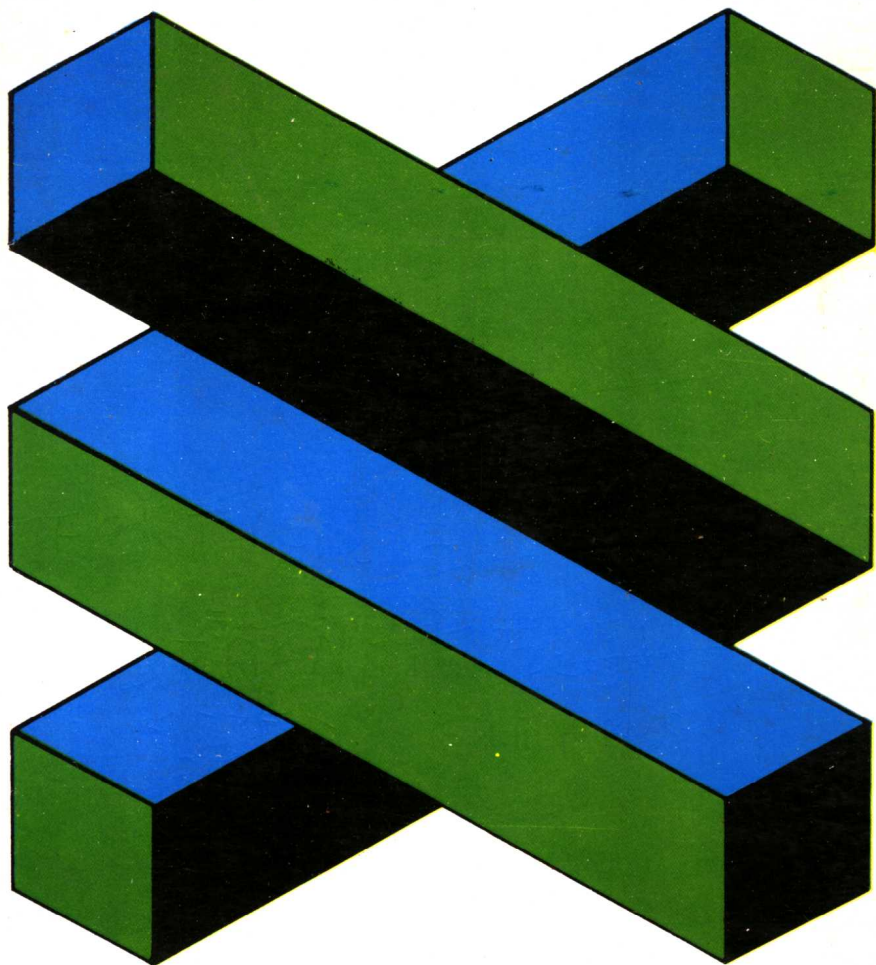


FORTRAN程式範例

李文龍 ■ 譯編



協群科技出版社

FORTRAN

程式範例

編譯・李文龍

協群科技出版社

FORTRAN程式範例

編譯者：李 文 龍

出版：協群科技出版社

發行：協群科技出版社

香港中環卑利街684號二樓

印刷者：廣 源 印 務 局

青山道875號工廠大廈

定價：H.K.\$

前言

本書是以「簡易程式計劃 3」（福傳的文法與實際）之續篇而整理的。亦即，編輯本書的目的在於使某程度具有電腦語言之知識（未必僅限於福傳）的人士，藉更多數的比較具體的問題，能充分地有效運用福傳。

其內容由

第 1 章 文法

第 2 章 運用

第 3 章 程式範例

所編成。

第 1 章網羅了一切較具代表性的福傳的文法，在編排上務求能有效地善用它來作為第 2 章、第 3 章的開導。因此凡是俱備某程度福傳知識的人士，即使不唸該章而從第 2 章開始唸起，照理說也能充分了解才對。如果唸了第 2 章以後，在文法上感覺有所不妥時，請再確實地查證目錄或索引的該部分。

即使不俱備福傳的知識，只要有其它電腦語言的知識，也能有充分的能力來閱讀，同時也能有效運用此書。在這情況下，請您從第 1 章開始仔細地讀。

第 2 章是列舉二次方程式，聯立一次方程式的問題，一面設計程式，一面更進一步地說明文法。只要充分了解第 2 章，那麼就能著手撰寫福傳程式。

第 3 章列舉了十個例題，先說明其定義再詳述計算方法、計算步

2 FORTRAN程式範例

驟，然後設計程式。

此例題含有數值分析、統計、資料處理及遊戲等，爲了要把各項獨立起見，即使從最感到興趣的問題開始來研究，照理也能充分地了解。

如果要撰寫實際上機器使用的福傳程式時，因各電腦系統之不同，在文法上也就有或多或少的不同。因此我們必須使用由該計算機指定的文法書。

學習福傳時，希望不要只注意基本原理而已，同時也必須學會其使用法。因此在第2章、第3章，也許有依存於特定計算機的要素也說不定，但是筆者認爲已經儘可能使用具體的例子來說明，而希望各位讀者學會其運用法。

