

科學圖書大庫

科學和工程上的PASCAL

譯者 石詔明

徐氏基金會出版

科學圖書大庫

科學和工程上的PASCAL

譯者 石詔明

徐氏基金會出版

徐氏基金會科學圖書編譯委員會

監修人 徐銘信

發行人 呂幻非

科學圖書大庫

版權所有



不許翻印

中華民國七十四年六月十八日初版

科學和工程上的PASCAL

基本定價 3.80

譯者 石詔明 健行工專電機科副教授

本書如發現裝訂錯誤或缺頁情形時，敬請「刷掛」寄回調換。 謝謝惠顧

局版臺業字第3033號

出版者 財團法人 徐氏基金會出版部 臺北市郵政信箱 13-306 號

發行者 財團法人 徐氏基金會出版部 郵政劃撥帳戶 00157952 號

承印者 大原彩色印製有限公司 台北市武成街三五巷九號

電話 9221763
9271575
9271576
9286842

電話 3070998

譯者序

本書包含有 112 個程式，該所有的程式均已在具有 64K memory 和 Apple Pascal 系統之 Apple II Plus 和 Apple IIe 微電腦測試過。在每一個應用裡包含有程式理論和演算 (algorithm) 的簡介，完整程式之列出，一範例運行和其結果的解釋。

本書共分 12 章，其內容分別是：

1. 一般的數學、2. 工程數學、3. 複數數學、4. 矩陣數學、5. 資料分析—它包含有座標軸轉換，導函數，積分，微分方程組，矩陣運算，資料分析之統計以及曲線填配等等。

6. 基本電學和 7. 基本電子學是處理電阻標示到諧振電路的計算。其他的五章是：

8. 電腦補助之電路設計、9. 主動濾波器設計、10. 通訊、11. 被動濾波器和 12. 衰減器裝填。

本書裡的程式均寫得很清楚，故很容易閱讀和修改，因此可以作為結構程序規畫 (Structured programming) 的模式，同時它也可以供作理工科同學或者是有志於電腦之工程人員的參考資料，尤其在電機、電子方面，於電路設計和分析時本書應該是一本相當好之輔助教材，本書亦可供作專科學生作數值分析之實習教材。

本書每節之最後一部分是顯示於螢幕上之例題，最後本書是譯者於課餘時翻譯的，由於資訊之發展神速，以及譯者才疏學淺，若有不當之處，祈請讀者諸君見諒。

石 詔 明

一九八五年二月七日於台北

原 序

本書之目的有二。首先，它是個立即的軟體圖書館，其中包含有 112 個 Pascal 程式，當任何時候需要時即可檢索。這些程式主要與數學和工程有關，尤其是電機方面。它們是好幾百小時工作的結果，今邀請您來使用它。

本書的第二個目的是藉着許多 Pascal 程式上之技巧，來作為新近之 Pascal 程式設計師的指引。於許多場合裡，類似之工作被以不同之方法設計，來證明不僅是只有一種正確的方法來寫程式。此外，程式是模組化 (modular) 的，因此可以很容易地加進你自己的程式。

電路設計之程式將需要一些額外的證明事項，如電路圖，但是大部分的圖是單獨的意思。圖裡並不包含有資料，它通常是在每一程式列表時詳細寫出。

本書裡所有的程式是用 Apple II Plus 電腦和 Apple UCSD Pascal 來寫和測試的。這使得大部分的程式非常地輕便。當程式內陳述 “ Uses Transcend ” (Transcend = 超越) 時，檢驗你的 Pascal 版號 (Version)，看看要使用什麼。

所有的程式都用相同的標題 (header)，它問道：

S (screen or P (rinter ?

輸入 P 或 p，其輸出報告只會送到印表機 (printer)。所有其他之輸出，包含程式提示 (prompts)，將顯示在螢幕上。

所有的程式都是已設計好，使它們能完全地再運行 (rerun)。這項特徵加入來適應 Pascal 操作系統之緩慢。假如輸入一些資料來保存結果，程式能再度地運行，且可選用螢幕或印表機。

變數的名稱使得所撰寫的程式儘可能的被瞭解。

在樣品運行時，若超過一套的資料被使用——如當問題“另一套資料（Another set of data）？（Y/N）”之解答是“Y”時——在上面之標題和程式之名稱通常會被重複。然而，為節省這種重複，標題在例子裡並不出現，而名稱僅顯示一次。

