

電子計算機科學叢書

郭德盛 主編

程式語言 FORTRAN 77

第四版

郭德盛・許舜欽 合著

田野出版社 印行

田野

電子計算機科學叢書

程式語言 FORTRAN 77

郭德盛 許俊樹 合著

田野出版社 印行

程式語言FORTRAN 77

翻
印
必
究



版
權
所
有

每本定價新台幣 160 元整

書號：210205

編著者：郭 德 盛・許 舜 欽

發行人：陳 碧 玉

發行所：田野出版社

台北市仁愛路二段——〇號三樓

電話：3930255・3930249

總經銷：松崗電腦圖書資料有限公司

台北市仁愛路二段——〇號三樓

電話：3930255・3930249

郵政劃撥：109030

經 銷：文 笙 書 局

台北市重慶南路一段六十九號

承印者：泉崗印刷股份有限公司

台北市仁愛路二段——〇號三樓

電話：3930255・3930249

中華民國七十年一月 初 版

中華民國七十一年十月 第二版

中華民國七十二年二月 第三版

中華民國七十二年九月 第四版

中華民國七十三年三月 第五版

本出版社經行政院新聞局核准登記，
登記證為局版台業字第〇二〇七號。

序 言

科技進步，一日千里。計算機已漸成爲人類解決問題不可或缺的工具，而福傳（FORTRAN）程式語言更是科學及工程應用方面必備的基本知識。

結構程式設計（Structured Programming）技巧已被公認爲軟體工程必備之程式設計技巧。FORTRAN 77爲1977年美國國家標準局制定之標準福傳程式語言，特別加強配合結構程式設計之特性。

本書旨在介紹繼FORTRAN IV之後成爲新標準ANSI FORTRAN藍本的FORTRAN 77。全書共分十章，第一、二章討論計算機及FORTRAN程式語言之基本概念，第三章至第八章配合精選例題，詳細解說FORTRAN各種敘述的用法，九、十兩章更深入討論其他FORTRAN敘述。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，然漏誤之處在所難免，尚祈讀者及諸先進不吝指正。

郭德盛，許舜欽
民國七十年六月於
臺灣大學電機系

增訂版序

本書自七十年六月問世以來，承蒙各方先進與讀者之指正和鼓勵，毋雙感荷。分析並綜合各方之指教，乃決定增訂本書，以求本書更加完善，並希望不負各方之期待。

此次除將本書各章之習題加以充實之外，並增訂兩章。在第十一章，以實例來展示福傳程式之各種應用，並隨時顯示結構化程式之精神，尤其在「薪資計算」之例題裡，特別提示結構化程式設計之標準步驟，以期所撰寫出來之程式較為可靠，且易於掌握。本章最後之實例，設計一「銷售分析」之程式介紹程式中加入「詳細註解」之觀念，增加程式之可讀性，此乃對程式維護上幫助最大的方法。

在熟悉福傳程式語言及其應用之後，為使讀者在程式設計能力上，百尺竿頭，更進一步。本書在第十二章特別列舉要寫好程式之要點，使讀者能寫出具有良好風格之程式，使所寫之程式清晰、明確，易於掌握，易於維護。

為使本書更加完備，尚祈讀者及諸位先進不吝指正。

郭德盛・許舜欽

民國七十二年二月 於

台灣大學電機系、資訊系

目 錄

第一章 導論	1
1-1 電子計算機簡介	1
1-1-1 計算機的特性	1
1-1-2 計算機的結構	2
1-2 計算機語言	4
1-2-1 機器語言與組合語言	4
1-2-2 高階語言	4
1-3 FORTRAN 程式語言	6
1-3-1 FORTRAN 簡史	6
1-3-2 FORTRAN 簡介	6
1-4 流程圖	8
1-5 使用計算機解題程序	10
1-5-1 準備原始程序	10
1-5-2 編譯與執行	12
1-5-3 工作控制語言	14
第二章 FORTRAN 語言特質	17
2-1 字元組	17
2-2 敘述結構	18
2-3 資料型式	19

2-3-1 整數	19
2-3-2 實數	20
2-3-3 邏輯及字元型式	21
2-4 常數	22
2-4-1 整常數	22
2-4-2 實常數	22
2-4-3 邏輯常數	23
2-4-4 字元及郝氏常數	23
2-4-5 常數名稱	24
2-5 變數	24
2-5-1 變數名稱	24
2-5-2 變數型式	25
2-5-3 變數值	25
2-6 資料結構	26
2-6-1 資料錄和資料檔	26
2-6-2 陣列	26
2-6-3 字元子串	27
2-7 程式結構	27

第三章 陳式與指定敘述	31
3-1 算術陳式	31
3-1-1 算術運算子	31
3-1-2 算術陳式之計值	31
3-1-3 算術陳式值的資料型式	33
3-2 算術指定敘述	34
3-3 字元陳式與字元指定敘述	37

3-3-1 字元陳式	37
3-3-2 字元指定敘述	37
3-4 邏輯陳式與邏輯指定敘述	38
3-4-1 關係陳式	38
3-4-2 邏輯陳式	39
3-4-3 邏輯指定敘述	41
第四章 基本輸入與輸出	45
4-1 輸入與輸出敘述的形式	45
4-1-1 控制資訊	45
4-1-2 輸入輸出名列	46
4-2 序列導向輸入或輸出	47
4-3 FORMAT 敘述及其用法	48
4-3-1 機架控制	49
4-3-2 I 格式碼	49
4-3-3 F 格式碼	51
4-3-4 X 格式碼	53
4-3-5 H 格式碼	54
4-3-6 引號格式	55
4-3-7 A 格式碼	55
4-3-8 斜線	56
4-3-9 輸入或輸出名列與格式碼	57
第五章 基本控制敘述	63
5-1 GO TO 敘述	63
5-1-1 無條件GO TO敘述	63