各位若由於本書，而能稱心如意的撰寫福傳程式，則爲筆者之幸。

筆者謹識

FORTRAN程式範例目錄

第1章 FORTRAN的文法

1 · 1	FORTRAN 程式	9
1 · 2	常數、變數、註標變數	10
1.2.1	整常數	10
1.2.2	實常數	10
1.2.3	複常數	11
1.2.4	邏輯常數	11
1.2.5	整變數	11
1.2.6	實變數	11
1.2.7	IMPLICIT 敘述的一般形態	11
1.2.8	DIMENSION 敘述的一般形態	12
1.2.9	註標變數的一般形態	13
1.2.10	類型敘述的一般形態	14
1 · 3	演算式	15
1.3.1	演算敘述	15
1.3.2	邏輯運算式	16
1 · 4	輸出入	17
1.4.1	READ 敘述的一般形態	17
1.4.2	WRITE 敘述的一般形態	18
1.4.3	註標變數的輸出入	19
1.4.4	FORMAT 敘述的一般形態	19
1.4.5	REWIND 的一般形態	27
1.4.6	BACK SPACE 敘述的一般形態	28

4 FORTRAN 程式範例

1.4.7	END FILE 敘述的一般形態	28
1.4.8	NAMelist 敘述的一般形態	29
1.4.9	直接存取的輸出入	30
1.5	執行順序	31
1.5.1	GO TO 敘述的一般形態	32
1.5.2	有條件GO TO 敘述的一般形態	32
1.5.3	指派GO TO 敘述的一般形態	33
1.5.4	算術IF 敘述的一般形態	33
1.5.5	邏輯IF 敘述的一般形態	34
1.6	重覆	34
1.6.1	DO 敘述的一般形態	34
1.6.2	CONTINUE 敘述的一般形態	35
1.7	終止的敘述	38
1.7.1	PAUSE 敘述的一般形態	38
1.7.2	STOP 敘述的一般形態	38
1.7.3	END 敘述的一般形態	39
1.8	副程式	39
1.8.1	敘述函數的一般形態	39
1.8.2	FUNCTION 副程式的一般形態	40
1.8.3	SUBROUTINE 副程式的一般形態	41
1.8.4	CALL 敘述的一般形態	41
1.8.5	RETURN 敘述的一般形態	42
1.8.6	ENTER 敘述的一般形態	43
1.8.7	EXTERNAL 敘述的一般形態	44
1.8.8	COMMON 敘述的一般形態	44
1.8.9	標籤COMMON 敘述的一般形態	45

1.8.10	EQUIVALENCE 敘述的一般形態	46
1.8.11	DATA 敘述的一般形態	47
1.8.12	BLOCK DATA 副程式的一般形態	47

第2章 FORTRAN的運用

2.1	程式之執行	51
2.1.1	指令	51
2.1.2	位址	52
2.2	資料之記憶形態	53
2.2.1	二進數	53
2.2.2	整數型	54
2.2.3	實數型	55
2.2.4	字元型	56
2.3	加、減、乘、除四則運算	58
2.3.1	敘述	59
2.3.2	敘述的順位	62
2.3.3	運算的順位	62
2.3.4	縮短計算時間與節省場所	63
2.4	輸出入敘述	64
2.4.1	READ 指令	64
2.4.2	WRITE 指令	66
2.4.3	輸出入運算	67
2.4.4	比較基本的程式	69
2.5	分歧指令	70
2.5.1	無條件GO TO	70
2.5.2	算術IF	71

6 FORTRAN程式範例

2.5.3	邏輯 I F	71
2.5.4	有條件 G O T O	72
2.5.5	指派 G O T O	73
2.6	完全的程式	77
2.6.1	敘述	77
2.6.2	流程圖	78
2.6.3	程式	78
2.7	資料型的處理	80
2.7.1	資料型的指定	80
2.7.2	文字符號型	82
2.7.3	複數	82
2.7.4	資料型的順位	84
2.7.5	資料的起始值	85
2.7.6	資料型的運用	88
2.8	副程式	91
2.8.1	敘述函數	94
2.8.2	FUNCTION 副程式	94
2.8.3	SUBROUTINE 副程式	97
2.8.4	資料的授受	98
2.9	聯立一次方程式	104
2.9.1	問題	104
2.9.2	一般形	106
2.10	註標變數的定義	107
2.10.1	區域之定義	107
2.10.2	區域之共有	109
2.11	註標變數的演算	111

