

计算机等级考试丛书( '98 大纲)

谭浩强 主编

# **FORTRAN** 程序设计(二级)教程

谭浩强 崔武子 田淑清 编著

清华大学出版社

(京)新登字 158 号

## 内 容 简 介

本书是根据教育部考试中心 1998 年公布的《全国计算机等级考试 考试大纲》中关于 FORTRAN 语言程序设计的考试要求而编写的。内容包括:FORTRAN 语言的基本概念、FORTRAN 语言的语法规则以及利用 FORTRAN 语言进行程序设计的有关知识。

本书覆盖了考试大纲所要求的范围,系统而简明地介绍了考生应该掌握的内容。每章之后均附有精选的练习题,以帮助读者学习和掌握。最后还附有考试大纲和一套笔试试题。

本书可以作为参加计算机等级考试人员的自学或培训教材,也可以作为大专院校 FORTRAN 语言程序设计课程的教学用书。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

书 名: FORTRAN 程序设计(二级)教程

作 者: 谭浩强 崔武子 田淑清 编著

出版者: 清华大学出版社(北京清华大学学研楼,邮编 100084)

[http:// www .tup .tsinghua edu cn](http://www.tup.tsinghua.edu.cn)

印刷者: 北京密云胶印厂

发行者: 新华书店总店北京发行所

开 本: 787 × 1092 1/ 16 印张: 17.25 字数: 398 千字

版 次: 2000 年 5 月第 1 版 2000 年 5 月第 1 次印刷

书 号: ISBN 7-302-01141-9/ TP·349

印 数: 0001 ~ 5000

定 价: 22.00 元

# 计算机等级考试丛书( '98 大纲)

## 序

进入 20 世纪 90 年代以来,我国掀起了第二次计算机普及高潮。人们已经认识到,要建设一个现代国家,离开计算机将寸步难行;无论从事什么工作,都必须学习计算机知识,掌握计算机应用。不少单位已经把通过计算机考试作为任职条件。

为了适应社会的需要,原国家教委考试中心于 1994 年推出了“全国计算机等级考试”,受到社会各界的热烈欢迎。至 1998 年底,已有 160 多万人报名考试,其中 70 多万人获得了等级证书。全国许多地区和部门也组织了本地区或本系统的计算机统一测试。许多同志认为,“学历”是从整体上反映了一个人的知识水平,而“证书”则反映了一个人在某一方面的能力。证书制度是学历制度的必要补充,是人才市场的需要,因而受到各方面的欢迎。

当然,计算机统一考试无论从内容上还是形式上都有一些问题需要进一步探索,例如,如何更好地反映计算机新技术的发展,怎样才能测试出应试者的实际应用能力等。相信随着时间的推移和经验的累积,这项制度会日趋完善。

全国和地区性的计算机等级考试在内容上、分级上、考试形式上大体相似。全国计算机等级考试目前分为四个等级。一级:具有计算机的初步知识和使用微机系统的初步能力。一级(B):面向公务员和在职干部,要求大致与一级相当,侧重应用能力。二级:具有计算机软件、硬件的基础知识和使用一种高级语言编制程序、上机调试的能力,可以从 QBASIC, FORTRAN, PASCAL, C, FoxBASE 五种语言中任选一种应试。三级分为两类:三级(A):具有计算机应用基础知识和计算机硬件系统开发的初步能力;三级(B):具有计算机应用基础知识和软件系统开发的初步能力。四级:具备深入而系统的计算机知识和较高的计算机应用能力。

为了帮助广大读者准备参加等级考试,1995 年我们组织编写了“计算机等级考试辅导”丛书,由清华大学出版社出版。该丛书包括一级、一级 B、二级(五种)的考试辅导共 8 本书,在出版后三年多时间内已发行 120 多万册。

根据计算机应用的发展,1998 年教育部考试中心修改了全国计算机等级考试大纲。和原大纲相比,新大纲有以下改动:一级除了可以选择 DOS 环境应试外,还可以选择 Windows 环境应试;二级的 BASIC 语言改为 QBASIC 语言;在二级的计算机基础知识部分中增加了对网络和多媒体的要求。

