作/用雙佇列來解
決特權問題/訂位系統/結語/重點整理/

第 4 章 鏈結串列

指標與動態記憶體分配/指標與位址/指標與陣列、字串及指標的運
算/動態記憶體分配/使用 malloc/ 使用 calloc/強制改型 (cast)
/NULL 指標/歸還 (free) 記憶體/ 指標的其它論題/指標陳列/型
態不一致/鏈結串列之定義與製作/ 使用陣列來製作鏈結串列/使用
指標來製作鏈結串列/ 鏈結串列之操作/建立一新串列/加入一新節
點/將節點加入串列之前面/ 加入一節點到串列的尾端/在串列中插
入一節點/刪除一節點/刪除串列的第一個節點/ 刪除串列最後一個
節點/刪除特定的節點/ 使用鏈結串列來製作堆疊與佇列/製作堆疊
/製作佇列/串列的搜尋與串列的其它運作/串列的搜尋/串列的其它
操作/將兩串列合併 /清除一串列/將串列反轉 (invert)/多項式相
加/環狀鏈結串列(Circular Linked List) /環狀串列之操作/建立
一個環狀串列/在環狀串列中加入一節點/ 環狀串列之追蹤/刪除環
狀串列中的節點/環狀串列之回收/ 約瑟夫 (Josephus) 問題/雙向
鏈結串列 (Doubly Linked List)/ 雙向鏈節串列之操作/加入節點
/刪除節點/串串列 (Lists of Lists)/ 重點整理/

第 5 章 函式之遞迴與選擇

遞迴 (Recursion)/迭代 (Iteration)/遞迴與迭代之選擇/ 遞迴之
實際應用/兩個有關數學運算之函式/河內塔 (Tower of Hanoi)/八
個皇后問題/騎士問題/遞迴的二分搜尋 (Binary Search) /函式指
標 (Function Pointer)與函式的選擇 (Function Selection)/重
點整理/

第 6 章 樹狀結構(Tree Structure)

樹狀結構之觀念與定義/樹之表示法與儲存 /二元樹/二元樹的儲存
/用一維陣列來儲存二元樹/ 用鏈結方式來儲存二元樹/二元樹之運

作與追蹤/ 在二元樹中加入一節點/完全平衡樹的建立/二元搜尋樹節點之刪除/ 二元樹的追蹤 (traversal)/二元樹之應用/二元樹之完整操作/ 二元樹排序 (sort) /二元算式樹 (Binary Expression Tree)/二元樹的其它運作/決定唯一的二元樹/如何將一般樹與森林化成二元樹/複製一棵二元樹/ 比較二棵二元樹是否相等/引線二元樹/ 其餘有關樹的論題/有關 m 元樹 (m -ary tree)的一些問題/樹的其它應用/ 決策樹/ Game Tree/重點整理/

附錄A C之關鍵字(keyword)

A.1 標準C的關鍵字/A.2 MicrosoftC之關鍵字/A.3 TurboC之關鍵字/

附錄B 運算子之優先權與運算順序

附錄C C語法總結(syntaxsummary)

C.1 運算式/C.2 宣告(Declaration)/C.3 敘述/C.4 外部定義 (External definitions)/C.5 前處理器(Preprocessor)/

附錄D 建議參考書籍

D.1 語言程式設計/D.2 有關資料結構方面之書籍/D.3 編譯器/

附錄E 上冊總結—程式實作

堆疊的應用——算式編譯器(Expression Compiler)/鏈結串列的應用——使用單向鏈結串列來製作一行編輯器(Line Editor)/練習題/

相關書籍:

- TURBO C 2.0 使用手冊
- TURBO C++ 程式設計師手冊
- TURBO C++ 入門指引
- TURBO C繪圖之技巧
- 認識 C語言

M • E • M • O



Daily BONANZA To YOU

目 錄

%%%

第 1 章 導論

1.0 何謂資料結構.....	1-2
1.1 C 與資料結構.....	1-3
1.2 演算法與演算法分析.....	1-5
1.3 Big-Oh 函數.....	1-10
1.4 書中程式使用說明.....	1-13
重點整理.....	1-14
第一章習題.....	1-15
第一章習題解答.....	1-20

第 2 章 c 之陣列與結構

2.1 陣列.....	2-2
2.2 陣列之應用.....	2-7
2.2.1 兩矩陣相加.....	2-7
2.2.2 轉置矩陣.....	2-13
2.3 特殊矩陣表示法.....	2-16
2.3.1 稀疏矩陣.....	2-16
2.3.2 上三角形與下三角形表示法.....	2-22
2.3.3 對稱矩陣.....	2-26
2.4 陣列之應用—多項式相加.....	2-26
2.5 C 之結構與組合.....	2-36
2.5.1 結構.....	2-37
2.5.1.1 typedef.....	2-40
2.5.1.2 結構指標.....	2-40
2.5.2 組合.....	2-42
2.5.3 結論.....	2-45
2.6 專題討論 -- 製作處理集合功能之函式.....	2-45

