



计算机辅助设计丛书

控制系统的 计算机辅助设计 程序汇编 (BASIC 语言)

孙增圻 叶 榛 钱宗华 编
丁冬花 张毓凯 袁曾任



清华大学出版社

控制系统的计算机辅助设计 程序汇编

(BASIC 语言)

孙增圻 叶 榛 钱宗华
丁冬花 张毓凯 袁曾任

编

清华大学出版社

内 容 简 介

本书汇编了近 50 个控制系统 CAD 常用算法程序。全部程序分为六块：(1) 控制系统 CAD 常用基础算法程序；(2) 控制系统仿真；(3) 控制系统模型转换；(4) 应用古典控制理论分析和设计控制系统；(5) 应用现代控制理论分析和设计控制系统；(6) 控制系统的模型辨识。这些程序为控制系统的分析、设计、建模和仿真提供了方便的工具。所有程序采用 BASIC 语言编制，并在 PDP-11/23、Dual 68000 及 IBM/PC 计算机上调试通过。本书给出的是 PDP-11/23 上的程序版本。

本书可与所列参考文献[1]配合使用。文献[1]主要介绍算法，本书介绍实用的计算程序。本书可供从事计算机应用及控制工程的技术人员使用，也可作为高校相应专业的学生、研究生、教师及有关科研人员的参考资料和工具书。

JS/30/26

控制系统的计算机辅助设计程序汇编 (BASIC 语言)

孙增圻 叶 榛 钱宗华 编
丁冬花 张毓凯 姜曾任

☆

清华大学出版社出版
(北京·清华园)

清华大学印刷厂印刷
新华书店北京发行所发行

☆

开本：787×1092 1/16 印张：27.5 字数：704 千字
1988 年 5 月第 1 版 1988 年 5 月第 1 次印刷
印数：00001~8000 定价：7.00 元
ISBN 7-302-00179-0/TP·69

出版说明

计算机辅助设计 (Computer Aided Design) 简称 CAD, 它是近 20 年发展起来的一门新兴技术。随着计算机硬件、软件、图形及智能模拟等方面技术的巨大进步, CAD 技术已成为工程设计及科学研究不可缺少的组成部分。CAD 技术充分利用了计算机的高速运算和数据处理能力, 因此不仅可以缩短产品的设计周期, 减少设计人员的繁杂劳动, 而且能够提高产品质量, 降低成本。CAD 技术与 CAM (Computer Aided Manufacturing——计算机辅助制造) 以及工业机器人相结合是当今工业生产过程自动化的发展方向。同时, CAD 技术也为科学研究的发展和教育的现代化提供了方便的工具。

对于不同的工程领域, CAD 技术的具体内容有很大不同。例如, 电子电路主要用节点网络表示其拓扑结构; 微波电路较多地研究其数学模型及矩阵方程; 机械和建筑设计的关键在于图形技术; 控制系统则主要利用计算机进行动态系统的建模、分析、设计和仿真等。因此, 计算机辅助设计是专业理论、计算方法、计算机软件及图形学等多方面结合的综合技术。

为了普及和推广 CAD 技术, 我们编辑出版一套计算机丛书。其中包括微波电路、机械、建筑、控制系统以及图形学等领域的 CAD 技术的原理及应用。我们希望这套丛书能对我国 CAD 技术的发展及其在工程设计和科学研究中的应用起到促进作用。