根据广大读者的要求,我们在原来的“计算机等级考试辅导”丛书的基础上,组织出版这套“计算机等级考试丛书( '98 大纲)”。该丛书由以下三个系列构成:(1)计算机等级考试教程:全面而系统地介绍考试大纲所规定的内容,是便于自学的教材。(2)计算机等级考试辅导:用来帮助已学过该课程的读者复习和准备考试,每本书的内容均包括各章要

点、各章难点、例题分析、思考题,并附有模拟试题。(3)计算机等级考试样题汇编:按照计算机等级考试的内容和试题形式,提供了500~600道样题,供应试者准备考试参考。

本丛中各书的作者都是高等学校或计算机应用部分中具有丰富教学经验并对计算机等级考试有较深入研究的教授、专家。相信该丛书的出版一定会受到广大准备参加计算机等级考试的读者的欢迎。

欢迎读者对本丛书提出宝贵意见。

“计算机等级考试丛书( '98 大纲)”主编

全国计算机等级考试委员会副主任

谭浩强

1999年3月

# 前 言

FORTRAN 77 是国内外广泛使用的计算机高级语言,它主要适用于数值计算领域。FORTRAN 语言是世界上最早出现的高级语言,多年来形成了一个功能丰富的子程序库,将常用的数值计算的算法编写成子程序,提供用户使用。因此,FORTRAN 语言是从事数值计算的人员首选的计算机语言。它拥有众多的使用者,直到今天仍然有强大的生命力。

由教育部考试中心主办的全国计算机等级考试将 FORTRAN 77 语言列为二级考试五种语言之一。每年都有许多人报考 FORTRAN 语言。各地区举办的大学生计算机统一考试,也多将 FORTRAN 语言列为考试内容。

为了帮助报考全国计算机等级考试和其它统一考试的同志准备考试,我们特编写了这本教材。本书是根据教育部考试中心 1998 年公布的全国计算机等级考试二级 FORTRAN 考试大纲编写的,内容覆盖大纲的全部内容。

本书是在谭浩强、田淑清编著的《FORTRAN 语言—FORTRAN 77 结构化程序设计》(清华大学出版社出版)一书的基础上根据等级考试的要求和特点加以改编、整理而成的。上述该书被认为是众多的 FORTRAN 77 语言教材中的优秀读物,发行量已超过 72 万册,居全国同类书之首,被国家教育部评为全国优秀教材,大多数高校均选它为正式教材。本书保留了该书通俗易懂的特点,同时对内容做了必要的增删,使之符合等级考试的需要。在本书的各章后面都附有符合等级考试要求和形式的习题,最后还附有一套模拟试题,供练习用。

本书由谭浩强、田淑清、崔武子合作完成,改写的工作主要由崔武子老师负责。

本书如有不足之处,敬请批评指正。

作 者

1999 年 9 月 1 日

# 目 录

<b>第 1 章 FORTRAN 程序的结构和书写规则</b> .....	1
1.1 计算机语言和程序设计 .....	1
1.1.1 基本概念 .....	1
1.1.2 算法 .....	1
1.1.3 结构化程序设计和模块化结构 .....	2
1.2 FORTRAN 77 源程序的结构 .....	2
1.3 FORTRAN 77 源程序的书写格式 .....	4
1.4 FORTRAN 77 程序的上机步骤 .....	5
习题 .....	6
<b>第 2 章 数据类型及其运算</b> .....	8
2.1 常量和变量的类型 .....	8
2.1.1 常量 .....	8
2.1.2 变量 .....	11
2.2 定义变量类型的方法 .....	12
2.3 符号常量及其定义方法 .....	13
2.4 运算符及其运算优先级 .....	14
2.4.1 运算符 .....	14
2.4.2 运算优先级 .....	16
2.5 表达式及其求值规则 .....	17
2.5.1 算术表达式 .....	17
2.5.2 关系表达式 .....	19
2.5.3 逻辑表达式 .....	20
2.5.4 字符表达式 .....	21
习题 .....	21
<b>第 3 章 顺序结构</b> .....	26
3.1 赋值语句 .....	26
3.1.1 算术赋值语句 .....	26
3.1.2 逻辑赋值语句 .....	26
3.1.3 字符型赋值语句 .....	27
3.2 表控输入、输出语句 .....	27
3.2.1 表控输出 .....	28
3.2.2 表控输入 .....	28
3.3 格式输入、输出语句 .....	30

