程序设计

复旦大学数学系 編著

上海科学技术出版社

程序設計

（試用本）

复旦大学数学系　編著

上海科学技术出版社

內 容 提 要

本书是复旦大学数学系計算数学专业教材之一，內容介紹了用电子計算机解决各种問題时經常遇到的一些程序設計問題，例如各种子程序、特殊运算等，并介紹了有关程序設計自动化的算子图、程序图的概念和苏联快速电子計算机 БЭСМ 的一个自动程序。本书可作綜合大学計算数学专业程序設計課程的教材，亦可作高等院校有关专业的参考书。

程 序 設 計

（試 用 本）

复旦大学数学系 編著

*

上海科学技术出版社出版

（上海瑞金二路450号）

上海市书刊出版业营业許可証出093号

新华书店上海发行所发行 各地新华书店經售

大众文化印刷厂印刷

*

开本850×1168 1/32 印張5 8/32 字数124,000

1960年9月第1版 1960年9月第1次印刷

印数1—6 000

統一书号：13119·388

定 价：（十）0.62元

編輯說明

我們受到全国持續跃进的大好形势的鼓舞和推动，积极响应了党的号召，在两年来教育大革命已經取得偉大成績的基础上，掀起了一个声势浩大的教学改革的群众运动。通过这个运动，我們揭露了現在教学体系、教学內容和教学方法上陈旧落后的状况，抓住訂方案、編大綱、写教材、搞試驗等重要环节，試图建立一套以馬克思列宁主义、毛澤东思想为指导的，反映現代科学发展水平的，理論联系实际的新的教学体系和內容，以及与之相适应的教学方法，使培养人才的工作更好地貫彻党的社会主义建設总路綫的精神。

作为这种新的探索和尝試，我們在教学改革运动中，师生結合編写了一套数学专业試用的基础課程教材。在这个基础上，我們又編写了計算数学专业試用的基础課程教材，这套教材包括計算方法、程序設計、綫性代数、程序实习四种。我們在編写这套教材的过程中力求遵循以下几个原則：

一、根据我国社会主义建設和現代科学发展的需要选取材料。例如在計算方法中选取了最近随着力学等方面发展而产生的一些新的非綫性方程計算方法；程序設計中选取了标准程序及自动程序等內容。

二、在材料的选取和闡述上注意到"实踐——理論——实踐"的原則。例如在計算方法及程序設計中，新概念的引进一方面注意到它的实踐来源，同时又尽可能指出所得結果的实踐意义。在材料的选取上还注意到学科間的相互联系。例如在計算方法課程

中注意到与数学分析、高等代数、程序实习等課程的相互配合；又如在程序实习課程中注意到与程序設計、数学物理方程、計算方法等課程内容的衔接。

三、根据各課程特点，在編写时也注意到教学方法，使更有利于同学掌握与运用。例如在程序实习中，强調同学自己动手編制程序，并进行群众性的課堂討論和总結提高的教学方式。

由于我系計算数学专业成立于1958年，絕大部分同志缺乏教学和实际工作的經驗，因此这套教材一定有很多不足之处，我們恳切地希望讀者提出批評与指正。

上海科学技术出版社和商务印书館上海印刷厂对这套教材的迅速出版，給了极大的支持，我們在这里表示衷心的感謝。

复旦大学数学系

1960年5月

序

現代高速数字电子計算机的誕生，是生产技术和科学研究高速发展的重要标志之一。这一現代計算工具正在日益广泛地被用来进行大量繁复的数学运算，从而促进了計算数学的大发展。使用电子計算机进行計算时必須运用所謂机器的語言，于是就逐步形成了程序設計这門学科。为了更有效地使用机器，最迅速地取得計算結果，減輕人們的劳动，程序設計的自动化就成为这門学科的重要发展方向之一，但現有文献中还缺乏完整的資料，它正在不断发展而日臻完善。

