

●高等学校非计算机专业计算机基础教育丛书●

FORTRAN 77/90 程序设计及软件基础

主 编 范贻明 沈雪勤
副主编 李惠然 陈国鹰

人民邮电出版社



TP312
F15

386260

高等学校非计算机专业计算机基础教育丛书

FORTRAN 77/90 程序 设计及软件基础

主 编 范贻明 沈雪勤
副主编 李惠然 陈国鹰

TP312
#15

人 民 邮 电 出 版 社

内 容 提 要

本书以国家教委工科计算机基础课程教学指导委员会对“高级语言程序设计”课程的基本要求为依据,同时参照国家教委考试中心制定的“全国计算机等级考试大纲”以及上海、北京、河北、天津等省市高教局、教委制定的考试大纲编写的。本书系统介绍了FORTRAN 77 的语法、程序设计方法及 Fortran 90 的内容,还介绍了软件工程的基础知识和有关数据结构的概念。

本书注重实用,内容深入浅出,讲解通俗易懂,适合作为高等院校理工科各类专业的程序设计课程的教材,也可作为有关人员的培训教材。

JS182/14

高等学校非计算机专业计算机基础教育丛书
FORTRAN 77/90 程序设计及软件基础

主 编 范贻明 沈雪勤
广 生 编 李基然 陈国鹰
责任编辑 赵桂珍

人民邮电出版社出版发行
北京市朝阳门内南竹杆胡同 111 号
北京密云春雷印刷厂印刷
新华书店总店科技发行所经销

*

开本:787×1092 1/16 1995 年 10 月 第一版
印张:20 1995 年 10 月 北京第 1 次印刷
字数:491 千字 印数:1—8 000 册

ISBN 7-115-05800-8/TP·230

定价:21.00 元

《高等学校非计算机专业计算机基础教育丛书》

编 委 会

主 任	范贻明			
副主任	李凤翔	李仲文	王凤先	沈雪勤
委 员	史庆周	崔来堂	魏世泽	赵淑荣
	高树森	孙光耀	杨季英	张金俊
	姜德森	岳国英	张一工	陈国鹰

序 言

近年来,我国的计算机应用发展迅速,推广普及计算机的使用已成为社会各界人士的共识。对在校的非计算机专业的学生普及计算机基础教育已是刻不容缓的任务。国家教委工科计算机基础课程教学指导委员会对高校理、工科非计算机专业拟定了五门计算机课程的教学基本要求。目前国内大多数高校都逐步开设了这五门课中的前三门课程,即《微机系统应用基础》、《高级语言程序设计》、《微机原理与应用》。这三门课的主要内容对于高等院校理、工科以外各专业学生学习计算机基础知识也是必须的。本丛书就是针对这种情况,以指导委员会制定的前三门计算机课程的基本要求为依据,同时兼顾到国家教委考试中心制定的“计算机等级考试大纲”以及上海、北京、河北、天津等省市高教局、教委有关考试大纲的要求编写的。

本丛书适合作为高等院校各专业计算机基础教育教材,也可供有关技术人员自学或用于各级等级考试补习班的教材。

为了使理论教学和实际应用密切配合,突出培养学生实际动手能力,本丛书中还包括相应的实验指导教材。

参加本书编写的人员都是多年在教学第一线从事计算机基础教学并指导实验的教师,有较丰富的理论知识和教学实践经验。

在本丛书的选题确定和编写过程中,编著者得到了河北工业大学校长颜威利教授以及其它院校的史庆周、李凤翔、魏世泽、崔来堂、赵淑荣等教授的大力支持和指导;在此谨表感谢。

欢迎广大读者对本书的内容多提批评和改进意见。

范贻明

1995年6月

前 言

本书根据国家教委工科计算机基础课程教学指导委员会对“高级语言程序设计”课程教学的基本要求,同时又参照国家教委考试中心制定的“计算机等级考试大纲”及河北、上海、北京、天津等省市高教局、教委制定的考试大纲的要求,由在教学第一线执教多年的教授、副教授等编写的。本书可作为高等院校理工科各类专业的教材。也可作为有关人员参加计算机等级考试(二级 FORTRAN 考试)的培训教材。

