

結構化程式語言 PASCAL

吳建平 合著
許舜欽



松崗電腦圖書資料有限公司

結構化程式語言 PASCAL

吳建平 合著
許舜欽

松崗電腦圖書資料有限公司 印行

松崗電腦圖書資料有限公司已
聘任本律師為常年法律顧問，
如有侵害其著作權或其他權益
者，本律師當依法保障之。

長立國際法律事務所

陳 長 律師



結構化程式語言PASCAL

編著者：吳建平・許舜欽

發行人：朱 小 珍

發行所：松崗電腦圖書資料有限公司

台北市敦化南路五九三號五樓

電 話：(02) 7082125(代表號)

郵政劃撥：0109030-8

印刷者：建發印刷設計公司

中華民國七十五年九月初版

中華民國七十六年九月第三版

版權
所有



翻
印
必
究

每本定價 200 元整

書號：2101230

本出版社經行政院新聞局核准登記，登記號碼為局版台業字第三一九六號

序 言

科技進步，一日千里。電子計算機（或稱電腦）已漸成為人類解決問題不可或缺的工具，而 Pascal 程式語言更是資訊科學與工程應用方面必備的基本知識。

本書主要根據 1982 年 ISO 標準 Pascal 規定，同時介紹 IBM-PC 和 16 位元微電腦上執行的 TURBO 系統。全書共分十二章，第一、二章討論計算機及 Pascal 程式語言之基本概念。第三、四、五章介紹簡單資料型態及輸入與輸出。第六、七章解說 Pascal 優雅的控制流程。第八、九、十、十一章介紹結構化的資料型態。最後在第十二章討論動態資料結構所用的指標型態。

本書除了詳細解說 Pascal 各種敘述之外，並從程式語言結構的觀點分析其利弊得失。同時更精選例題，示範良好的程式設計技巧。而在各章之後並附習題以供讀者溫故知新，舉一反三之用。

筆者利用授課之餘參考有關著作，編著此書。雖然竭盡心力，以求完備，然因付梓倉促，漏誤之處在所難免，尚祈讀者及諸先進不吝指正。

最後，我們感謝台大資訊系助教黃美玉小姐不辭辛勞，鍵入並測試本書各章節的所有程式，系友莊庭瑞先生幫忙準備附錄，以及台北商專許麗珠副教授所給予的寶貴意見。

吳建平、許舜欽

民國七十五年八月於

台灣大學資訊系、電機系

目 錄

序 言

第一章 導 論	1
1-1 電子計算機簡介.....	2
1-1-1 計算機的特性.....	2
1-1-2 計算機的結構.....	3
1-2 計算機語言.....	5
1-2-1 機器語言與組合語言.....	5
1-2-2 高階語言.....	6
1-3 PASCAL程式語言	8
1-3-1 PASCAL簡史	8
1-3-2 PASCAL簡介	9
1-4 流程圖.....	11
1-5 使用計算機解題之過程.....	12
1-5-1 準備原始程式.....	12
1-5-2 編譯與執行.....	12
習題一.....	15

第二章 PASCAL 語言特性 17

2-1	自由格式.....	18
2-2	語法圖.....	18
2-3	字元組.....	20
2-4	保留字.....	21
2-5	識別號.....	22
2-6	變數宣告.....	23
2-7	常數名稱.....	25
2-8	註解敘述.....	27
2-9	字元串.....	28
	習題二.....	30

第三章 標準資料型態 31

3-1	資料型態.....	32
3-2	整數型態.....	33
3-3	實數型態.....	37
3-4	字元型態.....	44
3-5	布林型態.....	46
3-6	序數型態.....	50
	習題三.....	52

第四章 使用者定義的資料型態 53

4-1	列舉資料型態.....	54
4-2	子區間型態.....	57
4-3	簡單資料型態的分類.....	60

習題四.....	61
第五章 基本輸入與輸出	63
5-1 輸入敘述.....	64
習題五.....	73
第六章 基本控制敘述	75
6-1 複合敘述.....	76
6-2 反覆敘述.....	77
6-2-1 repeat 敘述.....	78
6-2-2 while 敘述.....	82
6-2-3 for 敘述.....	87
6-3 條件敘述.....	92
6-3-1 if 敘述	92
6-3-2 case 敘述	105
6-4 標號與 goto 敘述	108
習題六.....	111
第七章 程序與函數.....	113
7-1 程序.....	114
7-2 數值參數.....	120
7-3 變數參數.....	124
7-4 函數.....	128
7-5 區段結構.....	132
7-6 附帶作用.....	134
7-7 遞迴副程式.....	136

7-8 前置程序	143
習題七	145
第八章 陣列資料型態	149
8-1 結構化資料型態	150
8-2 陣列型態	150
8-3 二維陣列	155
8-4 壓縮陣列	161
8-5 字串型態	162
8-6 應用程式	164
8-6-1 選擇整序法	164
8-6-2 線性搜尋	166
8-6-3 二元搜尋	167
8-6-4 數字排列	169
8-6-5 快速整序法	170
8-6-6 判定質數	172
習題八	173
第九章 記錄資料型態	175
9-1 記錄型態	176
9-2 with 敘述	178
9-3 變異部份	181
習題九	184
第十章 檔案資料型態	185
10-1 檔案型態	186