3.3.1	格式编辑符.....	31
3.3.2	格式语句与输入、输出语句的相互作用 .....	45
3.3.3	在输入、输出语句中包含格式说明的方法 .....	54
3.4	STOP 语句和 END 语句 .....	55
3.5	DATA 语句.....	55
3.6	GOTO 语句 .....	56
	习题 .....	56
<b>第 4 章</b>	<b>选择结构 .....</b>	<b>63</b>
4.1	逻辑 IF 语句 .....	63
4.2	块 IF 结构 .....	64
4.3	块 IF 结构的嵌套 .....	66
4.4	程序举例.....	68
	习题 .....	70
<b>第 5 章</b>	<b>循环结构 .....</b>	<b>77</b>
5.1	DO 循环结构 .....	77
5.1.1	DO 循环的一般形式 .....	77
5.1.2	DO 循环的执行过程 .....	79
5.1.3	用 CONTINUE 语句作终端语句 .....	80
5.2	用块 IF 和 GOTO 语句构成的第一种循环 .....	82
5.2.1	用块 IF 和 GOTO 语句构成第一种循环的形式 .....	82
5.2.2	循环的执行过程.....	82
5.3	用逻辑 IF 语句构成的第二种循环 .....	85
5.3.1	用逻辑 IF 语句构成第二种循环的形式 .....	85
5.3.2	循环的执行过程.....	85
5.4	循环结构的嵌套.....	86
5.5	程序举例.....	87
	习题 .....	92
<b>第 6 章</b>	<b>数组.....</b>	<b>102</b>
6.1	数组的定义和数组元素的引用 .....	102
6.1.1	数组的定义 .....	102
6.1.2	数组元素的引用 .....	104
6.2	数组的逻辑结构和存储结构 .....	104
6.3	对数组赋初值的方法 .....	105
6.4	数组的输入和输出 .....	106
6.4.1	利用 DO 循环进行输入和输出数组元素 .....	106
6.4.2	用数组名输入或输出整个数组 .....	107
6.4.3	在输入、输出语句中运用隐含 DO 循环 .....	109
6.5	程序举例 .....	110

习题.....	120
<b>第 7 章 字符处理</b> .....	136
7.1 字符串、字符型变量和字符型数组.....	136
7.2 字符型数据的赋值和运算 .....	136
7.2.1 字符型数据的赋值 .....	136
7.2.2 字符型数据的运算 .....	137
7.3 字符型数据的输入和输出 .....	139
习题.....	143
<b>第 8 章 函数和子程序</b> .....	150
8.1 内部函数的调用 .....	150
8.2 语句函数的定义及调用 .....	151
8.2.1 语句函数的定义 .....	151
8.2.2 语句函数的引用 .....	152
8.3 函数子程序的结构及调用 .....	153
8.3.1 函数子程序的结构 .....	153
8.3.2 函数子程序的调用 .....	155
8.4 子例程程序的结构及调用方法 .....	156
8.4.1 子例程程序的结构 .....	156
8.4.2 子例程程序的调用 .....	157
8.5 虚参和实参之间的数据传递 .....	159
8.5.1 变量作为虚参 .....	159
8.5.2 数组作为虚参 .....	160
8.5.3 子程序名作为虚参 .....	164
8.5.4 星号“ * ”作为虚参 .....	165
8.6 程序举例 .....	165
习题.....	172
<b>第 9 章 数据联系</b> .....	189
9.1 公用语句和公用区 .....	189
9.1.1 无名公用区 .....	189
9.1.2 有名公用区 .....	192
9.2 数据块子程序 .....	192
习题.....	193
<b>第 10 章 文件</b> .....	199
10.1 文件与记录的概念.....	199
10.2 文件的操作语句.....	200
10.2.1 文件的打开与关闭语句.....	200
10.2.2 返绕与回退语句.....	202
10.2.3 输入与输出语句.....	202