本书可讲授 45~50 学时,上机 20~25 学时。为了系统地培养学生上机调试程序的能力,配合本教材同时编写了“FORTRAN 实验指导及等级考试样题、习题”。

本书以我国国家标准 GB3057-82“程序设计语言 FORTRAN”为背景,突出在微机上能运行的语句形式,详细介绍了 FORTRAN77 语言及结构化程序设计方法。结合常用的算法,介绍各种程序设计方法及相应的源程序和运算结果,以便读者运用这些方法来解决各自学科领域中的有关问题。

本书共十三章,分三大部分。它们之间具有相对的独立性和内在的联系。第一章到第十章由浅入深地详细介绍了 FORTRAN77 的语法和程序设计方法。在第十一章中,对 Fortran90 语言与 FORTRAN77 语言的不同之处作了较全面的介绍。对已经学过 FORTRAN77 的读者来说,阅读了“Fortran90 介绍”这一章,可以说基本上就掌握了 Fortran90。抛开 Fortran90 这一章,其余各章也能构成完整的 FORTRAN77 内容。如果加上 Fortran90 这一章,又是一本系统介绍 Fortran 90 的书,这也正是本书的特色之一。这种安排体系既有利于新学 FORTRAN-77 语言的读者,又有利于已学过 FORTRAN77 语言的读者的学习。

为了适应二级等级考试的要求,在第十二章中介绍了软件技术基础知识,第十三章介绍了数据结构基础知识。

读者在学完本书后,配合上机实习,应能独立地编写出解决各种典型问题的程序。能满足国家教委对“高级语言程序设计”课教学的基本要求,也能参加国家教委考试中心及各省市对二级 FORTRAN 语言程序设计的考试。

参加本书编写工作的有李惠然、陈国鹰、王秀珍、李茹、韩卫、刘辰牛、付灵丽等。

在本书的编写过程中,得到了河北省高校计算机教育研究会的各位理事长的支持,并得到河北省、天津市有关高校同行的大力协助,编者在此表示衷心的感谢。

计算机的发展日新月异。由于编者水平有限,本书中不妥之处在所难免,恳请读者及有关专家批评指正。

编著者

1995 年 6 月

目 录

第一章 程序设计基本概念

1.1 算法及其表示方法	1
1.1.1 算法概念	1
1.1.2 计算机算法应具备的特点	1
1.1.3 算法的表示方法	2
1.2 算法的三种基本结构	3
1.2.1 顺序结构	3
1.2.2 选择结构	3
1.2.3 循环结构	3
1.2.4 用 N—S 结构化流程图表示算法	5
1.2.5 用 PAD 图及其它图示方法表示算法	7
1.3 结构化程序设计与程序设计风格	8
1.4 FORTRAN 的发展概况	10
习题一	10

第二章 FORTRAN 语言程序设计基础

2.1 FORTRAN 语言简单程序分析	12
2.2 FORTRAN 源程序的书写格式	14
2.3 FORTRAN77 字符集	15
2.4 FORTRAN 数据类型	16
2.4.1 在 FORTRAN77 中所使用的数据及数据类型	16
2.4.2 FORTRAN77 数据类型的特点	16
2.5 常量	17
2.5.1 整型常量	17
2.5.2 实型常量	17
2.6 变量	18
2.6.1 变量名	19
2.6.2 变量的类型	19
2.7 FORTRAN 的内部函数	20
2.8 数组	22
2.8.1 数组概念的引出	22
2.8.2 数组的定义	24
2.8.3 数组元素及下标	26
2.8.4 数组在内存中的存储顺序	26
2.9 FORTRAN77 算术表达式	28
2.9.1 算术运算符和运算规则	28