目前在国內关于程序設計的教材还比較少，鑒于这种情况，我們尝試編写了这本书，企图使讀者在具有初步程序設計知識的基础上，能以較短的时間，进一步深入学习。

本书在材料选擇上以密切联系当前社会主义建設为原則，講述了在使用机器解决各种生产問題时所經常遇到的一些程序設計問題，例如在教材的第一章，我們以較大篇幅来介紹許多被广泛使用的标准程序，这些內容大多取材于实踐；另外，我們介紹了扩充机器性能的几种特殊运算，例如定点机上的浮点运算系統，复数运算系統等，它們能帮助我們解决一些較广泛的生产問題。同时，当前科学研究和生产单位提出的大量数学問題，迫切要求計算数学工作者以高速度編出程序，因此，程序設計自动化的研究工作也提到日程上来了。在教材的后一部分我們介紹了有关算子图、程序图的概念，并对苏联快速电子計算机 БЭСМ 的一个自动編制程序的程序作了較詳細的介紹，可以作为讀者在开展这方面工作时的

参考，也可作为程序組織的参考方法。

限于我們的經驗和水平，本书难免有缺点和錯誤之处，迫切希望各方面多提意見，以便改正。

复旦大学数学系程序設計編写小組

1960年5月

目　录

第一章 子 程 序

在解决生产問題的計算工作中，許多程序往往具有公用性，在同一問題中，也常有一些程序可用于不同的时机，因而严密地組織程序設計工作，可大量节省編制程序的人力和物力。在一个問題中，当你把問題的各个計算部分分析清楚，找出了某些結构相同的算程之后，就有可能去組織公用的子程序。对一个計算单位来說，各个問題中常見的計算方法，編成标准程序就可以节省具体問題中的大量程序設計工作。不仅如此，有計划地事先編好一些标准程序及标准子程序，往往还可以提高程序水平，可以作为培訓干部的方法，也可作为計算工作者吸取程序技巧、組織思想的参考資料。本章介紹代数方面的几个典型問題、微分方程組数值解法中的三个典型方法及二次插值的标准程序等材料。由于我們的主題是討論程序設計的方法，故一些計算方法的論証推导从略。

§1 标准程序系統及一般問題

在現有資料中，由于生产問題的經常需要，我們把下列方案編成标准程序：

(1) 求行列式的值；

(2) 求多項式的根；

(3) 求綫性代数方程組的解；

(4) 求方陣的特征多項式的系数；

(5) 求逆陣；

(6) 二次插值;

(7) 三次插值;

(8) 求平方逼近多项式的系数;

(9) 常微分方程組的龙格-庫塔方法;

(10) 常微分方程組的欧拉-阿当姆斯方法;

(11) 其他。

我們所指的标准程序系統，特别强調这些程序要尽可能成套，規格統一。成套是指常用方法都包括进去，如果有相互衔接套用时要保証規格上能衔接。規格統一是指一些技术措施尽可能一律，以便使用人掌握这些資料。

标准程序与标准子程序不同之点，在于标准程序是根据一串已知量求出一串未知量，而标准子程序只根据一个或二个未知量去計算一个或二个函数值。共同之点在于这二类程序都是从第一条指令接受控制，然后从指定单元（所謂标准自变单元）去取已知量，将算出的量放到指定的单元中去（标准函数单元）。工作完毕，控制轉向标准单元，例如 d004。由于标准程序的任务不同，类型就較复杂，因而有时不能象标准子程序那样規定得很具体，相反地要让使用者有一定的自由。例如求多項式的根，允許使用者把系数及有关数据放在他認为方便的地方而把"地址"通知标准程序，甚至多項式的項数也允許使用人任意改变。标准程序系統一般地說有三个共同問題，分别介紹于下。