Apple II Plus之列表是在其常態之40行螢幕上工作。若是如此裝備，在鍵盤輸入（Key-in）期間，能很容易地改變成80行。

祝操作時愉快！

JULES H. GILDER 和 J. SCOTT BARRUS

需要的設備

爲了能在 Apple 電腦上使用本書的程式，需要下列設備：

64K RAM的 Apple II Plus 或附有語言卡 (Language card) 之 48K RAM

Apple Pascal 語言系統

Apple 磁碟機 (Disk Driver) (最好有二部)

顯示之監測機 (Monitor)

印表機 (隨意)

因爲本書上的程式是以 Apple II 電腦來準備的，它們只能在相同之設備上作不必修正之處理。若您有不同系統之電腦，您仍然可以使用這些程式，但是在以您的電腦在編譯和運行之前，必須改變在此之列表。

在寫的同時，UCSD Pascal 曾經在下列之所有的處理機裡執行過。
〔來源：軟體微處理機系統 (Softech Microsystems) 〕

6502	68000	9900
PDP-11	LSI-11	6809
Z 80	8080	8085
8086	8088	

若您的電腦是使用上述之一的處理機時，則 UCSD Pascal 之預期規畫之說法是有其可利用之機會。

目 錄

譯者序	I
原 序	II
需要的設備	V
第一章 一般的數學	1
1-1 階 乘	2
1-2 推廣的階乘	5
1-3 排 列	7
1-4 組 合	10
1-5 改良的組合	13
1-6 兩向量之純量積	16
1-7 兩向量之叉積	18
1-8 任意底數之對數	21
1-9 新座標	24
1-10 直角—極座標轉換	27
1-11 極大 / 極小定位	30
1-12 尺寸換算	32
1-13 梯級圖	36
1-14 圓的尋求	40
1-15 常態分佈的隨機數	43

1-16	一數的 n 次根	47
1-17	有理分數	49
第二章 工程數學		53
2-1	聯立方程式	54
2-2	多項式求值子	59
2-3	二次方程式解算機	64
2-4	牛頓 - 拉富生根	67
2-5	函數的導數	70
2-6	辛普森規則之積分	73
第三章 複數數學		79
3-1	複數加法	80
3-2	複數減法	83
3-3	複數乘法	87
3-4	複數除法	90
3-5	絕對值	94
3-6	複數的實數乘冪	97
3-7	複數根	101
3-8	複數指數	105
第四章 矩陣數學		111
4-1	二維方矩陣之加法和減法	112
4-2	二維矩陣乘法	116
4-3	二維矩陣之轉位	119
4-4	二維矩陣之純量乘法	123
4-5	矩陣反算	126
4-6	三維矩陣之加法和減法	131
4-7	三維矩陣之乘法	135
4-8	三維矩陣之純量乘法	140

第五章	資料分析	145
5-1	幾何平均值	146
5-2	平均值和中值計算	149
5-3	方差和標準偏差計算	152
5-4	三點內插法	155
5-5	四點內插法	159
5-6	線性最小平方填配	162
5-7	最小平方填配—對數縱座標	166
5-8	最小平方填配—對數橫座標	171
5-9	最小平方填配—全對數	176
第六章	基本電學	183
6-1	電阻器色碼標示	184
6-2	並聯電阻器	189
6-3	Δ 到Y轉換	192
6-4	Y到 Δ 轉換	196
6-5	由電阻和長度得線徑	200
6-6	由電流容量得線徑	203
6-7	導線電阻計算器	206
6-8	變壓器匝數比	208
第七章	基本電子學	213
7-1	直導線電感	214
7-2	平行導線對的電感	217
7-3	平行導線圓形組的電感	220
7-4	容抗計算器	224
7-5	感抗計算器	227
7-6	旁路電容器計算器	229
7-7	諧振電路計算	232

第八章 電腦輔助之電路設計 237

- 8-1 A類電晶體放大器設計 238
- 8-2 舒密特觸發電路設計 243
- 8-3 電源供應器之持續時間 247
- 8-4 無變壓器的電源供應器 250
- 8-5 二個極體的全波調節電源供應器 255
- 8-6 橋式整流的調節電源供應器 259
- 8-7 雙輸出調節電源供應器 263