10 3	有格式顺序文件的操作.....	203
10 4	有格式直接文件的操作.....	206
10 5	无格式顺序文件和直接文件的操作.....	208
	习题.....	210
附录	.....	218
附录 1	常用字符与 ASCII 代码对照表.....	218
附录 2	可执行语句和非执行语句表.....	219
附录 3	程序单位中语句和注释行的顺序.....	220
附录 4	FORTRAN 77 语句索引.....	220
附录 5	FORTRAN 77 内部函数.....	222
附录 6	全国计算机等级考试二级 (基础知识和 FORTRAN 语言程序设计) 考试大纲.....	224
附录 7	1999 年 4 月全国计算机等级考试二级 (基础知识和 FORTRAN 语言程序设计) 笔试试题及答案.....	227
附录 8	各章习题参考答案.....	240
参考文献	.....	266

# 第 1 章 FORTRAN 程序的结构和书写规则

## 1.1 计算机语言和程序设计

### 1.1.1 基本概念

人们使用计算机,使计算机按人们的意志进行工作,就需要与计算机交流信息。计算机语言是人与计算机之间通信的语言。计算机语言有机器语言、汇编语言和高级语言。机器语言和汇编语言是依赖于具体机器的,可移植性差,而且难学,不易推广。而高级语言接近于自然语言和数学语言,易懂易学,而且在各种计算机上都能通用。高级语言又分为过程化语言和非过程化语言,它们的区别在于前者既要解决“做什么”,又要解决“怎样做”的问题,而后者只指出“做什么”即可。FORTRAN 语言是过程化的高级语言。学习用过程化的语言编写程序,是目前学习计算机科学技术的一项基本训练,有了这个基础,掌握非过程化语言是轻而易举的。

程序是为了解决某一个特定问题而用某一种语言编写的指令序列。程序设计一般包括确定数据结构、确定算法、编码、调试程序、整理并写出文档资料等内容。

### 1.1.2 算法

计算机语言只是一种工具,学习计算机语言的目的是要用语言作为工具,设计出所需要的程序。要编写较好的程序,除了选定合理的数据结构外,一般来说,十分关键的一步是设计正确而有效的算法。算法是指为解决某个特定问题而采取的确定的且有限的步骤。算法是程序设计的核心,算法的好坏直接影响程序的运行效率。一个算法应当具有以下五个特性:

- 有穷性。一个算法应包含有限个操作步骤。
- 确定性。算法中的每一个步骤必须都有确切的含义。
- 有效性。算法中的每一个步骤都应当能有效地执行,并得到确定的结果。
- 有零个或多个输入。在大多数情况下要处理的数据对象需要通过输入来完成。
- 有一个或多个输出。算法的目的是为了求“解”,而通过输出才能得到这些“解”。

流程图是描述算法的很好的工具,常用的流程图有传统流程图和 N-S 流程图。前者由图 1.1 中所示的几种基本框组成,它的优点是形象直观,简单方便。而 N-S 流程图完全去掉了流程线,算法的每

图 1.1

一步都用一个矩形框来描述,并把一个个矩形框按执行的次序连接起来,它的优点是描述算法时所占篇幅较少,省时省力且易阅读。

### 1.1.3 结构化程序设计和模块化结构

#### 1. 结构化程序设计

结构化程序设计由三种基本结构即顺序结构、选择结构和循环结构组成。由三种基本结构构成的程序称为结构化程序。用结构化程序实现的算法可以解决任何复杂的问题。FORTRAN 77 是接近于结构化语言的高级语言,它提供了三种构成基本结构的语句,可以用来进行结构化程序设计。

#### 2. 模块化结构

因为解决复杂问题时所编写的程序由成千上万条语句组成,因此经常需要由许多人来共同完成,这时常常把任务分解为若干个子任务,每个子任务由若干模块构成,一个模块往往用来完成一项简单的功能,模块化结构是由一个个功能模块构成的程序结构。程序员在进行程序设计的时候,应当先考虑主程序中的算法,再依次完成各个子任务,这就是自顶向下、逐步细化、模块化的程序设计。