目 录

前 言	1
第一章 控制系统 CAD 常用算法程序	3
第一节 求解线性方程组	3
第二节 代数多项式求根	4
第三节 求解常微分方程	6
第四节 矩阵的基本运算	8
第五节 矩阵求逆	10
第六节 矩阵行列式的计算	11
第七节 矩阵特征值和特征向量的计算	12
第八节 多项式基本运算	13
第九节 多项式矩阵相乘	15
第十节 多项式矩阵求逆	18
第十一节 连续李雅普诺夫方程的求解	20
第十二节 离散李雅普诺夫方程的求解	21
第十三节 计算转移矩阵	24
第二章 控制系统仿真	26
第一节 线性连续系统仿真	26
第二节 非线性连续系统仿真	32
第三节 离散系统仿真	37
第四节 采样系统仿真	39
第五节 随机连续系统仿真	46
第六节 随机采样系统仿真	50
第三章 控制系统模型的相互转换	56
第一节 化连续状态方程为离散状态方程	56
第二节 化离散状态方程为连续状态方程	58
第三节 双线性变换	62
第四节 化状态方程为传递函数	66
第五节 化状态方程为能控标准形	68
第六节 化状态方程为能观标准形	71
第四章 应用古典控制理论分析和设计系统	74
第一节 Routh 稳定判据	74
第二节 传递函数的部分分式展开	76
第三节 计算对数频率特性	79
第四节 计算根轨迹	81
第五节 计算校正传递函数	85
第六节 利用长除法求 z 反变换	89

第五章 应用现代控制理论分析和设计系统	92
第一节 按极点配置设计控制规律	92
第二节 按极点配置设计观测器	96
第三节 连续系统线性二次型最优控制规律的计算	99
第四节 离散和采样系统中线性二次型最优控制规律的计算	101
第五节 离散和采样系统中最优状态估计器的计算	104
第六节 连续系统二次型性能指标函数的计算	108
第七节 离散和采样系统中二次型性能指标函数的计算	110
第八节 随机连续控制系统中平均性能指标函数的计算	113
第九节 随机离散和采样系统中平均性能指标函数的计算	114
第十节 离散和采样系统中二次型性能函数及其对控制器参数灵敏度的计算	120
第十一节 离散和采样系统中部分状态反馈次最优控制器的计算	122
第十二节 最小方差控制器的计算	125
第六章 控制系统的模型辨识	128
第一节 计算随机数列的统计特性	128
第二节 根据过渡过程响应确定传递函数	131
第三节 最小二乘估计辨识系统模型	136
第四节 广义最小二乘估计辨识系统模型	141
附录一 程序清单	147
1. 控制系统 CAD 常用算法程序	
1.1 求解线性方程组 (LINEQ)	147
1.2 代数多项式求根 (ROOT)	150
1.3 求解常微分方程 (DIFEQ)	154
1.4 矩阵的基本运算 (MATOP)	157
1.5 矩阵求逆 (INVMAT)	162
1.6 矩阵行列式的计算 (MATDET)	164
1.7 矩阵特征值和特征向量的计算 (EIGEN)	166
1.8 多项式的基本运算 (POLY)	173
1.9 多项式矩阵相乘 (POLMT)	175
1.10 多项式矩阵求逆 (POLINV)	177
1.11 连续李雅普诺夫方程的求解 (LYAP)	180
1.12 离散李雅普诺夫方程的求解 (LYAPD)	184
1.13 计算转移矩阵 (TRAMAT)	189
2. 控制系统仿真	
2.1 线性连续系统仿真 (RESP)	192
2.2 非线性连续系统仿真 (SIMCON)	198
2.3 离散系统仿真 (SIMDIS)	211
2.4 采样系统仿真 (SIMSAM)	214

