

FORTAN 77

结构化程序设计

孙家启 主编

机械工业出版社

2

1/2

TP312
STQ/2

FORTRAN77结构化程序设计

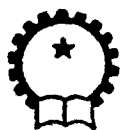
主编 孙家启

编者（以姓氏笔画为序）

马晓安 尹淑勤 张延炬 孙家启

宋顺林 吴国凤 姜占华 姜德森

柴保桂 黄新蓉



机械工业出版社

内 容 简 介

本书是为高等院校机械、电气类专业学习和使用电子计算机FORTRAN77语言及编程技术而编写的教学用书。

本书主要内容是根据FORTRAN语言发展新动向,除介绍FORTRAN77语言基本语法规则、逻辑结构、程序设计技巧外,还重点介绍了结构化程序设计新思想及方法。本书提供了国内常用几种机型的FORTRAN77语言上机操作指南以及列举大量应用程序举例。

本书除可作为高等院校机电类专业教材外,也可作为机械工程师、电气工程师自学读本,以及成人教育的教材或参考书。

35262 / 10

· FORTRAN77结构化程序设计

主 编 孙家启

*

责任编辑 李 敬

封面设计 刘 代

*

机械工业出版社出版(北京阜成门外百万庄南里一号)

(北京市书刊出版业营业许可证出字第117号)

通县建新印刷厂印刷

新华书店北京发行所发行·新华书店经售

*

开本 787×1092 1/16·印张 22 ·字数 537 千字

1988年7月北京第一版·1988年7月北京第一次印刷

印数 00,001—11500·定价: 5.30元

*

ISBN 7-111-00820-0/TP·57

前 言

本书系按照1984年原机械工业部教育局委托合肥工业大学召开的部属院校《计算机与算法语言》协作会上所制订的机电类专业《FORTRAN77结构化程序设计教学大纲》编写而成。

FORTRAN 语言是目前最流行的程序设计语言之一；FORTRAN77 是它的最新版本。目前，国内大多数计算机都已配置了FORTRAN77编译系统，国外的许多科学技术书籍中，都有用FORTRAN77编写的程序。FORTRAN77在字符处理等方面增加了多种功能，这就使得它不仅方便地进行科学计算，而且可以更广泛地应用到非数值计算的各个领域。

结构化程序设计是国际上近年来兴起的一种程序设计新方法。它要求按一定的结构形式来设计和编写程序，而不能随心所欲地编写程序。它使程序有良好的结构，使之易编写、易阅读、易修改和调试，以提高程序设计和维护工作的效率。本书考虑到国际上发展的趋势和国内的需要而增设了“FORTRAN77结构化设计与算法”一章内容，目的是：强化结构化思想，提倡结构化程序设计这一新方法。因为今天计算机的应用范围最广的已不是科学计算，而是数据处理，所以书中强调了文件系统的使用和字符处理功能等。

本书编写过程中力求做到概念清楚、深入浅出。书中所列举的程序举例都已在机器上调试通过，可供读者在工作和学习时参考。

本书可作为高等院校本科生、研究生及教师的教材和教学参考书，函授大学、业余大学、广播电视大学的教学用书，亦可供程序设计人员、科研人员、工程技术人员、管理人员自学。

合肥工业大学孙家启编写本书第一章、第十三章和附录三、四、五、六、七、十一；洛阳工学院柴保桂编写第二章、第四章和附录一、九；吉林工业大学尹淑勤编写第三章、第十五章；合肥工业大学黄新蓉编写第四章；合肥工业大学吴国凤编写第六章；江苏工学院宋顺林编写第七章；北京机械工业管理学院张延炬编写第八章、第九章和附录二；附录八，吉林工业大学姜占华编写第十章、第十一章和附录十；甘肃工业大学马晓安编写第十二章；哈尔滨电工学院姜德森编写第十四章。全书由孙家启副教授主编，张奠成教授、谢培均副教授主审。

在编写过程中，自始至终得到国家机械工业委员会教育局、合肥工业大学及兄弟院校大力支持，在此一并致谢。

编 者

1987年12月

目 录

第一章 概 论	1
§1-1 计算机系统的组成	1
§-2 FORTRAN77 语言程序设计	7
习 题	
第二章 FORTRAN77 程序的基本成份	16
§2-1 FORTRAN77 的字符集	16
§2-2 常 数	17
§2-3 变 量	19
§2-4 内部函数	21
§2-5 一维数组	22
§2-6 算术运算符、算术表达式及其类型	24
习 题	
第三章 简单FORTRAN77程序设计	30
§3-1 算术赋值语句	30
§3-2 表控输入/输出语句	31
§3-3 数据语句(DATA 语句)	36
§3-4 参数说明语句(PARAMETER 语句)	37
§3-5 停止语句(STOP语句)	37
§3-6 暂停语句(PAUSE 语句)	38
§3-7 结束语句(END 语句)	38
§3-8 应用程序举例	38
习 题	
第四章 逻辑运算和判定控制结构程序设计	41
§4-1 逻辑 IF 语句	41
§4-2 多重条件语句	46
§4-3 块IF语句	50
§4-4 应用程序举例	56
习 题	
第五章 循环结构程序设计	64
§5-1 条件控制的循环	64
§5-2 DO 循环语句	65
§5-3 继续语句(CONTINUE 语句)	68
§5-4 多重循环	69
§5-5 应用程序举例	72
习 题	
第六章 结构化程序设计与算法	77
§6-1 概 述	77
§6-2 结构程序设计的基本原则及算法的提出	78
§6-3 算法及算法的设计方法	80
§6-4 结构化程序设计的基本控制结构	84