第九章 主動濾波器設計 269

- 9-1 主動通頻帶濾波器 270
- 9-2 狀態可變之通頻帶濾波器 275
- 9-3 多回授通頻帶濾波器 279
- 9-4 韋恩電橋凹口濾波器 284
- 9-5 並聯T型凹口濾波器 289
- 9-6 第一號主動低通濾波器 293
- 9-7 第二號主動低通濾波器 299
- 9-8 巴特威士-湯普森低通濾波器 304

第十章 通 訊 311

- 10-1 高斯脈衝頻寬計算 312
- 10-2 微波導片設計 315
- 10-3 條狀線設計 319
- 10-4 無損失傳輸線 322
- 10-5 單導線傳輸線 326
- 10-6 雙導線傳輸線 329
- 10-7 RF 空心電感器設計 331
- 10-8 電壓駐波比率計算 335
- 10-9 分貝轉換 337

第十一章	被動濾波器	341
11-1	低通定K式 π 型濾波器	342
11-2	低通定K式T型濾波器	345
11-3	低通M推演式T型濾波器	347
11-4	低通M推演式L型濾波器	351
11-5	低通分路M推演式L型濾波器	354
11-6	低通分路M推演式 π 型濾波器	357
11-7	高通定K式 π 型濾波器	360
11-8	高通定K式T型濾波器	363
11-9	高通，串聯，M推演式T型濾波器	366
11-10	高通分路M推演式 π 型濾波器	370
11-11	通頻帶定K式 π 型濾波器	373
11-12	通頻帶定K式T型濾波器	376
11-13	除頻帶定K式 π 型濾波器	380
11-14	除頻帶定K式T型濾波器	383
第十二章	衰減器裝填	387
12-1	平衡—橋式H型衰減器	388
12-2	對稱不平衡—橋式T型衰減器	392
12-3	不對稱H型衰減器	395
12-4	不對稱方型衰減器	399
12-5	對稱方型衰減器	404
12-6	對稱H型衰減器	407
12-7	對稱格子型衰減器	411
12-8	對稱 π 型衰減器	414
12-9	對稱T型衰減器	416
12-10	不平衡最小損失衰減器裝填	419
12-11	平衡最小損失衰減器裝填	423

第一章 一般的數學

(GENERAL MATHEMATICS)

1.1	階 乘	2
1.2	推廣的階乘	5
1.3	排 列	7
1.4	組 合	10
1.5	改良的組合	13
1.6	兩向量之純量積	16
1.7	兩向量之叉積	18
1.8	任意底數之對數	21
1.9	新座標	24
1.10	直角—極座標轉換	27
1.11	極大 / 極小定位	30
1.12	尺寸換算	32
1.13	梯極圖	36
1.14	圓的尋求	40
1.15	常態分佈的隨機數	43
1.16	一數的 n 次根	47
1.17	有理分數	49

大部分的家用電腦都附有種類甚多之內在數學功能。然而，在許多場合裡，當它們不夠用，而需要更多的潛在能力時，本章內之程式是部分設計來推廣您機器的能力。

於某些程式裡（例如：推廣的階乘和改良的組合），計算機被以程式來處理一些數字，其大小一般會引起溢位。藉着小心地分析這種程式技巧，而可能藉着它來使用於其他方面的應用。

除了與機率相關之程式（第 1-1 節到第 1-5 節之外，第 1-6 節和第 1-7 節也證明向量的特殊數學能夠如何被處理。Pascal 具有 Log （10 為底）和 Ln （自然對數）的二種能力。於第 1-8 節裡的一個簡單的程式可供以任何作底之對數來工作之能力。

本章裡有兩個程式是用來處理座標系統。首先於第 1-9 節提供轉移或旋轉一已知座標系統之能力，若需要將一資料或測量自一參考座標轉換成另一種參考座標是很方便的，另一種與座標有關之程式是第 1-10 節，它是將資料由笛卡爾（直角）座標，轉換成極座標；反之亦然。

下一個程式是處理數字，若於處理資料繪圖時，第 1-11 節的程式可找出一已知數裡的極大和極小值，它是於繪圖時絕對需要的。

若你曾經將一物體準確地繪於一張紙上，你將會很快地明瞭尺寸換算程式（第 1-12 節）的價值。若你想做一些繪圖，則梯級圖（histogram）程式（第 1-13 節）會產生很吸引人的結果。如其所示，它被用來繪函數圖。然而，不費怎麼的努力，它能被用來繪製個別的資料點，將第 1-13 節和 min/max 定位（locator）程式組合可用來解此問題。