2.5	随机连续系统仿真 (SIMRAC).....	224
2.6	随机采样系统仿真 (SIMRAD).....	232
3.	控制系统模型的相互转换	
3.1	化连续状态方程为离散状态方程 (DISAB).....	244
3.2	化离散状态方程为连续状态方程 (CONFIG)	247
3.3	双线性变换 (BILIN)	252
3.4	化状态方程为传递函数 (ABTRAN)	255
3.5	化状态方程为能控标准形 (ABCONT)	260
3.6	化状态方程为能观标准形 (ABOBS)	363
4.	应用古典控制理论分析和设计系统	
4.1	Routh 稳定判据 (ROUTH)	266
4.2	传递函数的部分分式展开 (EXPAND)	269
4.3	计算对数频率特性 (BODE)	278
4.4	计算根轨迹 (LOCU, LOCU1)	282
4.5	计算校正传递函数 (CASCAD)	297
4.6	利用长除法求 z 反变换 (ZTFINV)	303
5.	应用现代控制理论分析和设计系统	
5.1	按极点配置设计控制规律 (CPOLAS)	305
5.2	按极点配置设计观测器 (OPOLAS)	310
5.3	连续系统线性二次型最优控制规律的计算 (COTLQ)	315
5.4	离散和采样系统中线性二次型最优控制规律的计算 (SAMLQ)	323
5.5	离散和采样系统中最优状态估计器的计算 (KALM).....	338
5.6	连续系统二次型性能指标函数的计算 (CONLOS).....	347
5.7	离散和采样系统二次型性能指标函数的计算 (SAMLOS).....	352
5.8	随机连续控制系统中平均性能指标函数的计算 (RACLOS).....	371
5.9	随机离散和采样系统中平均性能指标函数的计算 (RADLOS).....	374
5.10	离散和采样系统中二次型性能函数及其 对控制器参数灵敏度的计算 (DLOSD)	384
5.11	离散和采样系统中部分状态反馈次最优控制器的计算 (SUBOPD).....	392
5.12	最小方差控制器的计算 (MINVAR)	401
6.	控制系统的模型辨识	
6.1	计算随机数列的统计特性 (RANDAT)	405
6.2	根据过渡过程响应确定传递函数 (TRANS)	407
6.3	最小二乘估计辨识系统模型 (LS)	415
6.4	广义最小二乘估计辨识系统模型 (GLSB)	421
附录二	BASIC-11 语言介绍	425
参考文献		430

前 言

为了普及和推广控制系统 CAD 技术在教学、科研及生产中的应用,本书汇编了近五十个实用程序,共分六章,它们分别包括以下内容:(1)控制系统 CAD 常用基础算法程序;(2)控制系统的仿真;(3)控制系统模型的相互转换;(4)应用古典控制理论分析和设计控制系统;(5)利用现代控制理论分析和设计控制系统;(6)控制系统的模型辨识。这些程序为控制系统的分析、设计、建模和仿真提供了方便的工具,为控制理论到实际应用之间架起了一座桥梁。这里给出的全套程序已在一些高等学校和研究部门获得应用,并在它们的教学和科研工作中发挥了重要作用。该应用软件已于 85 年通过了鉴定,并在 85 年召开的全国微机应用会议上被评为一等奖。

该应用软件全部采用 BASIC 语言编制,在 PDP-11/23 计算机上调试通过。同时全部程序也已移植到了 Dual 68000 微机系统及 IBM-PC 机上,本书给出的程序则为 PDP-11/23 机上的版本。每个程序均为独立的完整程序,它采用了统一的输入输出格式,即每个程序均提供有两种输入格式——键盘输入和文件输入,三种输出格式——终端屏幕输出、打印机输出和文件输出,从而使用比较方便和灵活。当需要运行程序时,首先使计算机进入 BASIC 状态,再将所要运行的程序从磁盘调入主机,然后启动该程序并回答相应的输入输出设备名。具体询问过程如下(下面划线部分表示由用户输入的内容):

RUN↵

问: INPUT FILE NAME

答: TT↵ 由键盘输入数据

xxx.DAT↵ 从磁盘文件中读取数据 (xxx 表示文件名)

问: OUTPUT FILE NAME

答: TT↵ 终端屏幕输出

LP↵ 打印机输出

xxx.DAT↵ 输出以该文件名存入磁盘

当用键盘输入数据时,程序采用问答式工作方式,用户根据屏幕上给出的提示输入数据。当用文件方式输入时,须首先建立数据文件,数据文件的格式和顺序一般均与键盘输入时相同。由于以上操作过程对于每个程序均是相同的,因此这里给予统一说明,而不再在以后的每个程序中加以说明。