2.9.2 算术表达式及其运算顺序	28
2.9.3 算术表达式运算中的类型问题	29
习题二	30
第三章 顺序结构及基本语句	31
3.1 顺序结构	31
3.2 赋值语句及赋值规则	32
3.2.1 赋值语句的形式及作用	32
3.2.2 算术赋值语句规则	33
3.3 输入和输出	33
3.3.1 输入/输出的基本概念	33
3.3.2 输入/输出语句	34
3.4 表控输入/输出	35
3.4.1 表控输入语句	35
3.4.2 表控输出语句	37
3.5 格式输入/输出	38
3.5.1 微机常用格式输入/输出语句	38
3.5.2 格式语句	38
3.5.3 可重复格式说明符	39
3.5.4 非重复格式符	43
3.5.5 格式控制	47
3.6 参数说明语句	48
3.7 DATA 语句	49
3.8 END STOP PAUSE 语句	50
3.8.1 END 语句	50
3.8.2 STOP 语句	50
3.8.3 PAUSE 语句	50
3.9 程序举例	51
习题三	52
第四章 逻辑运算及选择结构	55
4.1 关系运算符和关系表达式	55
4.1.1 关系运算符	55
4.1.2 关系表达式	55
4.2 逻辑运算符和逻辑表达式	56
4.2.1 逻辑常量	57
4.2.2 逻辑变量	57
4.2.3 逻辑运算符	57
4.2.4 逻辑表达式	58
4.2.5 逻辑表达式的运算规则	58
4.3 逻辑赋值语句与逻辑数据的输入输出	60
4.3.1 逻辑赋值语句	60

4.3.2 逻辑数据的输入输出	60
4.4 用块 IF 实现选择结构	61
4.4.1 构成块 IF 的基本语句	61
4.4.2 块 IF 的基本结构	62
4.5 块 IF 的嵌套	65
4.6 GOTO 语句和逻辑 IF 语句	67
4.6.1 无条件转移语句——GOTO 语句	67
4.6.2 逻辑 IF 语句	68
4.7 程序举例	69
习题四	75
第五章 循环结构及数组操作	78
5.1 循环的定义和类型	78
5.2 用 GOTO 语句实现循环	78
5.3 DO 循环和 CONTINUE 语句	79
5.3.1 DO 语句的一般形式及循环次数的计算	79
5.3.2 DO 循环的结构及执行过程	81
5.3.3 对 DO 循环的有关规定	81
5.3.4 CONTINUE 语句	84
5.3.5 DO 循环语句举例	84
5.3.6 DO 循环的嵌套(多重循环)	85
5.3.7 DO 循环程序举例	87
5.4 当型循环的实现	90
5.4.1 标准 FORTRAN77 中当型循环的实现	90
5.4.2 用 DO WHILE 语句实现当型循环	93
5.5 直到型循环的实现	93
5.5.1 用 DO UNTIL 语句实现直到型循环	94
5.5.2 用逻辑 IF 语句内嵌 GOTO 语句实现直到型循环	94
5.6 几种循环的比较	96
5.7 数组常见操作(数组的输入/输出)	98
5.7.1 用 DATA 语句给数组赋初值	98
5.7.2 用数组名输入/输出	99
5.7.3 按数组元素名输入/输出	100
5.7.4 用 DO 语句控制输入/输出	101
5.7.5 隐含输入/输出表	102
5.8 DO 循环与数组应用举例	103
习题五	109
第六章 FORTRAN 数据结构	112
6.1 程序中的数据结构	112
6.2 双精度型数据	113
6.2.1 双精度型常数	113

6.2.2 双精度型变量和数组	113
6.2.3 双精度型表达式和赋值语句	114
6.2.4 双精度型数据的输入/输出	114
6.3 复型数据	116
6.3.1 复型常数	116
6.3.2 复型变量和数组	117
6.3.3 复型表达式及复型运算	117
6.3.4 复型数据的输入输出	118
6.3.5 举例	119
6.3.6 数值型数据小结	121
6.4 字符型数据	123
6.4.1 字符型常数	123
6.4.2 字符型变量和数组	123
6.4.3 子字符串	124
6.4.4 字符表达式及字符赋值语句	125
6.4.5 字符型关系表达式	126
6.4.6 字符型函数	127
6.4.7 字符型数据的输入/输出	128
6.4.8 举例	130
习题六	132
第七章 常用基本算法及程序设计举例	134
7.1 穷举法	134
7.2 递推和迭代算法	136
7.2.1 递推算法	136
7.2.2 迭代算法	137
7.2.3 用迭代法解一元方程	138
7.3 一元方程求根的其它算法	141
7.3.1 用二分法求方程的根	141
7.3.2 用弦截法求方程的根	143
7.4 排序算法	145
7.4.1 比较互换法	145
7.4.2 选择法排序	146
7.4.3 冒泡法排序算法	148
7.4.4 希尔法排序	149
7.5 检索	151
7.5.1 线性检索	152
7.5.2 对分检索	153
7.6 数值积分	154
7.6.1 矩形法	155
7.6.2 梯形法	156