重點整理.....	2-54
第二章習題.....	2-56
第二章習題解答.....	2-58

第 3 章 堆疊與佇列

3.1 堆疊之定義與操作.....	3-2
3.2 使用陣列與結構來製作堆疊.....	3-4
3.2.1 另一種方式的堆疊運算.....	3-10
3.2.2 多個堆疊.....	3-13
3.3 堆疊之應用.....	3-17
3.3.1 前序、中序及後序表示法.....	3-18
3.3.1.1 將中序式轉成後序式之演算法.....	3-22
3.3.1.2 將中序式轉成後序式之程式.....	3-25
3.3.1.3 由後序式求值.....	3-32
3.3.2 八個皇后問題 (Eight Queen Problem).....	3-37
3.3.3 迷宮問題 (Maze Problem).....	3-51
(1) 設定與儲存一個迷宮:.....	3-51
(2) 尋找之方向:.....	3-52
(3) 找尋路徑:.....	3-52
3.4 佇列.....	3-62
3.5 使用陣列與結構來製作佇列.....	3-64
3.5.1 單純之佇列操作.....	3-64
3.5.2 環狀佇列之操作.....	3-66
3.5.3 用雙佇列來解決特權問題.....	3-72
3.6 訂位系統.....	3-74
3.7 結語.....	3-86
重點整理.....	3-86
第三章習題.....	3-88

第 4 章 鏈結串列

4.1 指標與動態記憶體分配.....	4-2
---------------------	-----

4.1.1	指標與位址.....	4-2
4.1.2	指標與陣列、字串及指標的運算.....	4-5
4.1.3	動態記憶體分配.....	4-9
	使用 malloc.....	4-10
	使用 calloc.....	4-11
	強制改型 (cast).....	4-12
	NULL 指標.....	4-13
	歸還 (free) 記憶體.....	4-14
4.1.4	指標的其它論題.....	4-15
	指標陣列.....	4-15
	型態不一致.....	4-16
4.2	鏈結串列之定義與製作.....	4-17
4.2.1	使用陣列來製作鏈結串列.....	4-18
4.2.3	使用指標來製作鏈結串列.....	4-20
4.3	鏈結串列之操作.....	4-23
4.3.1	建立一新串列.....	4-26
4.3.2	加入一新節點.....	4-27
	4.3.2.1 將節點加入串列之前面.....	4-27
	4.3.2.2 加入一節點到串列的尾端.....	4-31
	4.3.2.3 在串列中插入一節點.....	4-32
4.3.3	刪除一節點.....	4-35
	4.3.3.1 刪除串列的第一個節點.....	4-35
	4.3.3.2 刪除串列最後一個節點.....	4-37
	4.3.3.3 刪除特定的節點.....	4-39
4.4	使用鏈結串列來製作堆疊與佇列.....	4-41
4.4.1	製作堆疊.....	4-41
4.4.2	製作佇列.....	4-44
4.5	串列的搜尋與串列的其它運作.....	4-44
4.5.1	串列的搜尋.....	4-44
4.5.2	串列的其它操作.....	4-49
	4.5.2.1 將兩串列合併.....	4-49
	4.5.2.2 清除一串列.....	4-51
	4.5.2.3 將串列反轉 (invert).....	4-52

4.6	多項式相加.....	4-53
4.7	環狀鏈結串列(Circular Linked List).....	4-67
4.8	環狀串列之操作.....	4-69
4.8.1	建立一個環狀串列.....	4-69
4.8.2	在環狀串列中加入一節點.....	4-70
4.8.3	環狀串列之追蹤.....	4-72
4.8.4	刪除環狀串列中的節點.....	4-74
4.8.5	環狀串列之回收.....	4-78
4.9	約瑟夫 (Josephus) 問題.....	4-79
4.10	雙向鏈結串列(Doubly Linked List).....	4-91
4.11	雙向鏈節串列之操作.....	4-92
4.11.1	加入節點.....	4-92
4.11.2	刪除節點.....	4-93
4.12	串串列 (Lists of Lists).....	4-94
	重點整理.....	4-97
	第四章 習題.....	4-98

第 5 章 函式之遞迴與選擇

5.1	遞迴 (Recursion).....	5-2
5.2	迭代 (Iteration).....	5-10
5.3	遞迴與迭代之選擇.....	5-11
5.4	遞迴之實際應用.....	5-12
5.4.1	兩個有關數學運算之函式.....	5-12
5.4.2	河內塔 (Tower of Hanoi).....	5-16
5.4.3	八個皇后問題.....	5-29
5.4.4	騎士問題.....	5-37
5.4.5	遞迴的二分搜尋 (Binary Search).....	5-44
5.5	函式指標 (Function Pointer)	
	與函式的選擇 (Function Selection).....	5-46
	重點整理.....	5-49
	第五章 習題.....	5-49

第 6 章 樹狀結構 (Tree Structure)