本书每一节介绍一个独立的程序,其中包括程序的主要功能、特点、算法简介、程序说明及计算举例。重点是介绍程序的使用,对于所采用的算法也作了扼要的介绍,详细的说明及推导过程可参考书后所列的参考文献,其中参考文献[1]是本书的主要参考资料,若未加特别说明,算法的详细说明均参考[1],同时每个程序均给出了必要的计算举例,以便于读者使用和验证程序。所有程序清单均作为附录统一列在书后。为了便于参考,附录中还对程序所采用的 BASIC-11 语言作了简单的介绍。

本书第一章由叶榛执笔，第二章由丁冬花执笔，第三章由钱宗华执笔，第四章由袁曾任执笔，第五章由孙增圻执笔，第六章由张毓凯执笔。

参加本书程序编制工作的除以上执笔者外，何克忠老师及部分大学生和研究生也参加了程序的编制工作。尤其是计九三班的学生为此作出了较大的贡献。同时，在该应用程序的研制过程中，得到了航天部二院二十三所、航天部三院三部、航空部 618 所及航天部 302 所等使用单位的大力支持和帮助。在此，向以上提到的各位一并表示感谢。

由于编者水平所限，书中肯定存在不少缺点和错误，欢迎广大读者批评指正。

编 者

一九八六年一月

第一章 控制系统 CAD 常用算法程序

第一节 求解线性方程组

一、概述

本程序是用来求解线性方程组，采用了高斯列主元素消去法。本程序考虑了唯一解、无解及无穷解三种情况，当方程组有无穷解时，可以求出其基础解系。

本程序适用于方程组的维数不大于 20，若超过时可修改定维语句。

二、算法简介

本程序采用了高斯列主元素消去法。其基本思想是：依次按列选主元素，然后使之变换到主元位置，进行消元计算，使增广矩阵的下三角矩阵全为零，有解时回代求解。

三、程序说明

现将程序使用中遇到的主要符号说明如下：

N——未知数个数（本程序中规定 ≤ 20 ）。

B(N,N+1)——增广矩阵。

X(N)——方程的解。

RANK——增广矩阵的秩。

C——表示三种解的形式。

C=1 无解

C=2 有无穷组解

C=3 有唯一解

四、计算举例

例 1 解线性方程组
$$\begin{cases} 2x_1 - x_2 = 0 \\ 3x_1 + x_2 = 2 \end{cases}$$

依次输入 N=2 及增广矩阵
$$B = \begin{bmatrix} 2 & -1 & 0 \\ 3 & +1 & 2 \end{bmatrix}$$

运算后在屏幕上显示有唯一解的信息，在输出文件上给出结果：

x(1)=0.4

x(2)=0.8

例 2 解线性方程组
$$\begin{cases} 2x_1 - x_2 = 1 \\ 4x_1 - 2x_2 = 2 \end{cases}$$

其中

$$N=2 \quad B=\begin{bmatrix} 2 & -1 & 1 \\ 4 & -2 & 2 \end{bmatrix}$$

运算后在屏幕上显示有无穷解的信息，并在输出文件上给出方程的基础解系：

$$x(1) = 0.5 + 0.5x(2)$$

例 3 解线性方程组
$$\begin{cases} 2x_1 + x_2 = 0 \\ 2x_1 + x_2 = 1 \end{cases}$$

其中

$$N=2 \quad B=\begin{bmatrix} 2 & 1 & 0 \\ 2 & 1 & 1 \end{bmatrix}$$

运算后在屏幕上显示无解的信息。

五、程序清单

见附录 1.1。该程序由阎保民编制。

第二节 代数多项式求根

一、概述

本程序采用劈因子法求实系数多项式方程

$$a_0s^n + a_1s^{n-1} + \dots + a_{n-1}s + a_n = 0$$

的全部实、复根。这种方法的特点是不需要复数运算，当初值与真根相近时，收敛速度较快，当根以递增方向（指绝对值）求解时，解的精度较高。

本程序在对自动控制系统各种典型的和特殊的系统所对应的低阶的、高阶的特征方程的多次运算中，总结出了一些当迭代失败时的处理方法，并在程序中予以实现。

二、算法简介

设 $\omega_0(x) = x^2 + p_0x + q_0 = 0$ 是一元 n 次方程

$$f(x) = \sum_{i=0}^n a_i x^{n-i} = a_0 x^n + a_1 x^{n-1} + \dots + a_{n-1}x + a_n = 0 \quad (2.1)$$