7.6.3 辛普生法	157
7.6.4 变步长辛普生求定积分	158
习题七	160
第八章 过程	161
8.1 过程分类	161
8.2 语句函数	162
8.2.1 语句函数的定义	162
8.2.2 语句函数的调用与执行	163
8.2.3 调用形式举例	164
8.3 函数子程序	166
8.3.1 函数子程序的引入	166
8.3.2 函数子程序的定义	167
8.3.3 函数子程序的调用	169
8.3.4 函数子程序应用举例	174
8.4 子例行程序	179
8.4.1 子例行程序的引入	179
8.4.2 子例行程序的定义	180
8.4.3 子例行程序的调用	181
8.4.4 子例行程序应用举例	181
8.5 可调数组及其应用	183
8.5.1 可调数组的概念	183
8.5.2 使用可调数组时应注意的事项	184
8.5.3 可调数组应用举例	184
8.6 外部语句(EXTERNAL)和内部语句(INTRINSIC)	185
8.7 综合应用举例	187
习题八	192
第九章 数据联系语句和数据块子程序	194
9.1 公用语句(COMMON 语句)	194
9.1.1 公用语句概述	194
9.1.2 COMMON(公用)语句的一般形式	194
9.1.3 无名公用区及其使用规则	195
9.1.4 有名公用区及其使用规则	197
9.1.5 举例	199
9.2 数据块子程序	200
9.3 等价语句(EQUIVALENCE 语句)	202
习题九	204
第十章 文件及其使用	206
10.1 引言	206
10.2 文件的基本概念	207
10.2.1 记录的格式与长度	207

10.2.2 文件的存取方式	208
10.2.3 外部文件	209
10.2.4 文件读写(输入/输出)过程概述	209
10.3 文件的打开与关闭	209
10.3.1 文件的打开	209
10.3.2 文件的关闭	212
10.4 文件读写语句	213
10.4.1 文件读写语句的一般形式及功能	213
10.4.2 有格式顺序文件的读写	214
10.4.3 有格式直接文件的读写	215
10.4.4 无格式文件	217
10.5 文件的定位与查询	218
10.5.1 文件定位	218
10.5.2 文件查询	220
10.6 文件使用举例	221
习题十	224
第十一章 Fortran90 介绍	226
11.1 字符集、基本术语和程序结构	227
11.1.1 字符集	227
11.1.2 新增基本术语及定义	227
11.1.3 源程序	229
11.2 基本语句功能的扩展及新增基本语句	231
11.2.1 类型说明语句	231
11.2.2 输入/输出语句功能的扩充	233
11.2.3 赋值语句的类型、种别及表达式的几点注释	235
11.3 含块控制构造	236
11.3.1 CASE 构造	236
11.3.2 块 DO 构造	238
11.4 数组的使用及动态结合	240
11.4.1 数组的使用	241
11.4.2 数组的运算	244
11.4.3 数组的内在函数	245
11.4.4 数组的形式	249
11.4.5 数组的动态结合	250
11.5 数组导出类型	251
11.5.1 数组结构	251
11.5.2 导出类型	251
11.6 模块与过程接口块	252
11.6.1 模块	252
11.6.2 过程接口块	255