一、标准程序的恢复問題

标准子程序工作完毕后能否再度工作？某些标准程序工作一次即自行破坏，不能再次工作，但通常編出的标准程序尽量采取能恢复的方案。其次，标准程序常常从大量标准单元中取已知数，算出的函数个数也往往很多，这样，某些标准程序就不能象标准子程序那样保持标准单元中的已知数。例如求行列式的值有 n^2 个已知

数，如果要求在标准程序工作过程中保持这些单元之值不被改动，将是一件困难或不經济的事情。所以掌握标准程序的資料时，一定要关心每一个标准程序对它自己及标准单元（自变量）的"破坏"情况，就是說这些单元是被保留的还是不被保留的。

二、标准程序的标准单元及信息

通常标准程序的标准单元是不預先固定的（例如 x^m 标准子程序的自变量 x 及 m 放在标准单元 d001 及 d003，而函数值放在标准单元 d002，这些規定一般情况下是固定的）。例如求行列式 $|a_{ij}|$ 之值时，a_{ij} 存放的存区是由两个数据 a_{11} 及 n 所給出的，并非把 a_{11} 放在 d001，a_{12} 放在 d002，…，因为主程序把 n^2 个数送到 d001 开始的一連 n^2 个单元中去，比起告訴标准程序〈a_{11}〉及 n 这两个数据要困难得多。所以，通常求行列式值的标准程序中包括 H 和 $C\Pi$ 二部分。$C\Pi$ 是計算行列式值的程序，而 H（称調整程序）是把主程序提出的数据〈a_{11}〉及 n 填到 $C\Pi$ 的有关指令或常数中去，使 $C\Pi$ 工作时直接到〈a_{11}〉开始的 n^2 个单元中去取数。以下我們来討論主程序与 H，$C\Pi$ 之間的衔接問題。

1．主程序把 00 n 〈a_{11}〉（称标准程序的調整信息）送 d001，指令型返回常数放在 d004，控制轉向 H 的第一条指令 $H000$。

2．H 从 d001 取得〈a_{11}〉及 n 后，填到 $C\Pi$ 的有关单元中的有关数位上去。全部填好之后控制轉向 d004。

3．主程序从 d004 接回控制后去做它所要做的一些工作，例如把 $a_{11},\cdots,a_{nn}$ 送到〈a_{11}〉，…，〈a_{nn}〉中去，或輸入 a_{ij}，或算出 a_{ij}（这件事也許早就做好，那么現在可以不做），把指令型返回常数送 d004，而后控制轉向 $C\Pi$ 的第一条指令 $C\Pi00$。

4．$C\Pi$ 从 $C\Pi00$ 开始工作，算出結果放 d002（一般标准程序把結果送到信息指出的单元中去）。工作完毕，控制轉向 d004。

5．主程序由 d004 接回控制后去做其他工作。

三、二进二出和信息处理

首先，为什么主程序一次轉入 H，一次轉入 CII（称二进二出）呢？如果由 H 直接轉向 CII，形式看来主程序是会省掉一些指令的，但由下列几个原因，可显示出"二进二出"的优越性。

1. 某些标准程序的 CII 部分很长，H 也很长，自变量很多，三者同时存入内存储会有困难。如果用"二进二出"的办法，当 H 工作时 CII 不工作，则自变量可以不輸入；由 H 轉向主程序后，再由主程序控制把自变量輸入到包括原先 H 占用的存区去，这样就可以部分地解决大量自变量的存储紧張問題。

2. 某些标准程序 H 工作一次后，CII 部分要反复多次工作，例如二次插值、三次插值等标准程序，而"二进二出"方案中就具有这种灵活性。

3. 从規格統一这观点来說，一套标准程序中应尽量在"二进二出"問題上取得一致方式。

其次是对信息的处理問題，因为各个标准程序的信息代表着不同的含义，有的一个单元能把所有信息存完，有的标准程序的信息要分别装在几个单元中。从規格統一要求来說，使得主程序通过一个单元 d001 把最关鍵的信息交給 H，由 H 根据 d001 的内容去取得所有技术資料，然后再来进行調整 CII 的工作，这样是比較方便的，因为我們将繁复的工作尽量給 H 来做，而使主程序做最必要的最少的工作。其具体材料可参閱本书龙格-庫塔方法的信息处理方案。

