

Visual FORTRAN

● 邓巍巍 王越男 编

编程指南

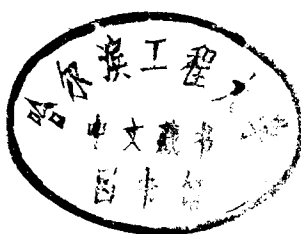


人民邮电出版社
www.pptph.com.cn

464662

Visual FORTRAN 编程指南

邓巍巍 王越男 编



00464882

人民邮电出版社

35151/22

内 容 提 要

Visual FORTRAN 是美国 DEC 公司推出的功能强大的 FORTRAN 开发工具。Visual FORTRAN 不仅继承了传统 FORTRAN 程序的高效、专业的科学计算能力,支持最新的 FORTRAN 语言标准,还可以轻而易举地创建基于 Windows 的科学计算程序和开发出友好美观的用户界面。

本书先用了一定的篇幅介绍了目前较为先进的 FORTRAN 90 语言标准,然后详细介绍了使用 Visual FORTRAN 开发科学计算程序的方法和技术。对于高级用户可以从最后一章获得更为深入的知识。

本书适合已经具有一定 FORTRAN 语言基础和科学计算经验的用户阅读,对于希望学习 FORTRAN 90 标准的读者,本书也有较好的参考价值。

Visual FORTRAN 编程指南

- ◆ 编 邓巍巍 王越男
责任编辑 陈 涛
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
北京汉魂图文设计有限公司制作
北京鸿佳印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本:787×1092 1/16
印张:22
字数:541 千字
印数:1-5 000 册

2000 年 5 月第 1 版

2000 年 5 月北京第 1 次印刷

ISBN 7-115-08530-7/TP·1636

定价:33.00 元

编 者 的 话

Digital Visual FORTRAN 是 Compaq 公司下属的 DEC 公司推出的功能强大的 FORTRAN 开发工具。Visual FORTRAN 基于 Microsoft 公司的 Developer Studio 集成开发环境, 因此除了具有 FORTRAN 语言擅长科学计算的优势以外, 还可以像 Visual C++ 甚至 Visual Basic 一样轻松开发出基于 Windows 的应用程序。Visual FORTRAN 改变了传统的科学计算程序漆黑单调的 DOS 界面, 代之以具有友好美观的 Windows 风格的用户界面, 这无疑为科学计算程序的用户提供了极大方便, 为科学计算程序本身的传播提供了条件。同时, 因为使用的是相同的开发平台, Visual FORTRAN 可以实现和 Visual C++ 的无缝连接, 极大地扩展了 Visual FORTRAN 的功能和使用范围。对于熟悉 Visual C++ 和 Visual Basic 等 Developer Studio 平台上的开发工具的用户, 可以很容易掌握 Visual FORTRAN。

Visual FORTRAN 支持酝酿了很久的 FORTRAN 语言标准: FORTRAN 90。FORTRAN 90 对以往的语言标准作了大量改动, 使之成为一种功能强大、具有现代语言特征的计算机语言, 使 FORTRAN 这种历史最悠久的高级语言又焕发出了青春。

本书共分十二章。第一章先对 Visual FORTRAN 做初步的介绍。第二章至第五章有针对性地介绍了 FORTRAN 90 语言标准的主要内容, 为以后编写 FORTRAN 90 程序打下基础; 第六章讨论用 Visual FORTRAN 进行开发基本组织单位: 项目; 第七章说明对 Visual FORTRAN 内置的文本和图形编辑器的有效使用; 第八章介绍快速开发 Windows 应用程序的方法; 第九章讨论如何在程序中使用图形元素和字体; 第十章介绍实现交互的重要途径: 对话框; 第十一章则说明使用 Visual FORTRAN 对 FORTRAN 语言和 C/C++、Basic、汇编语言进行混合语言编程所应该考虑的问题; 第十二章简要介绍包括数组高效应用、IMSL 数学和统计库、本国语言支持、多线程在内的一些高级主题。为方便读者查询, 本书附录部分把 Visual FORTRAN 语言部分按用途分类以表格的形式列出。