FORTRAN 程序可以由主程序块、函数子程序块或子例行程序块组成。主程序是 FORTRAN 程序中唯一必不可少的程序块,程序的执行总是从主程序的第一条可执行语句开始。函数子程序块和子例行程序块是被主程序或其它子程序调用的程序块,函数子程序必须由 FUNCTION 开头,而子例行程序必须由 SUBROUTINE 开头。子程序和函数不能单独执行。

## 1.2 FORTRAN 77 源程序的结构

下面通过一个简单的例子,介绍 FORTRAN 程序的一些基本结构。

[例 1-1] 从键盘输入两个数,求它们的平均值。

程序如下:

```
WRITE ( * , * ) Enter A and B:  
READ ( * , * ) A,B  
AVE = (A + B) / 2.0  
WRITE ( * , * ) AVE = , AVE  
END
```

以上程序的运行结果如下:

```
Enter A and B: 2 4 回车      (在本书中用户从键盘输入的数据都加下划线来表示)  
AVE =          3.000000
```

以上程序只有主程序,共包括五条语句,第一条和第四条语句是输出语句,在终端上输出指定的数据;第二条语句是输入语句,从键盘输入的数 2 和 4 分别赋给变量 A 和 B;

第三条语句是赋值语句,它将 2 与 4 的平均值赋给变量 AVE;第五条语句是结束语句,它的作用是使程序停止运行。

[例 1-2] 计算底面半径和高分别为 2、5 和 3、4 的圆柱体体积。

程序如下:

```
V1 = VOLUME(2.0,5.0)
WRITE (*,*) V1 = ,V1
V2 = VOLUME(3.0,4.0)
WRITE (*,*) V2 = ,V2
END
```

C a function subprogram

```
FUNCTION VOLUME(R,H)
VOLUME = 3.14159 * R * * 2 * H
END
```

以上程序的运行结果如下:

```
V1 =      62.83180
V2 =      113.0972
```

程序由一个主程序和一个函数子程序块构成。主程序通过对函数子程序的两次调用得到两个圆柱体的体积。函数子程序块的功能是根据传送来的半径 R 和高 H 的值计算体积,并将计算结果存放在函数子程序名 VOLUME 中。函数子程序的第一行第一列字符为 C,表示该行是注释行,说明子程序的性质;第二行是函数子程序的首行,其中的 VOLUME 是函数名,函数名后的一对括号内有两个名为 R 和 H 的参数,它们的值在此子程序被调用时得到。当执行函数子程序的最后一行 END 语句后,执行的流程返回主程序,并通过函数名 VOLUME 带回所求得的圆柱体体积的值。

以上举例只是对主程序和子程序的概念作简单介绍。关于子程序,后面章节还有详细的讲解。

从以上例子可以看到:

(1) 一个程序由一个或若干个程序块组成。主程序和每一个子程序分别是一个独立的程序块。在解决一个比较复杂的问题时,可以分别将每一个功能编为一个子程序,然后按搭积木一样将各有关程序块组成一个程序。一个程序块就是一个模块,结构化程序设计需要采用模块化方法,在 FORTRAN 中,子程序是实现模块化的有效方法。

(2) 每一个程序块都是以“END”结束的。在 FORTRAN 77 中,END 既是一个程序单位的结束标志,又是一个独立的语句(结束语句)。主程序中的 END 语句的作用是使程序运行“停止”。子程序中的 END 语句是“使流程返回调用程序”。应注意,在一个程序单位中只能有一个 END 语句,如果需要,可以使用多个 STOP 语句或 RETURN 语句组成各个程序单位。

(3) 一个程序块包括若干行。FORTRAN 行分为两类:

语句行。由一个 FORTRAN 语句组成,它对程序运行情况是有影响的。FOR-

TRAN 语句分为: 执行语句和非执行语句。执行语句使计算机在运行时产生某些操作, 如赋值语句、打印语句等。非执行语句(包括说明语句、数据语句等)将有关信息通知编译系统, 以便在编译时做出相应的处理, 例如, 类型说明语句、函数子程序语句等。