在本书中我們还作这样規定：

信息的内容能在一个单元存完时，信息由 d001 交給 H。

信息的内容能在两个单元存完时，信息由 d001、d003 交給 H。

信息的内容不能在两个单元存完时，所有信息占有的一串单

元出現在主程序的常数表里，第一个信息单元的"地址"放在 d001 的第一地址，H 由 d001 取得調整信息的存区的信息。

§2 代数問題的标准程序

在这一节中，我們先較詳細地講述一种特殊吉德尔迭代法的标准程序，再簡要地介紹多項式求根，求特征多項式系数及求逆陣等标准程序。

一、吉德尔迭代法

这是解綫性代数方程組的間接方法，在程序实习中已經講了这种方法的一个方案，現在介紹性能較好的另一个方案。

設給出 n 阶方程組

$$X=AX+B,$$

其中 A, B 分别为 n 阶方陣及 n 維的列向量

$$A=(a_{ij}) \quad (i,j=1,2,\cdots,n),$$

$$B=\begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}.$$

求解的吉德尔迭代公式为

$$x_i^{(\nu)}=\sum_{j=1}^{i-1}a_{ij}x_j^{(\nu)}+\sum_{j=i}^{n}a_{ij}x_j^{(\nu-1)}+b_i \quad (i=1,2,\cdots,n).$$

用这个方案求解时，在程序上的特点是 X 只用一套存儲单元，而程序的算式就写成

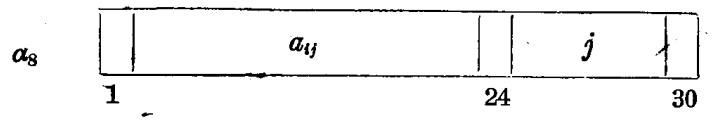
$$\sum_{j=1}^{n}a_{ij}x_j+b_i\Rightarrow x_i \quad (i=1,2,\cdots,n).$$

在程序实习中所講的正是关于它的程序設計問題。但是，在解决实际問題时，我們往往会遇到方陣 A 里出現大批零元素的情形。例如在用差分格式解微分方程边值問題时，所归納出来的綫性代

数方程組就是这种情形。在这种情况下，如果采用上面所說的吉德尔迭代法来求解，工作过程很不經济，因为大批的零元素都参加了运算，并占用了大批单元。但对間接方法而言，我們总可設法使零元素不参加运算。

这一段中，我們提出一种解决零元素不参加运算的方案，从而引进一种特殊的吉德尔迭代程序。

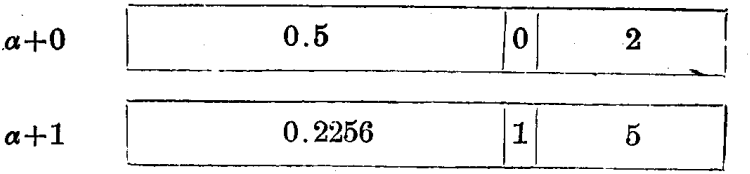
当精确度不是太高时，例如要求誤差不大于 10^{-5} 时，我們采用一种系数带行、列号的存储方案。也就是說，我們这样来存储数 a_{ij}：在一个单元的前 23 位上放数 a_{ij}，第 24 位上放 0 或 1，当放 1 时就表示該系数为同一行中的最后一个非零元素，第 25～30 位上放足标 j 的数值，即