2.11.1	註標.....	111
2.11.2	D O敘述.....	111
2.12.	註標變數的讀寫.....	114
2.12.1	資時記錄.....	114
2.12.2	資料的讀入.....	116
2.12.3	可變FORMAT.....	116
2.12.4	資料的起始值.....	118
2.13.	註標變數的授受.....	120
2.13.1	C O M M O N敘述.....	120
2.13.2	引數.....	121
2.14.	解聯立方程式的副程式.....	123
2.15.	利用磁帶、磁碟的記憶.....	125
2.15.1	順序存取.....	126
2.15.2	直接存取.....	128
2.16.	實際的程式.....	132
2.16.1	讀入寫出的格式.....	132
2.16.2	流程圖與程式.....	138

第3章 FORTRAN的程式範例

積分的問題.....	143
方程式的根.....	154
常微分方程式的起始值的問題.....	167
分佈圖製作.....	177
圖表製作.....	188
利用蒙地卡洛法的 π 之計算.....	207
佇列的模擬.....	210

8 FORTRAN 程式範例

最短距離.....	219
庫存計算（主檔的更新）.....	229
火柴棒遊戲.....	243

第 1 章

FORTRAN的文法

1.1

FORTRAN程式

FORTRAN 程式的第一行通常寫 80 個字，此 80 個字就在卡片 (Card) 上被打孔 (Punch)。並且從左至右編號，此號就相當於第幾行 (Column)，譬如說第 20 號就是第 20 行等等。

程式的一個式或指令 (Instruction) 是寫於第 7 行至第 72 行為止，即使後面的行空下來，也不寫其它的式或指令。寫在此 1 列之式或指令稱為敘述 (Statement)。一個敘述無法收容於 1 列 (從第 7 行起至第 72 行為止) 時，可以按在下一列 (下一張卡片) 也無妨，但是這時候必須在下一張卡片的第 6 行打上 0 以外的文字或數字。這種卡片叫做連續卡 (Continuation Card)。在一個敘述裡面，空白 (Blank) 除了當作文字的意思以外，一切全部都被忽視。

10 FORTRAN程式範例

在由 1 列 80 行所構成的紙上寫敘述，稱為**寫碼** (Coding)。

程式通常是按照被讀進於電腦裡面的卡片，從上面的敘述向下一個敘述順序被**執行** (Execution)，但若有必要，也可以向這以外的敘述轉移過去。這時候有該執行的**敘述號碼** (Statement Number)，此號碼是藉從第 1 行至第 5 行編以號碼藉以表示。但是，連續卡的第 1 行起至第 5 行則被忽視。

從第 73 行至第 80 行和程式的執行無關，通常是例如敘述的編號，程式的名字等，由寫程式的人便於處理卡片使用。如果將文字 C 記入第 1 行，那麼該卡片就叫做**註解卡** (Comment Card)，和程式的執行沒有關聯。此卡片用來說明式或名字，或用來找出敘述。

1.2 常數、變數、註標變數

1.2.1 整常數

這是不含小數點的正或負之數，由符號 +，- 與數字 0 ~ 9 表示。

例 0, 12 + 34 - 590

★ 不註明符號 + 亦無妨。

1.2.2 實常數

這是含有小數點的正或負之數，使用文字 E 或 D 也能藉以表示 10 的乘幂。但是只有 2 倍精度時才使用 D。

例 0.0 -27.0 1.0E 24 ($=1 \times 10^{24}$) 2.5E-5 ($=2.5 \times 10^{-5}$)

2 倍精度 : 13.563482953734

1.2.3 複常數

用逗點隔開 2 個常數，用括符圍起來表示。左邊的數就是實數部分，右邊的數表示虛數部分。

例 (3.2, -8.51) (-5.0E+3, 6E+2) (2.1, 0.0)

1.2.4 邏輯常數

有下面 2 種表示。

.TRUE. .FALSE.