§6-5 影响结构化程序设计的处理	88
习 题	
第七章 有格式输入/输出	91
§7-1 输入/输出概念	91
§7-2 有格式输入/输出概述	92
§7-3 格式说明及其应用	95
§7-4 格式说明和读/写语句的相互作用	105
§7-5 应用程序举例	108
习 题	
第八章 多维数组	114
§8-1 多维数组概述	114
§8-2 数组元素在内存中的存贮顺序	115
§8-3 数组的输入/输出	117
§8-4 数组运算中的常见操作	124
§8-5 应用程序举例	130
习 题	
第九章 双精度和复数运算	140
§9-1 双精度运算	140
§9-2 复数运算	142
§9-3 几种数值型数据类型的比较	145
§9-4 应用程序举例	150
习 题	
第十章 过 程	155
§10-1 语句函数	155
§10-2 函数子程序	159
§10-3 子例程子程序	167
§10-4 可调数组	168
§10-5 外部语句(EXTERNAL语句)	171
§10-6 内部语句(INTRINSIC语句)	173
§10-7 保留定义状态语句(SAVE语句)	174
§10-8 应用程序举例	175
习 题	
第十一章 数据的公用结合	181
§11-1 等价语句(EQUIVALENCE语句)	181
§11-2 公用语句(COMMON语句)	183
§11-3 数据块子程序	188
§11-4 应用程序举例	189
习 题	
第十二章 字符处理	204
§12-1 字符型常数	204
§12-2 字符型变量和字符赋值语句	204
§12-3 字符表达式	207

§12-4 子字符串	208
§12-5 字符关系表达式	210
§12-6 字符型数据的输入/输出	211
§12-7 字符函数	215
§12-8 字符格式说明	219
§12-9 应用程序举例	221
习 题	
第十三章 数据文件	230
§13-1 数据文件的概念	230
§13-2 数据文件的打开和关闭	232
§13-3 数据文件顺序存取	238
§13-4 数据文件直接存取	244
§13-5 输入/输出补充	248
§13-6 应用程序举例	251
习 题	
第十四章 其它的 FORTRAN 语句	267
§14-1 算术 IF 语句	267
§14-2 计算 GOTO 语句	269
§14-3 赋标号语句和赋值GOTO语句	270
§14-4 子程序的多重入口和选择返回点问题	272
§14-5 应用程序举例	276
习 题	
第十五章 常用 FORTRAN77 计算程序	285
§15-1 用拉格朗日插值法求函数值	285
§15-2 用辛普森变步长法求积分	286
§15-3 用缪里尔法求任意函数的零点	289
§15-4 求矩阵最大特征值及其相应特征向量的迭代法	295
§15-5 用列主元高斯消去法解线性方程组	297
§15-6 用龙格-库塔法解常微分方程组	300
附 录	
附录一 FORTRAN77与FORTRAN66的比较	305
附录二 FORTRAN77全集与子集的区别	306
附录三 FORTRAN77内部函数	309
附录四 FORTRAN77可执行语句和非执行语句表	311
附录五 FORTRAN77语句形式表	312
附录六 FORTRAN77程序单位内的语句顺序和注释行	314
附录七 SACH代码和 EBCDIC 代码对照表	314
附录八 IBM PC机FORTRAN77上机操作指南	317
附录九 DUAL/68000机FORTRAN77上机操作指南	326
附录十 PDP-11/23机FORTRAN77上机操作指南	333
附录十一 VAX-11/780机FORTRAN77上机操作指南	335

第一章 概 论

电子计算机是实现信息处理的自动机。它的出现和应用,给社会带来了深刻的变化,有力地推动了技术的飞速发展。从人造地球卫星的发射、航天飞机的飞行,到生产过程的自动控制、办公室自动化、家庭现代化等,都离不开电子计算机。今天,电子计算机已渗透到科学技术的各个领域以及社会生活的各个方面。可以毫不夸张地说,没有计算机就没有现代化。

在未来的社会里,每一个有知识的人都应该学会使用计算机。计算机知识应当成为新一代的知识分子知识结构和智能结构的一个重要组成部分。为了使用好计算机,必须首先了解计算机。

程序设计语言FORTRAN是科学计算中使用极为广泛的一种计算机语言。FORTRAN77是FORTRAN的最新发展。由于语言结构上的改进,FORTRAN77能较好地体现结构程序设计的原则,加以在一些语法规则上的放宽和用于文本处理方面的一些新功能的引入,它不仅在数值计算方面继续保持其重要的地位,而且在非数值计算方面也为人们所注目。