采用了这种記数方法之后，我們就可以除去零元素的存储单元。例如在方程組中有一方程为

$$0x_1+0.5x_2+0x_3+0x_4+0.2256x_5+0x_6+0x_7=0.3,$$

按照原来的系数存储法，就必須要用七个单元，但使用了这种方法后，只要两个单元



就可以了。这样来記系数，也不会使机器計算复杂化，因为“j”的記录指出了該系数应和 x_j 相乘。例如单元 $\alpha+0$ 就說明了应該做 $0.5\cdot x_2$ 这一工作，而第 24 位上的 0 則說明迭代算式的累加过程还要继續去做 $0.2256\cdot x_5$ 的工作，而在做了这个工作以后，我們发

現在第24位上为1，因此我們知道累加的循环过程結束，轉去做下一式的計算工作。

这样一来，我們的存儲单元就可以大大减少了。如果我們采用下面的存儲分配(标准单元的)方案：

$$\langle x_i\rangle = a+i-1,\ 1\leqslant i\leqslant n,$$
$$\langle \varepsilon\rangle = a+n,$$
$$\langle b_i\rangle = a+n+i,\ 1\leqslant i\leqslant n,$$
$$\langle a_{ij}\rangle = a_m,\ m=1,\cdots,m_n,$$

其中 m_n 为系数陣中非零元素的个数，我們就可以作出标准程序的信息及标准程序的邏輯图。

信息：d001 00 n a

邏輯图：

H_1 $1\to i$, $1\to i'$, H_2 $0\to m$, $\llcorner_3$ $0\Rightarrow b$, F_4 $m+1\to m$,

$\llcorner_5$ $a_m\Rightarrow a$, $a^{\text{II}}\to j$, $ax_j+b\Rightarrow b$, P_6 $\alpha_{24}=0\overset{7}{\sqsubset}$ $\llcorner_7$ $b+b_i\Rightarrow b$,

$P_8|x_i'-b|<\varepsilon\overset{9}{\sqsubset}^{10}$ H_9 $(\overset{11}{\sqcap})\Rightarrow P_8$, $\overset{11}{\sqcap}$ F_{10} $i'+1\to i'$, $\llcorner_{11} b\Rightarrow x_i$,

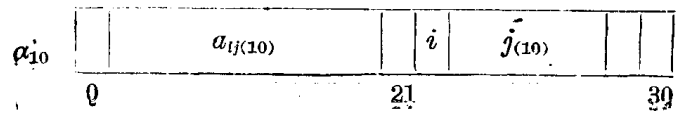
F_{12} $i+1\to i$, P_{13} $i<n+1\overset{14}{\sqsubset}^{3}$ P_{14} $p_8>p\overset{d4}{\sqsubset}^{1}$.

其中 α_{24} 表示 a 的第24位上的内容，p 为 +32 0000 0000 与 +74 0000 0000 之間的某一数。

为了和一般的吉德尔迭代程序加以区别，我們称这个程序为带行列号的吉德尔迭代程序。

由于系数存儲的特殊性，数字部分和行列号的記录是两个独立的部分，不可能当成一个整体来进行翻譯，必須分开翻譯。因此，就需要有特殊的 10→2 标准子程序(并且編成成組翻譯)。

带行列号的系数在輸入机器时有下面的格式：



带行列号吉德尔迭代程序

```
b000  +74        b036
b001  +15  d015  b007
b002  +15  d012  b017
b003  +15  d011  b020
b004  +15  d016  b026
b005  +01  d002  d002
b006  +00  b064  b007
b007 (+00  0000  0000)
b010  +76  b035
b011  +20  d013  b012
b012 (+00  0000  0000)
b013  +20  d002  d002
b014  +16  d003  b062
b015  +31  b062
b016  +34  b006  b017
b017 (+00  0000  0000)
b020 (+00  0000  0000)
b021 (+00  0000  0000)
b022  +34  b025  b023
b023  +15  b024  b020
b024  +74        b026
b025  +00  b064  b020
b026 (+00  0000  0000)
b027  +00  b063  b026
b030  +00  b064  b017
b031  +31  d017
b032  +34  b005  b033
b033  +11  b020  b032
```