非语句行, 即注释行。它不是 FORTRAN 语句, 它不被翻译成机器目标指令, 不产生任何机器操作。它仅仅是为了人们阅读程序的方便而加到程序中的。一个程序中注释行的数目不受限制, 根据需要而定。但一个程序单位不能只由注释行组成。注释行是程序的一部分, 在打印程序清单时, 注释行也原样打印。注释行的内容完全是根据程序设计人员需要而写的, 一般是为程序(或程序中某一部分)的作用作注释。注释可以用英语或汉语拼音文字, 如果所用的操作系统是中文操作系统(如 CCDOS), 还可以用汉字作注释。

FORTRAN 程序的基本成分是语句。FORTRAN 规定, 一行只能写一个语句。如果一个语句太长, 一行内写不下, 可以分行继续写(要使用“继续标志”)。

(4) FORTRAN 程序中的语句可以没有标号, 也可以有标号, 根据需要而定。标号的作用是标志一个语句, 以便被其它语句引用。在同一个程序单位中不能有两个相同标号的语句。标号不影响语句的执行顺序。程序中语句执行的顺序一般按照它们在程序中的位置而定, 即写在前面的语句先执行, 写在后面的语句后执行。

(5) 一个程序块中各类语句的位置是有规定的, 例如, PROGRAM 语句应是主程序的第一个语句, FUNCTION 语句是函数子程序的第一个语句, END 语句只能是程序单位中最后一行。

(6) FORTRAN 源程序必须按一定的格式书写。哪些内容应写在一行中的哪一列(或哪几列上)都有严格的规定。

### 1 3 FORTRAN 77 源程序的书写格式

用 FORTRAN 语句编写的程序称为 FORTRAN 源程序。FORTRAN 源程序必须按规定的格式书写。上机运行时, 也要按格式输入源程序。

FORTRAN 源程序由程序块组成, 每个程序块由若干行组成, 每一行有 80 列, 每一列只能写 1 个字符。每个程序块共分四个区, 即标号区、续行标志区、语句区和注释区。

(1) 标号区(第 1~5 列)。可以写 1~5 位整数, 也可以无标号。标号区内的空格不起作用, 标号的大小不影响程序的执行顺序。第一列上只能出现数字、空格、“C”或“\*”, 否则编译时按出错处理。第一列为“C”或“\*”的, 该行按注释行处理。

(2) 续行标志区(第 6 列)。如果在一行的第 6 列写上一个非零非空格的字符, 则表示该行是上一行的续行。在编译时把两行连接在一起, 成为一个语句。FORTRAN 标准规定, 一个语句可以有 19 个续行, 即一个语句最多可写 20 行, 每一行的开始和结尾不一定是第 7 列和第 72 列, 但续行符号必须写在第 6 列。注释行不能有续行。

(3) 语句区(第 7~72 列)。语句可以由第 7 列以后任何位置开始书写。但一行内只能写一个语句。语句区内的空格(除引号内字符串中的空格)在编译时被忽略。

(4) 注释区(第 73~80 列)。程序员常利用这 8 列为程序行编顺序号, 以便查找。一

般习惯是用前 3 列编页数,后 5 列编行数,例如,00100010 表示程序的第 1 页第 10 行。在编译时对 72~80 列不予处理,但打印源程序清单时原样打印。

## 1.4 FORTRAN 77 程序的上机步骤

用户写好一个 FORTRAN 77 源程序后要上机运行它,必须先安装 FORTRAN 77 系统,安装方法是:在 C 盘根目录下建立 FOR 子目录,再把 FORTRAN 77 系统程序复制到该目录。下面以[例 1-1]中程序为例,叙述 FORTRAN 77 源程序的编辑、编译、连接和运行过程。

### 步骤 1: 调用编辑程序输入源程序

上机的第一步是输入源程序,建立新文件。为此,必须调用编辑程序,编辑程序种类较多,下面只介绍目前常用的 EDIT 编辑程序,它是 DOS 提供的编辑器。

方法 1:在 C:\FOR>提示符下,键入 EDIT F0101.FOR 回车,则启动 EDIT 编辑器,并建立了名为 F0101.FOR 的新文件,此处 F0101 是由用户自己起的名字,其后的 .FOR 是扩展名,对于 FORTRAN 程序,扩展名必须用 .FOR。