的一个近似二次因式。用 $\omega_0(x)$ 除 $f(x)$ 得商

$$Q(x) = \sum_{i=0}^{n-2} b_i x^{n-i-2}$$

和余式

$$R(x) = r_0x + r_1$$

于是有

$$f(x) = \omega_0(x)Q(x) + R(x) \quad (2.2)$$

余式 $R(x)$ 中的系数 r_0 和 r_1 由 $\omega_0(x)$ 的系数 p_0 和 q_0 所决定：

$$\begin{cases} r_0 = r_0(p_0, q_0) \\ r_1 = r_1(p_0, q_0) \end{cases} \quad (2.3)$$

若能求得 $\Delta p, \Delta q$ 使得

$$\begin{cases} r_0(p_0 + \Delta p, q_0 + \Delta q) = 0 \\ r_1(p_0 + \Delta p, q_0 + \Delta q) = 0 \end{cases} \quad (2.4)$$

成立, 则求得了 $f(x)$ 的一个二次因式, 从而可以得到 $f(x)$ 的两个根 x_1, x_2 (求根算法从略)。将 $f(x)$ 除以因式 $(x - x_1)(x - x_2)$ 降阶后重复上面的过程。

三、使用说明

现将程序使用中遇到的主要符号说明如下:

N —— 方程的阶数

$A_9(I)$ —— 方程的系数。

$D1$ —— 由用户确定的精度。

$p(1)$ —— 各因子的初始值 p_0 。

$q(1)$ —— 各因子的初始值 q_0 。

1. 本程序由终端依次输入:

$N = ?$ n 各因子初始值 p_0, q_0 可输入 0, 1 或其它任意值

$p(1) = ?$ x x

$q(1) = ?$ x x

⋮

THE EXPECTED ACUURACY IS: ? x x

输入精度要求

2. 若计算成功, 则显示计算结果, 并将以上所有信息写入用户给出的输出文件, 然后显示:

RESTART OR CONTINUE OR AC.IMPROVE OR STOP? i.e. 10 or 0 or -1 or -10?

用户可输入上面给出的任一数字, 表示希望继续运算的方式, 其中:

10 —— 重新输入所有数据。

0 —— 重新输入 p, q 。

-1 —— 改进精度, 重新输入运算精度。

-10 —— 停止、退出程序。

3. 若迭代失败, 程序将分析失败原因, 并输出下次劈因子时 p_0, q_0 的初值参考值, 此时用户只要根据这些值的大小将它们输入即可。然后再输入一次期望精度值, 此次若迭代成功则转向“2”否则“3” (这时应适当降低运算精度)。

四、计算举例

例 1 求方程 $x^4 + 2x^2 + 1 = 0$ 的根。

依次输入 $N = 4, a_0 = 1, a_1 = 0, a_2 = 2, a_3 = 0, a_4 = 0, N_1 = 0.00001$;

选取 $p(1) = 500, q(1) = 300$ 。

运行后输出结果为

-3.34818E-04+j 0.999697, -3.34818E-04-j 0.999697,
3.34818E-04+j 1.0003, 3.34818E-04-j1.0003。

此结果满足要求。若希望得到更准确的结果，根据上述运算结果重新运行程序，并选 $p(1)=0$, $q(1)=1$, (可同时提高运算精度)，则结果为 $0+j$, $0-j$, $0+j$, $0-j$ 。