你曾經用三個點來畫一個圓嗎？試試看，在一圖表紙上繪三個點，且試着繪一圓通過此三個點。它並不容易，現在將這三個點的位置輸入第 1-14 節的程式，它將會告訴你圓的半徑和圓心的位置。其餘的程式產生一般分佈之數，以及如求任意數之任意根，和將任意數轉換成有理分數。

1-1 階 乘

一個數的階乘在機率計算裡是常常被使用到的運算。由下列之計算

可得：

$$n! = n(n-1)(n-2)\dots 1$$

$$5! = 5(4)(3)(2)(1) = 120$$

$n!$ 的符號常用來指定階乘的運算，而讀作“ n 階乘”。 $0!$ 是一特例，其大小規定是 1。階乘的計算是在過程 `Fact ( Procedure Fact )` 上執行。其可變“因數 (factor)”預置 (initialize) 為 1，若不如此做，Apple Pascal 將預置數字為任意數。程式循環通過計算 `factor := factor * variable` 通過 `number` 週期。若 `number` 等於零，程式將再次循環，且 `factor` 將等於 1。

```
(*****
(*                                     *)
(* Title: Factorial                  *)
(* Program Summary: Program computes *)
(*   the factorial of a given       *)
(*   number.                        *)
(*                                     *)
(*                                     *)
(*****)
```

Program FACTORIAL;

```
Var
  number,
  variable      : Integer;
  factor        : Integer[35];
  answer        : Char;
  out           : Text;
  cont          : Char;
```

```
Procedure Outputs;
begin
  page(output);
  repeat
    gotoxy(0,6);
    Write('Screen or Printer? ');
    readln(answer);
  until answer in ['s','S','p','P'];
end; ( outputs )
```

```
Procedure Title;
begin
  page(output);

  Gotoxy(15,5);
  Writeln('Factorial');
  Writeln;Writeln;
  Writeln('This program computes the factorial of');
  Writeln('a given number. Operation is limited ');
  Writeln('by the largest number your computer can')
```

```

    Writeln('handle. Only integers are valid.');
```

end; (title)

```

Procedure Fact;
begin
    factor:=1;
    for variable:= 1 to number do
        factor:= factor * variable;
    end; ( factor )

Procedure Input;
begin
    Write('Enter number: ');
    Readln(number);
    end; ( input )

Procedure Printout;
begin
    case answer of
        's','S' : begin
                    Writeln(number,' != ',factor);
                    end;
        'p','P' : begin
                    rewrite(out,'printer:');
                    Writeln(out,number,' != ',factor);
                    Writeln(number,' != ',factor);
                    close(out);
                    end;
    end; ( case of )
end; ( printout )

begin ( main program )
    Repeat
        Outputs;
        Title;
        Writeln;Writeln;
        Input;
        Writeln;Writeln;
        Fact;
        Printout;
        Writeln;Writeln;
        Write('Do you have another number?(Y/N) ');
        Readln(cont);
    until cont in ['n','N'];
end.
```

階 乘 (Factorial)

This program computes the factorial of a given number. Operation is limited by the largest number your computer can handle. Only integers are valid.

Enter number: 9

9! = 362880

Do you have another number? (Y/N) Y

Enter number: 35

(system overflow error)

1-2 推廣的階乘¹

先前程式之主要問題是，會很快的到達電腦之數字處理之極限。大部分的家用電腦當其計算結果的數字大於 10^{39} 時會過載。這個程式含有一個別的例程序（routine），而使解答分解成 10 的乘冪，而能表現出更大的能力（Procedure Fact 之後半部）。

如先前之程式，階乘是在 Procedure Fact 裡計算。而使大於 121! 之數能很容易地計算出，然而是在它們使電腦過載之前算出。

```
(*=====*)
(*                                     *)
(* Title:  Extended Factorial         *)
(* Program Summary: Program computes *)
(* the factorial of a given          *)
(* number.  This program is          *)
(* like factorial, but it will        *)
(* accept larger numbers.            *)
(*                                     *)
(*=====*)
```

Program EXTENDFACTORIAL:

```
Var
  expo,
  variable,
  number           : Integer;
  factor           : Real;
  answer           : Char;
  out              : Text;
  cont             : Char;
```

```
Procedure Outputs;
begin
  page(output);
  repeat
    gotoxy(0,6);
```

¹ 取自 W.M.Bunker 著的 “Computer Program Extends Computation of Factorials” 刊於 Electronic Design (1971 年四月) 之 p.86