方法 2:在 C:\FOR>提示符下,键入 EDIT 回车,进入编辑状态后,选择“save as”命令,在出现的对话框中输入 F0101.FOR 回车。

以上两种方法的功能完全相同。当用户进入编辑状态后,就可以从键盘输入程序。输入程序时,一定要及时将输入的内容保存到磁盘中,否则,如果突然断电,就会丢失已输入的内容。存盘用“save”命令。

要注意的是,输入源程序时必须按规定的格式输入。

### 步骤 2: 编译

在调用编辑程序建立了一个 FORTRAN 源程序后,可调用 FORTRAN 编译程序(FOR77),对源程序文件进行编译,编译后将产生一个目标文件,对于上述 F0101.FOR 文件所产生的目标文件名是 F0101.OBJ(其中的扩展名 .OBJ 由系统自动生成)。进行编译的命令是:

```
C:\FOR>FOR77 F0101.FOR 回车
```

如果编译发现错误,则必须返回到步骤 1,重新进入编辑状态进行修改,用“save”命令保存,然后再对文件进行编译,直到编译正确通过为止。

### 步骤 3: 调用连接程序进行连接

调用连接程序(LINK86),对编译生成的目标文件(扩展名为 .OBJ 的文件)进行连接,将生成一个可执行文件,对上述目标文件生成的可执行文件名为 F0101.EXE。操作如下:

```
C:\FOR>LINK86 F0101 回车 (此处隐含的扩展名是 .OBJ)
```

如果连接时发现错误,则重新修改程序后,再进行编译和连接。

### 步骤 4: 运行程序

通过编辑建立的源程序 F0101.FOR,在编译和连接后,所产生的可执行文件

F0101.EXE 在 DOS 系统下可直接运行。键入以下命令：

```
C: \ FOR > F0101 回车 (此处隐含的扩展名是 .EXE)
```

屏幕上将出现：

```
Enter A and B:
```

这时从键盘输入 2,4 回车 ,则输出：

```
AVE = 3.000000
```

如果运行结果与预期的不符,则应对源程序进行检查、修改。修改后的源程序要重新进行编译、连接和运行,直到运行结果与预期的值一样为止。

如果程序在运行的过程中发生“死循环”或“死机”现象,可用 Ctrl 键 + C 键(即:同时按下 Ctrl 键和 C 键)强行退出运行状态,如果不见效,只能采用重新启动计算机的方法。

根据以上例子可以看出,要执行一个 FORTRAN 77 程序,必须先输入源程序,然后经过编译、连接和运行这三个过程,而且前一步的通过是后一步进行的前提。有时,为简化操作过程,可以建立如下批处理文件：

```
@ECHO OFF
DEL * .BAK
DEL * .OBJ
F77 %1.FOR
LINK86 %1
%1
```

此批处理文件将编译、连接和运行过程变为一体,假设该批处理文件名为 F.BAT,那么,通过编辑产生 F0101.FOR 文件后,执行命令: C: \ FOR \ > F F0101 回车,它相当于执行一组命令：

```
C: \ FOR \ > DEL * .BAK 回车
C: \ FOR \ > DEL * .OBJ 回车
C: \ FOR \ > F77 F0101.FOR 回车
C: \ FOR \ > LINK86 F0101 回车
C: \ FOR \ > F0101 回车
```

根据需要,可以改变批处理文件的内容。

不同的 FORTRAN 77 版本,有微小的区别,请读者使用时注意。

## 习 题

### 一、选择题

- 1-1** 一个 FORTRAN 程序由若干程序块组成,它们是
- A) 由一个主程序和至少一个子程序组成
  - B) 由任意个(含零个)主程序和若干个子程序组成

- C) 由一个主程序和任意个(含零个)子程序组成
- D) 由至少一个主程序和任意个(含零个)子程序组成

**1-2** 下列叙述中正确的是

- A) 注释行是非执行语句,不是程序的一部分
- B) 一个语句太长,可写在下一行
- C) 执行语句是使程序运行时让编译系统作相应处理的语句
- D) 一个程序块中,各类语句的位置是有规定的