本书假定读者已经具有 FORTRAN 77 标准的语言基础, 并动手编写过一些 FORTRAN 程序。此外, 还最好了解 Windows 编程的基本概念。

本书由方可燕策划, 主要由邓巍巍和王越男编写, 向军审校, 其他参加编写和提供资料的还有马利勇, 董旭, 杜以恒, 桂萌, 付胜利, 张大为, 赵汝庄, 李明东, 郝春强, 周兆确, 董朝旭, 何小威, 唐家才, 许刚, 王文浩, 段小华, 邓增涛, 叶青, 毛霞等。

尽管作者使用和研究 FORTRAN 语言多年, 但 Visual FORTRAN 和 FORTRAN 90 都是较新的内容, 难免认识不够深入, 加之时间仓促, 书中错误和疏漏之处在所难免, 希望广大读者不吝批评指正。

编 者

2000 年 3 月

目 录

第一章 Visual FORTRAN 初步	1
1.1 重新认识 FORTRAN	1
1.1.1 FORTRAN 语言的发展简介	1
1.1.2 FORTRAN 90 语言标准的新特性	2
1.1.3 FORTRAN 语言与其它语言的比较	3
1.2 Visual FORTRAN 简介和安装	5
1.2.1 Visual FORTRAN 简介	5
1.2.2 Visual FORTRAN 5.0 的特性	6
1.2.3 Visual FORTRAN 5.0 的安装	7
1.3 Microsoft Developer Studio 开发环境	10
1.3.1 Microsoft Developer Studio 开发环境简介	10
1.3.2 工具栏和菜单	11
1.3.3 环境窗口	13
1.3.4 工作空间 (Workspace) 窗口和输出 (Output) 窗口	14
1.4 在线帮助	16
1.4.1 使用 InfoView	16
1.4.2 使用上下文相关的帮助	19
1.4.3 其它帮助途径	19
1.5 Developer Studio 与 Web	21
第二章 FORTRAN 90 基础知识	23
2.1 字符集	23
2.2 程序结构	24
2.2.1 程序单元	24
2.2.2 语句	25
2.2.3 名称	27
2.2.3 关键字	28
2.3 表达式	28
2.3.1 内部操作符	28
2.3.2 创建表达式	30
2.3.3 数值表达式	30
2.3.4 字符表达式	31
2.3.5 关系表达式	32
2.3.6 逻辑表达式	33
2.4 源程序书写格式	35

2.4.1	概述	36
2.4.2	自由格式	37
2.4.3	固定格式和 Tab 格式	38
2.4.4	所有格式都适用的格式	39
第三章	数据类型	40
3.1	概述	40
3.2	内部数据类型	40
3.2.1	整型数据	42
3.2.2	实型数据	42
3.2.3	复型数据	43
3.2.4	字符型数据	44
3.2.5	逻辑型数据	47
3.3	派生数据类型	48
3.3.1	派生数据类型	48
3.3.2	派生数据类型的缺省初始化	51
3.3.3	派生类型的值	52
3.4	数据属性	52
3.4.1	参数 (PARAMETER) 属性和语句	53
3.4.2	公共 (PUBLIC) 与个别 (PRIVATE) 属性和语句	54
3.4.3	保存 (SAVE) 属性和语句	54
3.4.4	静态 (STATIC) 属性和语句	55
3.4.5	自动 (AUTOMATIC) 属性和语句	56
3.4.6	用编译器指令指定属性	56
3.5	数组和指针	57
3.5.1	数组的性质和定义	57
3.5.2	数组元素和数组片段	61
3.5.3	数组赋值	65
3.5.4	数组操作	67
3.5.5	内部数组操作函数	68
3.5.5	指针	73
3.5.6	数组与指针的动态联合	75
3.5.7	DIGITAL FORTRAN 指针	78
第四章	程序单元和块结构	81
4.1	概述	81
4.2	主程序	82
4.2.1	主程序格式	82
4.2.2	程序的执行	83
4.3	模块	84
4.3.1	概述	84