11.7 指针	256
11.7.1 指针(POINTER)属性	256
11.7.2 目标(TARGET)属性	257
11.7.3 指针赋值	257
11.8 递归过程	258
11.8.1 递归函数	258
11.8.2 递归子程序	259
11.8.3 递归过程举例	259
11.9 定义运算与赋值	260
11.9.1 定义运算	260
11.9.2 定义赋值	261
11.10 新增内在函数及内在子程序	262
11.10.1 新增类属内在函数	262
11.10.2 内在子程序	265
11.11 FORTRAN77 中过时或将被淘汰的内容	265
11.11.1 过时的内容	265
11.11.2 将被淘汰的内容	266
习题十一	267
第十二章 软件技术基础知识	268
12.1 软件工程概述	268
12.2 软件需求分析	269
12.3 软件概要设计	271
12.4 详细设计及其工具	274
12.5 结构化编程和程序设计风格	276
12.5.1 对源程序的质量要求	276
12.5.2 程序设计语言的选择	277
12.5.3 结构化程序设计	277
12.5.4 程序设计风格	279
12.6 软件测试和维护	280
12.6.1 软件测试	280
12.6.2 软件维护	284
习题十二	285
第十三章 数据结构基础知识	286
13.1 数据结构的产生和发展	286
13.2 数组	286
13.2.1 一维数组	286
13.2.2 二维数组	287
13.2.3 多维数组	289
13.2.4 稀疏矩阵	289
13.3 线性表	294

13.3.1 线性表	294
13.3.2 栈的定义及基本操作	295
13.3.3 队列的定义及基本操作	298
13.4 链表	298
13.4.1 线性链表	298
13.4.2 环形链表和双向链表	300
13.5 树	301
13.5.1 树的基本概念	301
13.5.2 二叉树	301
13.5.3 查找树	303
习题十三	303
附录 ASCII 码表	304

第一章 程序设计基本概念

计算机的应用虽已深入到各行各业,但是要用计算机处理问题,就必须写出使计算机按人们意愿进行工作的计算机程序。所以每一个学习计算机知识的人都应当学习如何进行程序设计。

为了有效地进行程序设计,至少要具有两方面的知识,即:①掌握解决问题的方法和步骤——算法。②掌握一门计算机语言的语法规则,编写出计算机程序。一般来说,程序设计过程中的基本工作是:定义问题、算法设计和数据结构设计、流程图(或N-S图等)设计、用选定的语言编制程序、调试程序、测试程序、编制文档资料、维护和再设计等。

1.1 算法及其表示方法

1.1.1 算法概念

日常生活中做任何事情都有一定的步骤。所谓算法,就是为解决某一个问题而采取的方法和步骤。当然对同一个问题,可以有不同的解题方法和步骤,这样,算法就有优劣之分。

一般应采用方法简单,运算步骤少的算法,除保证算法正确外,还要考虑算法的质量。

计算机算法分为数值运算算法和非数值运算算法两类。数值运算算法其目的是求数值解。例如,代数方程的数值解,常微分、偏微分方程的数值解算法等。对数值算法的研究比较深入和成熟。非数值算法多用于信息处理和人工智能等方面。例如,情报检索,排序算法等。

1.1.2 计算机算法应具备的特点

1. 有限性(有穷性)

一个算法应包括有限的操作步骤,应在有限次执行后完成。

2. 确定性

一个算法中的每一个步骤,必须有明确的定义。而不应含糊不清模棱两可,算法的含义应当是唯一的。

3. 有零个或多个输入

所谓输入是指在执行算法时需要从外界取得必要信息。算法总是要施加到运算对象上,输入描述了运算对象的初始情况。

4. 可行性

算法的每一步都是能够准确地执行的,并能得到确定的结果。例如,在执行 $1/N$ 除法中,当 $N=0$ 时,计算机运算 $1/N$ 溢出,这表明算法无法执行。

5. 有一个或多个输出

算法的最终归宿就是要得到输出结果。没有输出的算法是没有意义的。

1.1.3 算法的表示方法

对一个算法的描述可以采用多种表示方法,主要有以下几种:

- ① 用自然语言;
- ② 用框图或流程图;
- ③ 用计算机语言或伪代码。

用计算机语言描述的算法就是计算机程序。

一、用类似于数学语言的自然语言

自然语言就是人们日常使用的语言,可以是汉语、英语或其它文字语言。用自然语言描述一个算法通俗易懂,但文字冗长,易出现“多意性”,含意不太严格,除一些很简单的问题外,一般不用自然语言来描述算法。