5-1-2 計值 GO TO	66
5-1-3 指定 GO TO 敘述與 ASSIGN 敘述	69
5-2 IF 敘述	70
5-2-1 算術 IF 敘述	70
5-2-2 邏輯 IF 敘述	72
5-2-3 版段 IF 敘述	76
5-2-4 ELSE IF 敘述	82
5-2-5 ENDIF 敘述	87
5-3 程式的結束或暫停	89
5-3-1 STOP 敘述	89
5-3-2 PAUSE 敘述	89
5-3-3 END 敘述	90
5-3-4 CONTINUE 敘述	90

第六章 DO 敘述 101

6-1 迴圈與 DO 敘述	101
6-1-1 DO 迴圈的形成	102
6-1-2 DO 迴圈的結構及執行	103
6-1-3 DO 迴圈應注意事項	106
6-2 內隱 DO 迴圈	115
6-2-1 內隱 DO 迴圈的形式	115
6-2-2 迴圈與資料及格式碼的配合	119

第七章 函數與副程式 125

7-1 函數	125
7-1-1 內在函數	125
7-1-2 敘述函數	129

7-2 副程式.....	131
7-2-1 函數副程式.....	131
7-2-2 次常式副程式.....	135
7-3 虛擬引數和真實引數.....	138
第八章 宣告敘述.....	145
8-1 EQUIVALENCE 敘述	145
8-2 COMMON 敘述	147
8-3 DATA 敘述.....	151
8-4 EXTERNAL 敘述	152
8-5 INTRINSIC 敘述	153
8-6 SAVE 敘述.....	154
第九章 其他輸入與輸出敘述.....	159
9-1 錄和檔的處理.....	159
9-1-1 循序出入與直接出入.....	159
9-1-2 OPEN 敘述	160
9-1-3 CLOSE 敘述	162
9-1-4 INQUIRE 敘述	162
9-1-5 定位敘述.....	163
9-2 其他格式碼.....	165
9-2-1 E 格式碼和D 格式碼.....	165
9-2-2 G 格式碼.....	167
9-2-3 P 格式碼.....	168
9-2-4 L 格式碼.....	169
9-2-5 T, TR, TL 格式碼.....	169
9-2-6 S, SP, SS 格式碼.....	170

9-2-7 BN, BZ 格式碼	170
第十章 其他 FORTRAN 敘述	173
10-1 複數運算	173
10-2 倍準運算	175
10-3 BLOCK DATA 程式單元	176
10-4 ENTRY 敘述	177
10-5 RETURN 敘述	179
第十一章 應用程式	185
11-1 計算面積	185
11-2 插入排序法	188
11-3 選擇排序法	190
11-4 正弦及餘弦曲線圖	192
11-5 薪資計算	194
11-6 銷售分析	199
11-7 中文字形輸出	204
11-8 疊表的操作	211
第十二章 如何寫好程式	213
12-1 一般要點	213
12-2 程式結構要點	220
12-3 輸入與輸出敘述要點	226
習題解答	233
附錄 A	237
附錄 B	241
附錄 C	243

附錄 D	265
程式索引	267
案 引	269

第一章 導 論

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**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。

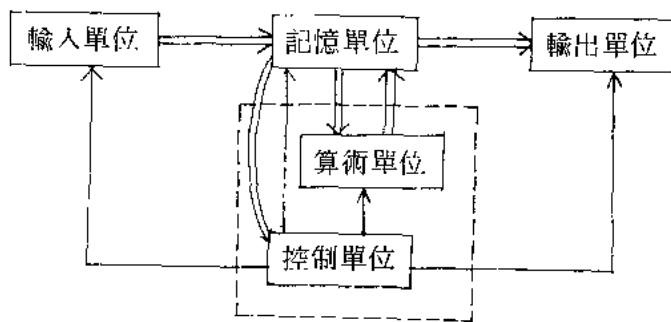
(4)不怕繁，不怕累：只要程式需求，再繁再久的計算，它都能夠照

2 程式語言 FORTRAN 77

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)
- * 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

記憶裝置大致可分為**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。最近更有**磁泡** (Magnetic bubble) 裝置上市。

(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。一般使用者只是為了解決特殊問題，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

高階語言是為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：

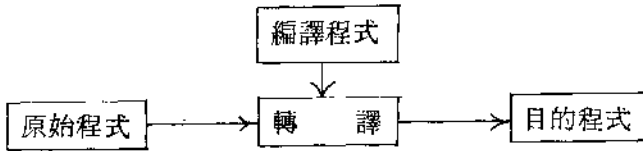


圖 1-2 高階語言之編譯

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 是為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 (ALGOrithmic 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) PASCAL *

PASCAL 繼承 ALGOL，並在資料結構方面加以改進，是極有前途的程式語言。

此外，還有以處理串列為主的 LISP (LIST Processing language)；以處理文字為主的 SNOBOL，和終端機上所用的 APL (A Programming Language) 以及簡單易學的 BASIC (Beginner All-purpose Symbolic Instruction Code) 等都是常見的高階語言。