10-2 檔案的使用.....	187
10-3 文書檔.....	194
習題十.....	196
第十一章 集合資料型態	197
11-1 集合型態.....	198
11-2 集合運算.....	199
習題十一.....	204
第十二章 指標資料型態	205
12-1 指標宣告.....	206
12-2 鏈結串列.....	208
12-3 二元樹	212
習題十二	215
附錄A 簡介TURBO Pascal的程式設計環境	217
附錄B TURBO命令摘要	225
附錄C TURBO的編輯程式的命令	229
附錄D TURBO Pascal中新增加的預先定義 的常式和函數	235
附錄E TURBO Pascal編譯程式中常見的 選擇項摘要	239

附錄 F	TURBO Pascal裏使用調入檔案的情形	245
附錄 G	COM檔案——編譯TURBO Pascal的 程式以便日後使用	247
附錄 H	Pascal語圖法	251

第一章

導 論

2 程式語言 Pascal

人類使用工具，推動世界文明的進展。十八世紀，蒸汽機的發明改變人類使用肌肉的方式，促成第一次的產業革命。二十世紀電子計算機（computer）的發明，又延伸人類使用腦筋的範疇，帶來了第二次的產業革命。

隨著科技文明的發展，計算機（俗稱電腦）的應用範圍日益廣泛，如今已經達到無可限界的程度。展望未來，人類生活與計算機將更密切相關。因此，如何使用計算機，實在值得吾人研究。

1-1 電子計算機簡介

1-1-1 計算機的特性

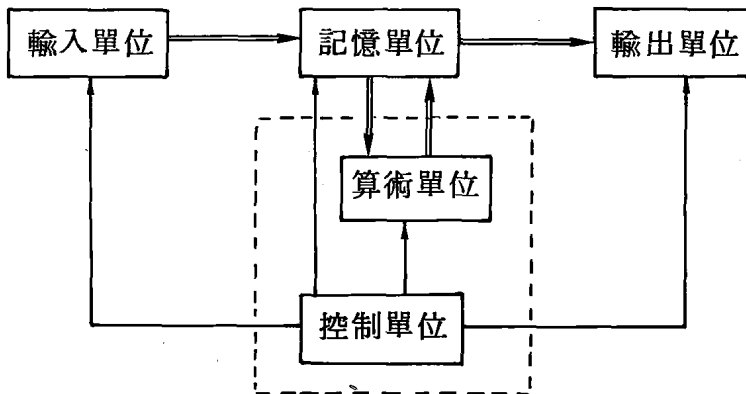
我們使用計算機，主要借重它的下列幾個特性：

- (1)計算速度快：計算機以極高的速度運算資料。第一代的真空管計算機，速度以千分之一秒（milli-second）為單位。第二代的電晶體計算機，速度以百萬分之一秒（micro-second）為單位。第三代的積體電路計算機，則以十億分之一秒（nano-second）為單位。對於爭取時效的計算工作，如飛彈防空系統，就要利用這種特性。
- (2)記憶容量大：計算機除了主要記憶器之外，還有輔助記憶裝置，我們可以利用這些記憶器，存放大量的資料，形成資料庫（data-base），以供隨時檢索（retrieve）。
- (3)高度正確性：計算機不容易發生偶發性的錯誤，這種高度的正確性，使得人們樂於利用它來整理帳目，控制儀器，以及解決各種問題。

(4)不怕煩，不怕累：只要程式需求，再煩再久的計算，它都能夠照令執行，毫無怨尤。這種刻苦耐勞的精神，使得人們能夠利用它來克服許多繁瑣複雜的難題。

1-1-2 計算機的結構

我們要使用計算機，首先要了解它的結構。一般的計算機係由記憶、算術、控制、以及輸入、輸出等五個單位所組成，它們之間的關係大體上如圖 1-1 所示。



其中 $\Rightarrow$ 表示資訊流向 $\rightarrow$ 表示控制信號

圖 1-1 計算機結構方塊圖

以下分別略述圖 1-1 中各單位常用的各種裝置 (device)

(1)輸入裝置

常見的輸入裝置如下：

- * 讀紙帶機 (paper-tape reader)
- * 讀卡機 (card reader)

4 程式語言 Pascal

- * 電傳打字機 (tele-typewriter, 簡稱 TTY)
- * 光學讀字機 (optical character reader, 簡稱 OCR)
- * 磁性墨水讀字機 (magnetic ink character reader , 簡稱 MICR)
- * 磁帶機 (magnetic tape drive)
- * 磁碟機 (magnetic disc drive)