二、用框图或流程图

在用图示的方法来描述算法的方法中,有传统流程图、N-S 结构化流程图和 PAD 图等。

1. 用传统流程图表示算法

一个算法可以用流程图(框图)来表示。传统的流程图是用一些图框来表示一些特定含意的操作。用图形表示算法,一目了然,易于理解。

美国国家标准化协会(ANSI)规定了一些常用的流程图符号,如图 1-1 所示。现已被世界各国所普遍采用。

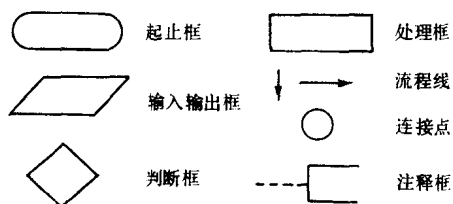


图 1-1 流程图使用符号

图 1-1 中的处理框,表示一般的叙述事项;菱形框的作用是对一个给定的条件进行判断,它有一个入口两个出口,根据判断给定条件是否成立,而选定两出口之一向下运行;连接点(圆圈)是将两幅流程图连起来的接点。采用连接点,可以避免流程线过长或交叉,避免流程图的图幅过大。

2. 用 ISO 流程图表示算法

随着结构化程序设计方法的出现,使流程图也暴露出一些不足之处,比如它没有表示循环结构的图框,使用带箭头的流程线,允许流程的随意 GOTO 转移等。针对上述情况,国际标准化组织(ISO)在 ANSI 提出的流程图符号的基础上,提出了 ISO 流程图符号。除包括 ANSI 的流程图符号外又增了表示循环的符号;表示多出口的判断符号;规定当流程由上到下和由左到右时,流程线可以不带箭头,反之,若流程从下向上或从右向左时,流程线要加箭头。ISO 补充 ANSI 的流程图符号见图 1-2。

我国已接受 ISO 标准提出的程序流程图符号,并在此基础上制订了国家标准。国标中除保留了 ISO 的图框符号外还增加了少量框图符号。

为了提高算法的质量,人们设想,如果规定出几种基本结构,由这些基本结构由上而下按一定规律组成一个算法结构。这样的算法结构,结构清晰,易被人理解,从而提高了算法的质量。

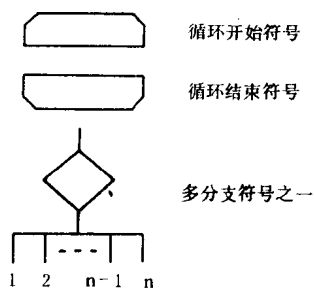


图 1-2 ISO 部分流程图符号

1.2 算法的三种基本结构

1966 年,Bohra 和 Jacopini 提出了三种基本结构,即顺序结构、选择结构和循环结构。人们可以用这三种基本结构作为表示一个良好算法的算法结构。

1.2.1 顺序结构

顺序结构如图 1-3 所示。虚线框的内容是一个顺序结构。其中,图框 A 和图框 B 是顺序执行的处理框。顺序结构是最简单的一种结构。其入口为 a 点,出口为 b 点。

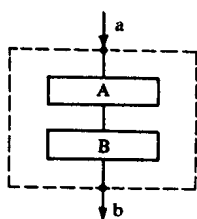


图 1-3 顺序结构框图

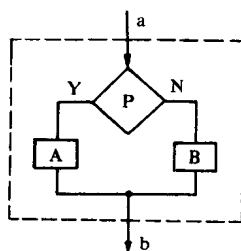


图 1-4 选择结构框图

1.2.2 选择结构

选择结构如图 1-4 所示。此种结构中必定包含一个判断框。根据给定的条件 P 是否成立而选择执行 A 框或 B 框。当条件成立时,执行 A,否则执行 B。根据需要 A 框或 B 框可以是空框,即不执行任何操作。但判断框中的两个分支,执行完 A 或 B 之后都是经过唯一的出口 b 点。

1.2.3 循环结构

循环结构又称重复结构。分为当型循环结构和直到型循环结构两类。二者都是重复执行