§1-1 计算机系统的组成

长期以来,电子计算机常被一种神秘的浓雾笼罩着,一旦人们拨开迷雾,便可清晰地看到它毕竟是人们长期生产实践和科学研究的产物,它所具有的功能,都是人类赋予的。为了帮助读者建立一个较为完整而又概括的总体概念,下面将介绍计算机系统的组成。

一、计算机的硬件组成

人们用算盘来解算题目是非常熟悉的,而用算盘计算题目,一般要用到纸、笔、算盘和使用这些工具的人。计算机要能自动地进行算题,也必须具备与其相似的几个基本装置。

首先,类似于算盘功能的运算部件,我们称之为运算器,它能对数字进行运算。

其次,类似于纸的功能的记忆装置,我们称之为存贮器,它能保存和记录原始数据、运算步骤以及运算的中间结果和最终结果。

再次,类似于笔的功能的输入输出装置,我们称之为输入和输出设备,它能 把原始数据、运算步骤等信息送到主机中去,而把运算结果显示出来。

最后,类似于人的大脑功能的控制部件,我们称之为控制器,它能使运算器自动地进行运算,并使各部分均能协调地有节奏地进行工作。

因此,计算机硬件由运算器,存贮器(指内存贮器)、控制器、输入设备和输出设备五大部分组成。除此之外,一般计算机还包括外存贮器等部件。我们把运算器、内存贮器和控制器这三部分统称为主机,而把运算器和控制器合起来称为中央处理机(Central Processing Unit),简称CPU。把输入设备、输出设备、外存贮器等统称为外部设备。各部分相互之间的联系可以用图1-1简单框图表示。图中实线表示控制信号,虚线表示反馈信号,各外部设备必须通过接口电路与主机联系。

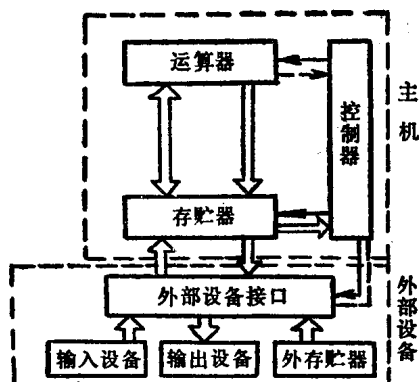


图 1-1

二、计算机的软件组成

我们知道，在用算盘算题时，除算盘本身外，还需要运算法则和计算步骤，并运用珠算口诀来进行。如果仅有算盘，而没有运算法则和解题步骤，就不能用算盘算题。因此，算盘本身就是硬件，而运算法则和解题步骤等就是算盘算题的软件。同样，计算机也是如此，如果只有计算机硬件，而不配备计算机软件，就不能高速自动地完成各种运算。有时就把除了软件外的计算机称为“裸机”。

实际上，利用计算机进行计算、控制或作其它工作时，除了裸机外，还需要各种各样用途的程序。因此，凡是用于一台计算机的各种各样的程序，统称为这台计算机的软件或程序系统。

计算机软件一般分为两大类：系统程序和应用程序。

1. 系统程序。

系统程序用来简化程序设计或使程序设计自动化，简化使用方法，提高计算机的使用效率，发挥和扩大计算机的功能和用途，使不了解计算机的人们也能使用比较复杂的计算机。它包括以下三类：

- (1) 服务程序，如诊断程序、排错程序、练习程序等。
- (2) 语言程序，如汇编程序、解释程序、编译程序等。
- (3) 管理程序或操作系统。

2. 应用程序。

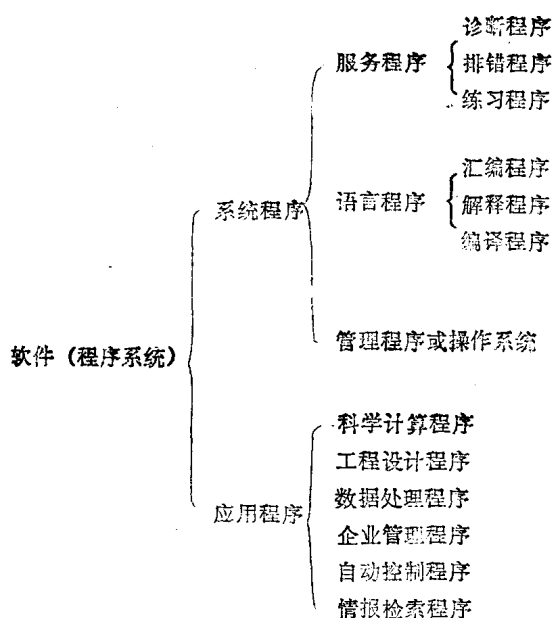
应用程序是用户为使用计算机解决某些问题而编制的程序。随着计算机的广泛应用和普及，应用程序种类日益繁多，例如：作科学和技术计算用的科学计算程序；作工程设计、结构设计、工艺设计、产品设计、建筑设计等用的工程设计程序；对试验数据、统计数据进行处理用的数据处理程序；对企业生产管理、质量管理、仓库管理等用的企业管理程序；对生产过程进行自动控制用的自动控制程序；对情报资料进行检索的情报检索程序等等。这些统称为应用程序。