6.1	樹狀結構之觀念與定義	6-2
6.2	樹之表示法與儲存	6-7
6.3	二元樹	6-10
6.4	二元樹的儲存	6-15
6.4.1	用一維陣列來儲存二元樹	6-15
6.4.2	用鏈結方式來儲存二元樹	6-17
6.5	二元樹之運作與追蹤	6-18
6.5.1	在二元樹中加入一節點	6-19
6.5.2	完全平衡樹的建立	6-24
6.5.3	二元搜尋樹節點之刪除	6-26
6.5.4	二元樹的追蹤 (traversal)	6-33
6.6	二元樹之應用	6-39
6.6.1	二元樹之完整操作	6-39
6.6.2	二元樹排序 (sort)	6-57
6.6.3	二元算式樹(Binary Expression Tree)	6-59
6.7	二元樹的其它運作	6-76
6.7.1	決定唯一的二元樹	6-76
6.7.2	如何將一般樹與森林化成二元樹	6-78
6.7.3	複製一棵二元樹	6-80
6.7.4	比較二棵二元樹是否相等	6-81
6.7.5	引線二元樹	6-83
6.8	其餘有關樹的論題	6-96
6.8.1	有關 m 元樹 (m -ary tree) 的一些問題	6-96
6.8.2	樹的其它應用	6-98
6.8.2.1	決策樹	6-98
6.8.2.2	Game Tree	6-101
	重點整理	6-104
	第六章 習題	6-107

附錄A C之關鍵字 (keyword)

A.1	標準 C 的關鍵字.....	A-1
A.2	Microsoft C 之關鍵字.....	A-2
A.3	Turbo C 之關鍵字.....	A-3

附錄B 運算子之優先權與運算順序

附錄C C語法總結 (syntax summary)

C.1	運算式.....	C-1
C.2	宣告 (Declaration).....	C-4
C.3	敘述.....	C-7
C.4	外部定義(External definitions).....	C-9
C.5	前處理器(Preprocessor).....	C-10

附錄D 建議參考書籍

D.1	語言程式設計.....	D-1
D.2	有關資料結構方面之書籍.....	D-1
D.3	編譯器.....	D-2

附錄E 上册總結 — 程式實作

堆疊的應用——算式編譯器.....	E-25
(Expression Compiler).....	E-25
鏈結串列的應用——使用單向鏈結串列	
來製作一行編輯器 (Line Editor).....	E-37
練習題.....	E-66

第1章 導論

本章目標

- 能對一問題使用虛擬碼進行由上而下的設計。
- 區別程式和演算法之差異。
- 使用數學函數 Big-Oh, 評估演算法之效率。

1.0 何謂資料結構

BONANZA 榮園 (R)

1. 在應用方面將探討小量資料如何利用主記憶體來進行排序(sorting), 以及大量資料如何利用輔助記憶體來排序的各種方法。

2. 將介紹如何在主記憶體內搜尋小量資料, 以及如何在輔助記憶體內搜尋大量資料的各種方法。

資料結構是探討電腦內部各種資料儲存方式, 並對於資料的運用、演算和處理提供一個評估之方法, 它是電腦科學中最基本之課題之一。本書將介紹一些基本之資料結構觀念, 譬如: 陣列 (array), 串列 (linked list), 堆疊 (stack), 佇列 (queue), 以及樹 (tree)。在應用方面將探討小量資料在主記憶體內排序 (sorting) 之各種方法, 以及大量資料在輔助記憶體內排序之各種方法。同時在另一個搜尋 (searching) 的領域裡, 將介紹小量資料在主記憶體內之各種搜尋方法, 以及大量資料在次記憶體之各種搜尋方法。最後, 將介紹檔案中之索引檔 (indexed sequential file) 以及隨機檔 (random file) 之設計原理及技巧。

以上內容將提供讀者解決計算機問題之一些工具, 並就 CPU 時間與記憶體空間兩因素加以評估工具的效率, 以期使讀者在設計資訊系統時能重視資源之效率問題, 並提高軟體之品質。

1.1 C 與資料結構

BONANZA 集團 (R)

以前，處理資料結構的問題，均是使用 Pascal 語言。但是隨著 UNIX 系統與 C 語言的流行，再加上 C 本身許多的優點：簡單，容易，功能強大，更重要的是標準性（各種 C 編譯器均是以 Brian M. Kernighan 和 Dennis M. Ritchie 合著之 The C Programming Language 一書做標準），使 C 語言逐漸取代 Pascal 語言。

C 提供了兩項特點，使我們在處理資料結構的問題時更加容易，那就是：指標 (Pointer) 與結構 (Structure)。

指標使我們在處理動態的資料結構時更加方便。C 的指標還有一項優點，那就是限制很少，使我們能做的事更多。以交換一陣列中之資料為例，如下之程式片斷：

```
sort(data,...)
int *data;
.
.
{
    int temp, i;
    .
    .
    temp = *data;
    *data = *(data+i);
    *(data+i) = temp;
    .
    .
}
```

要比使用如下之程式片斷：

```
sort(data,...)
int data[];
.
.
{
```