五、程序清单

见附录 1.2。该程序由王劲编制。

第三节 求解常微分方程

一、概述

本程序采用变步长龙格-库塔法。应用了关于步长的折半和加倍措施，以提高运算精度。用户可根据需要选择计算精度和计算步长。

二、算法简介

对于一阶常微分方程组的初值问题

$$\begin{cases} Y' = F(T, Y) \\ Y(T_0) = Y_0 \end{cases}$$

四阶龙格-库塔法的求解公式为

$$\begin{cases} K_1 = hF(T_i, Y_i) & (i = 0, 1, 2, \dots, n) \\ K_2 = hF\left(T_i + \frac{h}{2}, Y_i + \frac{K_1}{2}\right) \\ K_3 = hF\left(T_i + \frac{h}{2}, Y_i + \frac{K_2}{2}\right) \\ K_4 = hF\left(T_i + h, Y_i + K_3\right) \end{cases} \quad (3.1)$$

$$Y_{i+1} = Y_i + \frac{1}{6}(K_1 + 2K_2 + 2K_3 + K_4) \quad (3.2)$$

变步长定点输出的算法请见所列参考文献[2]。

三、程序说明

现将程序使用中遇到的符号说明如下：

N——方程组中方程的个数。

T——自变量 T 的初值 T_0 。

T_1 ——T 的最大值。

E_1 ——允许误差。

H——初始步长。

Z(N) —— 方程的初值。

1. 用户根据对不同的一阶微分方程组的求解要求, 应在本程序中修改 1310—1330 句中 F(N) 的内容。对不同的一阶微分方程组应改为不同的内容, 根据方程组中方程的个数 N, 确定其中 F(N) 的个数。

2. 原方程左边 “ $y_i =$ ” 或 “ $\frac{dy_i}{dT} =$ ” 改写为 “F(i) =” 方程右边的自变量代之以 T, 因变量代之以 $y_i (i = 1, 2, \dots, n)$

四、计算举例

例 给定初值问题

$$\begin{cases} \frac{dy_1}{dx} = \frac{1}{y_2} \\ \frac{dy_2}{dx} = -\frac{1}{y_1} \end{cases}$$
$$y_1(0) = y_2(0) = 1$$

1. 修改程序 1310—1330 句

1310 F(1) = 1/Y(2)

1320 F(2) = -1/Y(1)

1330 RETURN

2. 取 $N = 2$, $T = 0$, $T_1 = 1$, $E4 = 1E-5$, $H = 0.1$, $Z(1) = 1$, $Z(2) = 1$, 并依次输入这些参数。

3. 输出计算结果为

T	Y ₁	Y ₂
T = 0	1.10517	•904837
T = .1	1.2214	•818731
T = .2	1.34986	•740818
T = .3	1.49182	•67032
T = .4	1.64872	•606531
T = .5	1.82212	•548812
T = .6	2.01375	•496585
T = .7	2.22554	•449329
T = .8	2.4596	•40657
T = .9	2.71828	•367879
T = 1	3.00417	•332871

五、程序清单

见附录 1.3。该程序由丁冬花编制。

第四节 矩阵的基本运算

一、概述

本程序可对矩阵进行下列八种基本运算：

1. 两个矩阵相加，
2. 两个矩阵相减，
3. 两个矩阵相乘，
4. 求一个矩阵的转置矩阵，
5. 一个矩阵和另一个矩阵的转置矩阵相乘，
6. 一个矩阵的转置矩阵和另一个矩阵相乘，
7. 把一个矩阵对称化，
8. 求一个矩阵的逆矩阵。

程序中矩阵的最大维数定为 40×40 ，超过时，可修改程序中标号为 20 的定维语句。

二、算法简介

矩阵求逆算法见本章第五节。其它算法比较简单，下面只列出这八种基本运算的符号表示：

- | | |
|----------------|-------------------------------|
| 1. $C = A + B$ | 5. $C = AB^T$ |
| 2. $C = A - B$ | 6. $C = A^T B$ |
| 3. $C = AB$ | 7. $C = \frac{1}{2}(A + A^T)$ |
| 4. $C = A^T$ | 8. $C = A^{-1}$ |

三、程序说明

1. 程序中使用的主要变量说明如下：

$A(N_1, M_1)$ —— 第一个输入矩阵。

$B(N_2, M_2)$ —— 第二个输入矩阵。

$C(N, M)$ —— 运算后的结果矩阵。

2. 使用说明

(1) 当程序中提问

WHICH OPERATION DO YOU WANT?