计算机软件的分类可见表1-1。

三、系统软件概念

我们平时讲计算机，实际上它都分为硬件和软件两大部分。硬件是“硬梆梆”的计算机实体，它是构成整个机器的物质基础。但要计算机真正做事情，必须靠软件。软件是用某种语言编制的程序，它使计算机自己管好自己，以及让计算机跟外界联系起来，并完成各种指定

表1-1 计算机软件分类



的工作。因此,这台计算机软件越丰富,它懂得的语言种类也越多,管理自己也就越好,为外界服务也越优,则它的功能就越强。反之,若我们搞一台纯硬件的计算机,那只是毫无生气的“死铁”一堆了。

四、软件的基本概念

由于微处理器正在逐步取代数字逻辑设计,所以程序设计就成为系统设计师的主要工具。对于广大用户来讲,微型计算机的发展,已经将硬件知识简单到最低限度,比中学生装一架电视机还要方便。我们用户仅仅需要选购一台适用的、廉价的计算机,按照应用的需要,着眼于接口技术和软件编制上即可。

这里,我们先把一些基本知识再罗列起来,稍作说明,见图1-2。

1. 语言(language)。

(1) 机器语言 (machine language)

计算机内部只认识“1”和“0”,即二进制数。这是计算机使用的语言,亦叫目标语言。由这种语言编写的程序毋需经过翻译就可被计算机直接执行,所以亦叫目标程序(object program)。

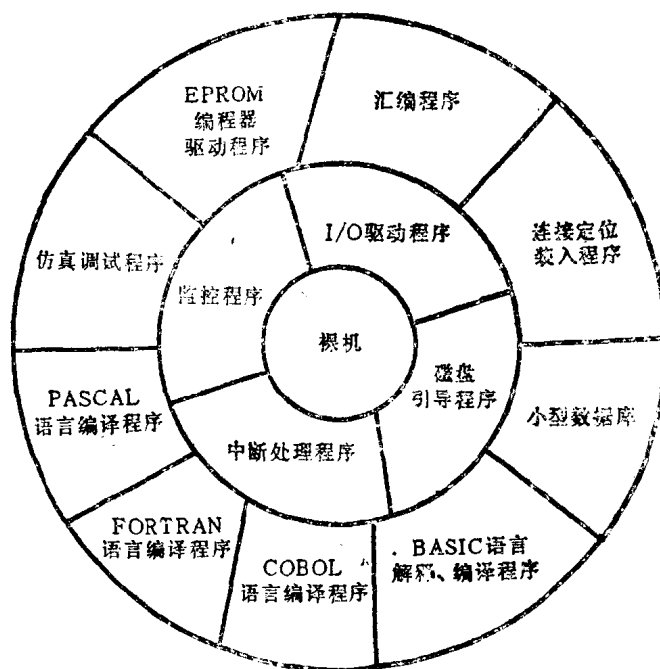


图1-2 微型机操作系统及其支持软件的组织结构

4

机器语言即计算机的指令系统。

用“1”和“0”来编制程序，编的人烦，看的人也吃不消，当然跟机器最“亲近”。

(2) 汇编语言 (assembly language)

它是针对某种计算机抽象出来的一种记忆符号。它还是面向机器的，属于低级语言的范畴。不过这样的符号还是容易理解和记忆，如 LD B, A 指令就是表示把 A 里的内容装到 B 里去。这里的 LD 是英文 LOAD 的缩写。意思是装载。用这样的语言来编制程序，就易编易懂。当然，计算机内部必有一本“词典”加以翻译，把汇编语言写的程序翻译成机器语言的指令序列，这一“翻译官”就叫汇编程序 (assembler)

但是汇编语言也有缺点：①记忆符号毕竟还有些抽象。以 Z80 CPU 的指令系统来说，就有158条这样抽象的符号，这会吓倒不少人的；②它能直接指挥计算机的内部结构，所以使用汇编语言还得比较了解该计算机的具体细节问题；③汇编语言对各个厂商的机器并不通用，如 Z80、8080、6800各搞一套。

尽管如此，用汇编语言毕竟可以直接指挥机器，程序较为紧凑，运行速度也快，还容易掌握，而且它成为学习计算机原理的入门知识。

(3) 高级语言

有没有另外一些语言，可以使对计算机“完全”不懂的人使用起来更为方便，也可以“完全”不关心计算机的内部结构，照样可以把计算机作为工具和手段来使用呢？回答是有的，且有许许多多种类语言供你选择使用。

最简单的就是 BASIC 语言，这是人机对话的一种语言。机器又是怎样会识别的呢？即计算机必须备有“一位 BASIC 语言的翻译官”。从常见的 IBM PC、APPLE-II 为例，这个“翻译官”就固化于机器的 ROM 中，因此用户使用极为方便，称之为“开箱即能使用的”机器。这个翻译过程是对逐条语句边解释边执行的，类似于“口译”，速度较慢。