(2)輸出裝置

常見的輸出裝置如下：

- * 打紙帶機 (paper - tape punch)
- * 打卡機 (card punch)
- * 電傳打字機 (TTY)
- * 列印機 (line printer)
- * 繪圖機 (plotter)
- * 陰極射線管顯示器 (cathod-ray-tube display, 簡稱 CRT)

(3)記憶裝置

記憶裝置大致可分爲主記憶器 (main memory) 和輔助記憶器 (auxiliary memory) 兩類。

主記憶器通常採用磁蕊 (magnetic core) 或積體電路 (integrated circuit) 製成。它們傳送資料的速度較快，但較昂貴，因此容量有限。

輔助記憶器的速度雖然較慢，但容量較大，可以用來儲存大量資料。常見的輔助記憶器有磁帶 (magnetic tape)，磁碟 (magnetic disc)，磁鼓 (magnetic drum) 等裝置。

(4)算術單位和控制單位

算術單位通常使用極快的暫存器 (register) 和邏輯電路來實施各種運算。

控制單位也是經由複雜的電路，產生各種信號，來控制計算機的運轉。

這兩個單位經常合稱為中央處理單位 (central processing unit 簡稱 CPU)。

1-2 計算機語言

1-2-1 機器語言與組合語言

電子計算機和手邊常用的計算器 (calculator) 最主要的不同點乃是：一般計算器必須由人手逐步操作運算，而計算機則是把操作的指令事先做成程式 (program) 存放在主記憶器裏，再由控制單位循序擷取指令加以執行。

控制單位所執行的指令是由一連串的 0 與 1 構成的，叫做機器語言 (machine language)。1940 年代電子計算機剛出現時，就是使用這種語言。

由於機器語言不易記憶，程式製作也不方便，於是有組合語言 (assembly language) 出現。組合語言係以助記符號 (mnemonic symbol) 來代表指令和記憶位址 (address)，使人們較易撰寫程式。但是機器並不懂這些符號，所以必須由組合程式 (assembler) 將之轉譯為機器語言才能執行。

組合語言雖然使用符號，較為方便，但是程式本身與機器語言仍

是非常相似，效率雖高，却嫌繁雜。一般使用者只是爲了解決特殊問題，並沒有了解機器內部構造的必要，因此高階語言（ high level' language ）應運而生。

1-2-2 高階語言

高階語言是爲各種應用而設計的語言，它與計算機的特性無關，因此可使程式設計人員減少許多不必要的細節考慮，而能夠專注於處理問題的邏輯設計。

高階語言和組合語言一樣，必須經過編譯程式（ compiler ）轉譯成機器語言才能執行。轉譯之前的程式叫做原始程式（ source program），而轉譯之後的程式叫做目的程式（ object program），這個過程如圖 1-2 所示：

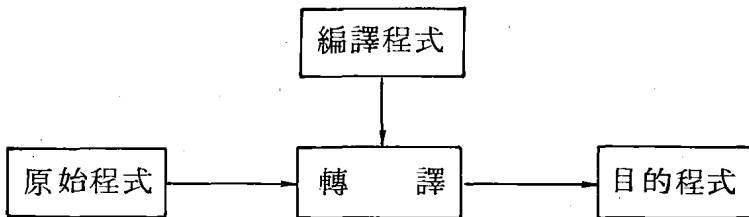


圖 1-2 高階語言之編譯

常見的高階語言有以下幾種：

(1) FORTRAN (FORmula TRANslation language)

FORTRAN 是針對科學及工程計算而設計的語言，由於能夠簡易表達計算的過程，廣被採用，是流傳最廣的高階語言。

(2) COBOL (COmmon Business Oriented Language)

COBOL 是爲商業資料處理而設計的語言，段落分明，近似英文，是商業上資料處理最常用的語言。

(3) ALGOL (ALGO^rithmic Language)

ALGOL 是使用正規文法 (Formal grammar) 定義的程式語言，結構優良，是廣受學術界重視，而影響深遠的科學用途程式語言。

(4) PL/1 (Programming Language / 1)

PL/1 是 IBM 公司綜合上列三種語言的特點而發展出的多功能 (Multi-purpose) 程式語言，由於用途廣泛，內容較為複雜，但由於 IBM 公司的強力支持，也是一個相當重要的程式語言。

(5) BASIC (Beginner's All-purposed Symbolic Instruction Code)

BASIC 是將 FORTRAN 簡化之語言，易學易用，功能亦足，是目前微電腦的主要語言。

(6) Pascal

Pascal 繼承 ALGOL 之架構，並在資料結構方面加以改進，語法清晰、簡潔，是資訊教育和一般應用程式設計的最佳語言。

(7) C

C 是 UNIX 作業系統的主要語言，它具有高階語言的結構，並有組合語言的效率，雖然較不可靠，但因限制少，功能強，為熟練的系統設計師所愛用。

(8) Ada

Ada 是美國國防部支持而發展出來的，它綜合所有語言的特點，適合大型軟體程式之製作。

(9) LISP (List Processor)

LISP 屬於非數值程式語言 (Nonnumerical program-