时，回答下面八个符号之一，它分别代表需要做的八个基本运算。

- ① $\begin{array}{c} + \\ \swarrow \end{array}$ —— 表示要做两个矩阵相加。
- ② $\begin{array}{c} - \\ \swarrow \end{array}$ —— 表示要做两个矩阵相减。
- ③ $\begin{array}{c} * \\ \swarrow \end{array}$ —— 表示要做两个矩阵相乘。
- ④ $\text{TR} \begin{array}{c} \swarrow \end{array}$ —— 表示求一个矩阵的转置矩阵。
- ⑤ $* \text{TR} \begin{array}{c} \swarrow \end{array}$ —— 表示一个矩阵和另一个矩阵的转置矩阵相乘。

⑥ $TR * \swarrow$ ——表示一个矩阵的转置矩阵和另一个矩阵相乘。

⑦ $+TR \swarrow$ ——表示把一个矩阵对称化。

⑧ $INV \swarrow$ ——表示求一个矩阵的逆矩阵。

(2) 输入参数的次序为第一个输入矩阵的行、列、元素值。如有第二个输入矩阵时，次序也相同，但两个矩阵运算之间对行、列数有一定要求时，第二个输入矩阵的行或列值就不再需要输入了。如两个矩阵的加、减，必须相应的行、列值相等，则第二个矩阵输入时就不再要求行、列值，直接输入元素值即可。这对一个矩阵的情况也适用，如矩阵求逆，一定是方阵，则只要输入矩阵阶数一个值即可输入元素值。

若输入参数为文件输入时，数据文件的建立都不包含矩阵的行、列值，只要元素值。元素值的次序为第一输入矩阵，第二输入矩阵……行、列值根据程序提问，由键盘即时输入。

(3) 当做完一种运算后，程序问

DO YOU WANT ANY OPERATION MORE?

如果不再需要进行运算，可回答

$N \swarrow$ ，程序就退出运行，回到 BASIC 状态；如果还需要进行运算，可回答

$Y \swarrow$ ，程序就回到 (1) 中的提问，重复进行。

在重复运算时，不再重新选择输入输出方式，而沿用原来的人、出方式，这在文件输入时，数据文件的建立要注意这一点。若每次重复运算时，一定要重新选择输入输出方式，则可将程序中标号为 240 语句中的转向标号从 110 改为 30 即可。

四、计算举例

按照程序提问要求输入各参数后，程序输出结果如下：

WHICH OPERATION DO YOU WANT?

$* \swarrow$

INPUT OF MATRIX A(2, 4)

1 0 3 -1

2 1 0 2

INPUT OF MATRIX B(4, 3)

4 1 0

- 1 1 3

2 0 1

1 3 4

OUTPUT OF MATRIX C(2, 3)

9 - 2 - 1

9 9 11

DO YOU WANT ANY OPERATION MORE?

$Y \swarrow$

WHICH OPERATION DO YOU WANT?

$* TR \swarrow$

INPUT OF MATRIX A(4, 1)

4

-1

2

1

INPUT OF MATRIX B(4, 1)

1

1

0

2

OUTPUT OF MATRIX C(4, 4)

4	4	0	8
---	---	---	---

-1	-1	0	-2
----	----	---	----

2	2	0	4
---	---	---	---

1	1	0	2
---	---	---	---

DO YOU WANT ANY OPERATION MORE?

Y↵

WHICH OPERATION DO YOU WANT?

TR↵

INPUT OF MATRIX A(2, 4)

1	2	3	4
---	---	---	---

5	6	7	8
---	---	---	---

OUTPUT OF MATRIX C(4, 2)

1	5
---	---

2	6
---	---

3	7
---	---

4	8
---	---

DO YOU WANT ANY OPERATION MORE?

N↵

五、程序清单

见附录 1.4。本程序由钱宗华编制。

第五节 矩阵求逆

一、概述

本程序采用列主元消去法求给定满秩矩阵的逆。当给定矩阵为奇异时，程序输出“SINGULAR”的信息并停止计算。