**1-3** 针对以下的说法,正确的选择是

- 语句标号的大小影响程序执行的顺序
- 程序的执行绝对按语句出现的先后顺序来进行
- 不同程序单位中不可以有相同的语句标号
- 语句标号只可以选小于 5 位的正整数值

- A) 、 有错    B) 、 有错    C) 、 、 有错    D) 、 、 、 都有错

**1-4** 书写源程序时,下列叙述正确的是

- A) 续行标志使用非零字符表示
- B) 标号区为第 1 ~ 6 列
- C) 标号区不得出现标号以外的内容
- D) 语句区内书写的 SIN(X)与 S<sub>1</sub>IN(<sub>1</sub>X)等效(符号<sub>1</sub>表示空格)

**1-5** 下面叙述中正确的是

- A) 在 FORTRAN 77 程序中,各类语句的位置是任意的
- B) FORTRAN 77 程序中的任何一个语句在程序执行时,机器都会产生操作
- C) 在 FORTRAN 77 程序中的每一行上,都有标号
- D) FORTRAN 77 程序中的所有可执行语句在程序执行时,机器都会产生操作

## 二、填空题

**1-6** FORTRAN 中的最大语句标号为 1。

**1-7** 一行上的第一列字符为 2 时, FORTRAN 表示该行为注释行,注释行可出现在程序中的 3 位置上。

**1-8** FORTRAN 程序中的一行有 4 列,共分为 5 个区,其中第 6 列为续行标志区。

**1-9** 7 语句是主程序的第一个语句, FUNCTION 语句是函数子程序的 8, 9 语句只能在程序单位中最后一行。

## 第2章 数据类型及其运算

程序处理的对象是数据。FORTRAN 77 提供两类数据类型:基本类型和构造类型。

基本类型又称简单类型。基本类型中的数据项是不可再分的。基本类型中数据之间是相互独立的,互相没有内在联系。这是一些简单的数据结构。

构造类型是由已知类型,按一定规则构造而成的复合类型数据,包括数组、记录、文件和集合等。构造类型数据之间存在某种内在联系。这是一些较为复杂的数据结构。

对于同一个问题的求解,可以采用不同的数据结构和不同的算法。因此,程序不仅要描述算法,还要描述数据结构,即对所使用的数据类型进行定义和说明。

### 2.1 常量和变量的类型

FORTRAN 77 中允许使用的常量和变量的类型各有六种:整型、实型、双精度型、复型、逻辑型和字符型。

#### 2.1.1 常量

常量是指在程序中其值固定不变的一些量。

前四种整型、实型、双精度型、复型常量属于数值型常量(算术型常量,亦称常数),逻辑型和字符型常量的值不是数值,称为非数值型常量。

##### 1. 整型常量

整型常量或称整型常数或整数,它是一个正的或负的或值为零的数,整数中不能包括小数点,也不允许加入逗号,但可以有数符(正、负号),没有数符的整数作为正整数处理。例如,在 0、-1、+6、8、12.3、0.0、12,000 中,前四个数是合法的 FORTRAN 整数,其中 8 与 +8 等价,而后三个数是不合法的整数。

FORTRAN 所能表示的整数范围与具体的计算机系统有关。在一般的微型机系统中,一个整数通常用两个字节来存放,因此,整数的取值范围为 -32768 ~ 32767。

##### 2. 实型常量

实型常量也称实数。在 FORTRAN 中,实数有两种表示形式:

(1) 小数形式。即日常习惯使用的小数形式,例如,+12.3、-68.56、3.75、0.0 等。它由一个或多个数字和一个小数点组成(必须包含一个小数点而且只能包含一个小数点)。小数点前或小数点后可以不出现数字,但不能在小数点前后都不出现数字。例如,+0.8、-8.0、0.0、60.、.0 合法,而 18、36、7、.(只有一个小数点而无数字)非法。

(2) 指数形式。实数的指数形式表示法也称为科学记数法。在数学中常用指数形式