1.2.5 整變數

這是處理整數的變數，由 I, J, K, L, M, N 其中一個字元為字首的 6 個字以內的文字或數字來表示。

例 I INT J5 JHON MAX

1.2.6 實變數

這是處理實數的變數，由 A~H, O~Z, \$ 的其中一個字元為字首的 6 個字以內的文字或數字所表示。

例 A TEMP BETA RATE \$5

1.2.7 IMPLICIT 敘述的一般形態

12 FORTRAN程式範例

IMPLICIT type*s (a₁, a₂,),, type*s (a₁, a₂,)

type: 記入其中之一

INTEGER: 整數 REAL: 實數 COMPLEX: 複數

LOGICAL: 邏輯

s: 表示位元組數。詳細第 2 章

a₁, a₂,; 記入 1 個字母。使此英文字母作為第 1 個字母的變數, 有被指定的形式的性質。

整數、實數以外的變數, 是藉此敘述指定此類型。

例 IMPLICIT COMPLEX(A, B, X)

此例是以 A, B, X 作為第 1 個字母的變數, 變成複變數。

1 按照字母順序連續的一連串文字, 如果有同一性質時, 使用一符號連接該一連串字母的最初與最後的文字便能表示。

例 IMPLICIT LOGICAL(A - F)

在此例, 以 A, B, C, D, E, F 為第 1 字母的變數, 可以認為是邏輯變數。

2 關於整數、實數, 根據 1.2.5、1.2.6 的定義, 以此 IMPLICIT 敘述的定義為優先。

例 IMPLICIT REAL(I, J)

3 關於 S 所示的位元組數, 可以有下面之數。

INTEGER 2 或 4 (標準是 4)

REAL 4 或 8 (標準是 4)

COMPLEX 8 或 16 (標準是 8)

LOGICAL 1 或 4 (標準是 4)

在敘述, 沒有設定位元組數 S 時, 則用標準的位元組數。

DIMENSION $a_1(k_1), a_2(k_2), \dots, a_n(k_n)$

$a_1, a_2, \dots, a_n$: 變數名

$k_1, k_2, \dots, k_n$: 從 1 至 7 個為止的整數

對於 1 個變數名稱，這是使數個呈連續的記憶位址對應時所需要的敘述。

例 **DIMENSION** A (10 , 5), IX (10 , 10 , 10 , 10)

在此例，對變數 A 須準備 50 (10×5) 個記憶位置，對 IX 則準備 10000 ($= 10 \times 10 \times 10 \times 10$) 個記憶位置。

★ 在 **DIMENSION** A (10 , 5)，則按照

A (1 , 1) , A (2 , 1) , ... , A (10 , 1)

A (1 , 2) ... , A (9 , 5) , A (10 , 5)

的順序準備記憶位置。

1.2.9 註標變數的一般形態

$a(k_1, k_2, \dots, k_n)$

a : 變數名

$k_1, k_2, \dots, k_n$: 註標 (n 最高至 7 為止)，有下面的 7 個形式。

v

c'

$v + c'$

$v - c'$

$c * v$

$c * v + c'$

$c * v - c'$

但是， v : 整變數

c, c' : 整常數

14 FORTRAN程式範例

在 DIMENSION 敘述所表示的變數當作註標數使用於此程式裡面。

例 $A(10) B(I) X(I1 + 2, J1 - 1)$

102010 類型敘述的一般形態

Type * s $a_1(k_1)/x_1/, a_2(k_2)/x_2/, \dots, a_n(k_n)/x_n/$

Type : 記入下面其中之一

INTEGER : 整數 REAL : 實數 COMPLEX : 複數

LOGICAL : 邏輯

s : 位元組數

$a_1, a_2, \dots, a_n$; 變數名

$k_1, k_2, \dots, k_n$; 表示變數 a 的 DIMENSION 尺寸。

$x_1, x_2, \dots, x_n$; 欲在變數 a 的初期值時記入其常數。

也可以代替 IMPLICIT 敘述來指定變數的類型，同時也能完成 DIMENSION 敘述的任務。

- 1 如果 DIMENSION 尺寸 k 之值是 1 時，也可以省略。
- 2 不需要初期值時，可以省略其資料 $/x/$ 。
- 3 關於整數、實數，根據 1.2.5, 1.2.6, 1.2.7 之定義，以此類型敘述的定義為優先。

例 INTEGER TE/10/, TIME

將 TE 與 TIME 都定義為整變數，具有此 TE 獲得 10 之值的意思。

- 4 欲在呈連續的位置得幾個同樣的起始值時，也可以用 $n * x$ 這種表示方法。

例 COMPLEX A(10)/(10, 0.0), 9*(0.0, 0.0)/
此例是對變數複數 A 取 10 個記憶位置，其中開始的 A(1) 得 1.0，剩下的 9 個 A(2)~A(10) 得起始值 0.0。