计算机要为用户方便使用，接近于英语书写的各种高级语言应运而生，如适于科学技术运算用的 FORTRAN 语言，适于管理事务处理用的 COBOL 语言，号称“无所不包的天下第一号”的 PL/1 语言以及 ALGOL、PASCAL、C、Ada、FORTH、LISP、PROLOG/E、SPARKS/A 语言等。计算机懂不懂这种语言，问题仅仅在于有没有具备这种语言的“翻译官”——编译程序 (Compiler)。编译程序就是能把输入的源程序翻译成等价的目标程序，类似于“笔译”，耗费的存贮空间极大。

2. 指令 (instruction)

顾名思义就是指和调动计算机执行某个具体动作的“指示”和“命令”，这犹如一个指挥员 (CPU) 对整个队伍 (计算机) 喊出的口令 (指令)。它规定了计算机的操作种类、操作数的值或其地址。

指令系统 (instruction set) 就是该指挥员能够使之执行的各种不同类型指令的总和。口令的花色品种越多，质量越高，则我们可以讲该机的指令系统很丰富。

3. 程序 (program)

为了完成某件事情，往往需要许多条指令 (包括重复使用) 排成有机的序列，逐条执行，最终来完成。这样，按你的要求有机编排出来的指令序列，组成一段程序。程序可以用机器语言 (汇编语言) 来编写，也可以用程序设计语言来编写，使计算机实现所预期的目的

(如解某一算题或控制某一过程)。

为了充分发挥计算机效能的各种程序的总称叫软件。包括监控程序、操作系统 各种程序设计语言、编译程序, 以及检查和诊断程序等。也有人认为, 广义地讲应把用户为了利用计算机解决某些问题而编制的程序(称为用户程序)也包括在内。

如果说硬件是“基础”, 则软件是“灵魂”。微型计算机的硬件急速发展, 且日臻完善, 价格暴跌, 形成了一股“硅冲击波”。尽管大力推动了软件的相应发展, 但从某种意义上来说, 软件技术成为整个微型机技术发展“瓶颈”, 也称之为“软件危机”。

我们不妨把软件系统分成两大类: 系统软件和应用软件。

(1) 系统软件

系统软件有: 引导程序、监控程序、操作系统、文本编辑程序、连接程序、定位程序、汇编程序、宏汇编程序、交叉汇编程序、反汇编程序、调整程序、测试诊断程序等等。

我们把高级语言的各种编译程序(还包括 BASIC 的解释程序)也囊括在内。

此外, 还有程序库(浮点运算库、汇编程序库、子程序库以及各种标准程序等)。

(2) 应用软件

应用软件有: 在线仿真器软件包, EPROM 编程的工作软件, A/D 变换程序, D/A 变换程序, 模拟程序等等。总之, 这是专门为了解决某个应用领域里的一类或几类实际问题而编写的程序。它对某一类问题有较强的通用性, 供用户选用。

五、操作系统(Operating system, 简称OS)

操作系统是整个计算机系统资源的“总管家”, 它是负责管理资源(包括硬件和软件)的程序。注意, 它本身也是软件。

1. 操作系统的位置

操作系统在计算机系统的位置如图 1-2 所示。从图上可以看到, 操作系统成为人和计算机裸机之间的一个接口程序, 与其说人在接触和使用计算机硬件, 还不如说人仅仅是接触和使用操作系统(软件), 或者说是扩充了功能的(包了一层操作系统)计算机系统。

它提供给用户的是一台操作简便、安全可靠、能协助用户经济有效地解决问题的计算机。操作系统除对计算机的资源管理和调度外, 还给用户建立了一些操作命令语言。人们仅需按照它的使用手册操作, 而使人对裸机的直接干预(对硬件的具体操作)减少到最低限度。此外, 在操作系统的支持下, 还可以对各种系统软件(包括各种程序设计语言)和应用软件方便地使用, 简化了用户编写、调试、修改与运行程序的手续。从图 1-2 来看, 犹如在操作系统这一层外面可以再“贴”上各种不同的程序。这样计算机的功能又扩充了, 不同的用户根据不同的需要各自使用“贴”在外面的软件, 而都能够通过操作系统与整个计算机系统打交道。

从图 1-2 中还可以看到, 往往紧“裹”在裸机最内层的另外给它一个名称, 即监控程序, 它属于操作系统的一部分, 当然功能差得多。有些廉价低档机(如单板机)只有监控程序, 而没有操作系统, 因为它们不必带操作系统之大驾, 由于监控程序比较简单, 又象是操作系统的“雏型”。请参阅有关专著加以剖析。

广义地说, 在操作系统监控程序支持下运行的所有子系统(如操作命令处理程序, 程序语言处理程序, 系统服务程序, 以及用户开发的用来针对具体问题进行管理的各种处理程序等等)都可成为操作系统的一部分。

2. 操作系统的功能

操作系统的主要功能就在于：①监视计算机系统中的各种资源；②决定谁获得这种资源，何时获得？获得多少？③分配这种资源；④当资源使用完后，收回各种资源。简言之，它负责对整个资源的监督、调度、分配和管理，决定了对计算机资源的使用方法与策略。

计算机资源按其性质可归纳为四大类：处理机资源；存贮器资源；外部设备资源以及信息（程序和数据）资源。所以操作系统都设置有与这四类相应的程序管理模块。特别值得强调的是对信息的管理，操作系统中建立了文件系统。也可以说，操作系统的最重要的功能是对信息进行有效的管理。与信息管理有关的操作系统各模块往往称为文件系统。

操作系统一般都放置于外部存贮器里（如磁盘）。因此人们称之磁盘操作系统（Disk Operating System，简称 DOS）。这里有必要结合存贮器管理来谈一下怎样与计算机的内存和外存打交道，以及操作系统是如何调度的问题。

我们知道计算机的内存较之外存其容量小得多，一般微型机为 64k 字节。①在内存中必须有一些最基本的“常驻部队”，最起码总得有一个“迎接”外存与内存进出的所谓“磁盘引导程序”。在内存中常驻程序占用的空间称为常驻段（root segment）；②内存中还得考虑让外存的操作系统或其一部分“进驻”，依靠它来管理计算机；这样又得划出一部分存贮空间；③内存中总得给用户留出足够大的空间，让用户能够尽量多地存放一些信息。实在盛不下，怎么办？这就不得不借助外存贮器，这样就有了一个比实际内存容量大得多的虚拟存贮空间。我们可以这样来比喻，一个旅游胜地（内存）容量有限，必有它本身的管理部门（常驻程序）。现有多方面旅游团体（多用户、多任务，只是大量的游人）（用户程序）欲来此一游，他们又各有旅游社工作人员陪同和管理，不妨假定工作人员也是大量的（操作系统各个部分）。那么，只可能在总调度（操作系统核心）管理下使所有的人员分期分批来胜地一游，这就叫覆盖技术，就是使外存中的若干个数据块或程序块重叠地占用内存中的某一部分。这样一来，就可以在最大内存空间不允许装入整个程序时，也同样运行了整个程序。覆盖技术是普遍采用的基本技术之一。对于用户来说，它好象比原来的内存扩大很多很多倍了，我们把这种表面上扩大了存贮空间叫做虚拟存贮空间（Virtual Memory Space），这称为虚拟存贮技术。操作系统所能管理这一技术的能力是其重要特性之一。

CP/M（Control Program/Monitor）被誉为 8 位微型机通用的操作系统和“软件总线”。它适用于 Intel 公司的 8080、8085 和 Zilog 公司的 Z80 为 CPU 的微机系统。

CP/M 就是采用层次分明的模块结构，它便于移植、修改和开发。

六、计算机的工作原理

前面已述，计算机只有硬件，那末，它还只是具备了计算的性能。欲把这种可能性变成现实，还必须依靠软件的配合。这就是说，为了让计算机能按照人们的意图进行运算，就必须事先把计算方法和解题步骤翻译成机器能够理解的语言，例如用二进制代码形式的机器语言，来编制解题步骤，这就是编制程序的过程。接着，把编好的程序连同原始数据，通过输入设备存入计算机的存贮器中。然后，启动计算机，计算机便在程序控制下，按人的意图自动地进行操作，直至全部计算完毕后，通过输出设备送出结果。以上就是迄今为止，电子数字计算机所共同遵循的程序存贮和程序控制的原理。这种原理是 1945 年冯·诺依曼提出的，故又称为冯·诺依曼（John von Neumann）型计算机原理。

计算机只有同时具有硬件和软件，才能自动地、快速地工作，从而完成人们可要求的任务。

现在，我们以 $17+14\times 2$ 这道题为列，说明计算机的工作过程：

第一步：由输入设备将事先编制好的计算步骤（即计算程序），原始数据 17、14、2 输入到存储器存放起来。

第二步：启动计算机，在控制器的指挥下，计算机按计算程序自动地进行操作。

(1) 从存储器取出乘数 2，送到运算器中。

(2) 从存储器取出被乘数 14 到运算器，进行 14×2 的乘法操作，在运算器中求得中间结果 28。

(3) 将运算器中的中间结果 28 送到存储器暂存。

(4) 从存储器中取出被加数 17，送到运算器中。

(5) 从存储器中取出加数到运算器，进行 $17+28$ 的加法操作，在运算器中求得加法结果 45。

(6) 将运算器中的最后结果 45 存入存储器。

第三步：由输出设备将最后结果 45 打印在纸上。

第四步：停机。

§1-2 FORTRAN77语言程序设计

一、FORTRAN 语言的发展

FORTRAN是由FORmula TRANslation 中的前几个字母组合而成的，意为“公式翻译”。

FORTRAN 语言是世界上出现最早的一种高级计算机语言。它自 1954 年提出，1956 年在 IBM 704 计算机上实现以来，已经历了三十多年。在这三十多年当中，它几经修改和标准化，形成过几种不同的版本。如 1958 年的 FORTRAN II 和 1962 年的 FORTRAN IV，都是广为流行的。为了统一 FORTRAN 语言，美国标准化协会(American National Standard Institute，简称 ANSI)于 1964 年提出两个标准文本草案。它们是：

基本 FORTRAN (相当 FORTRAN II)

FORTRAN (相当 FORTRAN IV)

又在 1966 年公布了两个美国标准文本：

标准 FORTRAN(X3.9-1966)(大体相当于 FORTRAN IV)，称作 FORTRAN66。

标准基本 FORTRAN (X3.10-1966) (大体相当于 FORTRAN II)。

因 FORTRAN 语言在国际上广为使用，影响很大。国际标准化组织(International Standard Organization，简称 ISO)于 1972 年公布了完全的(一级)，中间的(二级)，基本的(三级) FORTRAN。其中，一级的相当于 FORTRAN IV，三级的相当于 FORTRAN II，并在一级与三级之间增加了一个二级的。

在其后的十多年中，不少的计算机厂家又对 FORTRAN 作了不少扩充、改进，增加了不少新的功能。1976 年美国标准化协会对 FORTRAN66 提出修改意见，于 1978 年公布了

新的美国国家标准, 即 ANSI X3.9-1978 FORTRAN。为同原来的老标准相区别, 特地把它称作“FORTRAN77”。由于在制订 FORTRAN77 时充分考虑到 FORTRAN66 应用的普遍性和众多的程序能继续使用, 因此 FORTRAN77 与 FORTRAN66 是兼容的。

尽管如此, FORTRAN77 与 FORTRAN66 相比, 还是作了重要的改进, 增加了新的功能, 在方便用户方面前进了一大步。FORTRAN77 的主要特点可概括为: 增加了字符型数据及其处理功能, 这使 FORTRAN77 语言也适应于非数值处理方面的应用。增加了块 IF 语句及块结构, 这使 FORTRAN77 在构成控制结构时可以避免 GOTO 语句, 也使 FORTRAN 语言更加适应结构化程序设计的需要; 丰富和发展了输入/输出语句的功能放宽了对不同类型的数据进行混合运算的限制; 增加了部分说明语句和内部函数; 丰富了处理功能, 也扩充了部分语句的功能等。

二、FORTRAN77 程序结构

FORTRAN77 源程序是块状结构, 它是由一个主程序块或者加上若干个子程序块组成的。子程序包括函数子程序和子例程子程序, 统称为过程子程序, 还有一个子程序叫数据块子程序。

程序块又称为程序单位, 程序单位由语句序列和任选的注解行的序列所组成。一个程序单位或者是一个主程序或者是一个子程序。每个程序单位中由若干行组成, 每行由若干列组成。程序单位中的行不超过 72 个字符的序列。在一行中的字符位置称为列, 并且按从左到右的顺序排列, 因此, 程序单位是由整个有序字符集组成。

程序单位中的行分为:

1. 注解行。注解行是在第 1 列有字母 C 或星号 * 或在第 1 列至第 72 列仅有空白符的任何行。第 1 列有 C 或 * 的注解行, 在第 2 列至第 72 列中可以包含计算机处理系统能够表示的任意字符。在注解行中一般允许使用小写字母和小写字。注解行可以出现在程序单位中的任何地方, 它不参与程序的执行, 仅为程序提供资料说明, 便于阅读程序。它可以被输入输出。

2. 始行。始行为不是注解行的任何行, 而且第 6 列必须是空白或数字 0。第 1 列至第 5 列可以含有语句标号或空白符。

3. 继续行。继续行为第 6 列含有除空白符或数字 0 以外的字符集中任意字符的行, 而且第 1 列至第 5 列必须是空白符。继续行是上一行的继续, 一个语句最多可有 19 个继续行。

4. 结束行。每个程序单位必须以 END 结束, END 语句仅写在第 7 列到第 72 列上。

总之每个 FORTRAN77 语句必须写在第 7 列至第 72 列上, 且一个语句不能多于 1320 个字符。任何语句的前面可以带有标号, 用来标识这个语句的名字和位置。语句标号的形式是 1 到 5 个数的序列。语句标号可以放在语句始行的第 1 列到第 5 列的任何地方, 标号的前 0 和空白无意义。在一个程序单位中, 同一语句标号不能给予一个以上的语句。

FORTRAN77 程序是最简单的“功能块结构”, 由各种功能块组合成更复杂的程序实体, 直到产生完整的 FORTRAN77 程序为止。对于读者来说, 直到现在还不知道标准的 FORTRAN77 程序是什么样, 为了介绍 FORTRAN77 的基本概念使读者了解它的结构, 我们介绍一个完整的 FORTRAN77 程序, 使读者先有一个感性认识。

读入三角形三个边长, 计算它的面积和三个角的大小, 见图 1-3。为了说明方便, 在程序的每行左边印上了行号, 这与实际程序无关。

下面给出了这个问题的完整的 FORTRAN77 程序，称它为可执行程序。这个程序由两个程序单位组成，一个程序单位是从第 1 行至第 17 行称为主程序。它以 PROGRAM 开始，程序名字是 TRIANG。另一个程序单位是从第 18 行至第 22 行，称为函数子程序，它以 FUNCTION 开始，函数子程序名字是 ANGVAL。

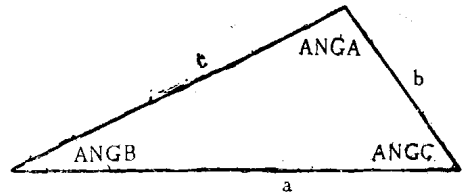


图 1-3

每一个程序单位由一系列语句组成，其中有些是执行型语句，有些则是非执行型语句。如：主程序的前几行是非执行型语句，从第 4 行开始到第 16 行是执行型语句。粗略地说非执行型语句是为向机器提供必要的信息，它不被编译成可执行的机器指令，而执行型语句被编译成可执行的机器指令。

程序为：

```

C 1   PROGRAM TRIANG
      2   LOGICAL FLAG
      3   READ*, A, B, C
      4   PRINT*, 'THE LENGTHS OF THE SIDES OF THE TRIAN
        GLE', A, B, C
      5   FLAG= (A+B) .GT. C. AND. (B+C) .GT. A. AND. (C+A) .G
        T. B
      6   IF (FLAG) THEN
      7   S=0.5*(A+B+C)
      8   AREA=SQRT(S*(S-A)*(S-B)*(S-C))
      9   ANGA=ANGVAL (B, C, A)
     10   ANGB=ANGVAL (C, A, B)
     11   ANGC=180- (ANGA+ANGB)
     12   PRINT*, 'AREA OF TRIANGLE IS', AREA,
     13   *      'ANGLES OF TRIANGLE ARE', ANGA, ANGB, AN
        GC
     14   ELSE
     15   PRINT *, 'INVLIID INPUT DATA'
     16   END IF
     17   END
     18   FUNCTION ANGVAL (X, Y, Z)
     19   PI=3.14159265
     20   TEMP=(X*X+Y*Y-Z*Z) / (Z*X*Y)
     21   ANGVAL= (180/PI) * ACOS (TEMP)
     22   END
  
```

这些语句里存在常数和变量的名字。如第 7 行的 0.5 和第 11 行的 180 是两个常数，而程序中的 B 和 ANGB 是两个变量的名字，B 代表三角形的 b 边，ANGB 代表三角形的

B 角。

下面对程序进行逐行地说明。

第 1 行 描述主程序的名字为 TRIANG。

第 2 行 类型语句,说明 FLAG 是一逻辑变量的名字,它只接受逻辑值。

第 3 行 输入语句,在指定的设备上读入三个数送给变量 A、B、C,即三角形三个边的边长值。

第 4 行 输出语句,打印一串字符信息和 A、B、C 的值,帮助核对一下是不是我们需要输入的数据。

第 5 行 把三角形任意二边之和是否大于第三边的逻辑值赋给变量 FLAG,如果是的话 FLAG 的值为真,否则为假。

第 6 行 这是 FORTRAN77 特有的块如果语句 (BLOCK IF),表明如果 FLAG 取真值,则第 7 行至第 13 行被执行,否则执行第 15 行的语句。

注意块 IF 结构语句中的 IF—THEN—ELSE 和 END IF 是组合在一起用的。

第 7 行和第 8 行 计算三角形面积,其值赋给 AREA。其中 SQRT 是开平方标准函数。面积公式为 $AREA = \sqrt{s(s-a)(s-b)(s-c)}$ 其中 $S = \frac{1}{2}(a+b+c)$

第 9 行至第 11 行 通过函数子程序 ANGVAL 计算三个角的值。

第 12 行至第 14 行 打印字符串和三角形三个角的值。其中第 13 行中前面的 * 号为继续行的标志。

第 15 行 当输入的数据无效时(即二边之和不大于第三边)打印必要的信息,以告诉用户。

第 17 行 结束语句。表示主程序到此结束。

第 18 行 描述一个求三角形角度值的函数子程序的名字为 ANGVAL 和该子程序中所用到的“参数变量”。

第 19 行 赋值语句,即将 3.14159265 送给 PI。

第 20 行 赋值语句,将等号右边的算术表达式的值赋给变量 TEMP(利用余弦定理)。

第 21 行 赋值语句,将计算出来的角度赋给函数子程序名字 ANGVAL。

乘以 $\frac{180}{\pi}$ 是将弧度值化成角度值。

第 22 行 结束语句,表示这个子程序到此结束。

三、FORTRAN77 源程序的书写格式

由于书写 FORTRAN77 程序时的格式要求很严格。为此,总把 FORTRAN77 程序书写在格式规范化的程序纸上,程序纸的格式见表 1-2。

这种程序纸一般包含 20 行,每行被均分为 80 列,每列可写一个字符,每行最多可写 80 个字符。列的编号自左至右进行,统编为 1~80 列。同时,又把程序纸上每行这 80 列分成四个区段。这四个区段的名称范围和作用是:

1. 语句标号段

语句标号段的范围是从第 1 列至第 5 列,用来写语句的语句标号用。如果一个语句需要写语句标号时,则必须写在这个区段内。