4.3.2 模块的定义	85
4.3.3 模块的引用 (USE 语句)	86
4.4 过程	88
4.4.1 外部过程	89
4.4.2 块数据程序单元	89
4.5 过程接口块	90
4.6 作用范围	92
4.6.1 名称的范围	92
4.6.2 解决过程引用问题	95
4.7 联合	97
4.7.1 参数联合	98
4.7.2 使用联合	99
4.7.3 宿主联合	99
4.8 可执行结构和可执行块	100
4.8.1 概述	100
4.8.2 结构命名	101
4.8.3 IF 结构	101
4.8.4 CASE 结构	102
4.8.5 DO 循环控制	104
4.9 分支选择	107
4.9.1 GOTO 语句	107
4.9.2 CONTINUE 和 STOP 语句	108
4.10 递归过程	108
4.10.1 递归函数	108
4.10.2 递归子程序	109
第五章 输入输出	111
5.1 文件、设备和输入输出硬件	111
5.1.1 逻辑设备	111
5.1.2 文件	114
5.1.3 输入输出硬件	120
5.2 输入输出编辑	121
5.2.1 I/O 列表	122
5.2.2 I/O 编辑的方法	124
5.2.3 格式化 I/O	125
5.2.4 可重复编辑描述符	127
5.2.5 不可重复编辑描述符	131
5.2.6 直接列表 I/O	134
5.2.7 名称列表 I/O	138
5.3 输入输出语句	142

5.3.1	输入输出语句概览	142
5.3.2	I/O 语句说明符	143
第六章	使用项目进行工作	147
6.1	运行第一个程序	147
6.1.1	打开一个存在的工程	147
6.1.2	建立和执行项目	148
6.2	Visual FORTRAN 的项目	149
6.2.1	项目中的文件	149
6.2.2	Visual FORTRAN 项目的类型	150
6.2.3	Visual FORTRAN 项目的配置	152
6.2.4	创建 Visual FORTRAN 工作空间和项目	153
6.3	编写程序的一般步骤	155
6.3.1	新建一个工程	155
6.3.2	向项目中添加文件	155
第七章	使用编辑器提高效率	158
7.1	前言	158
7.2	文本编辑器	158
7.2.1	启动文本编辑器	159
7.2.2	文档	159
7.2.3	文本搜索	165
7.2.4	定制编辑器	167
7.3	图形编辑器	169
7.3.1	位图, 工具栏和光标	169
7.3.2	启动图形编辑器	170
7.3.3	图形编辑器的工具栏	171
第八章	使用 QuickWin	172
8.1	概述	172
8.1.1	QuickWin 的能力	173
8.1.2	QuickWin 和基于 Windows 的应用程序的比较	175
8.2	QuickWin 程序的类型	175
8.2.1	标准图形应用程序	176
8.2.2	QuickWin 图形应用程序	176
8.2.3	QuickWin 用户界面	176
8.2.4	缺省的 QuickWin 菜单	176
8.3	创建 QuickWin 窗口	177
8.3.1	访问窗口属性	177
8.3.2	创建子窗口	179
8.3.3	赋给窗口焦点和设置活动窗口	180
8.3.4	保持窗口打开	181

8.3.5 控制窗口的大小和位置	181
8.4 定义图形特性	182
8.4.1 选择显示选项	182
8.4.2 设置图形坐标	182
8.4.3 使用颜色	183
8.4.4 设置图像属性	184
8.5 显示图形输出	184
8.5.1 绘制图形	184
8.5.2 显示基于字符的文本	186
8.5.3 显示基于字体的字符	187
8.6 屏幕图像	187
8.6.1 在内存中传输图像	188
8.6.2 载入图像和保存图像到文件	188
8.6.3 从 QuickWin 编辑菜单编辑文本和图形	188
8.7 定制 QuickWin 程序	189
8.7.1 菜单程序控制	189
8.7.2 改变状态条和状态信息	193
8.7.3 显示信息框	194
8.7.4 定义关于 (About) 框	195
8.7.5 使用定制图标	195
8.8 使用鼠标	197
8.8.1 基于事件的函数	197
8.8.2 顺序函数	199
8.8.3 缺省的 QuickWin 处理	200
8.9 增强 QuickWin 应用程序	200
第九章 图形和字体	201
9.1 使用图形模式	201
9.1.1 检测当前图形模式	201
9.1.2 设置图形模式	202
9.1.3 编写图形程序	202
9.2 添加颜色	209
9.2.1 颜色混合	209
9.2.2 VGA 颜色面板	211
9.2.3 使用文本颜色	212
9.3 坐标系	212
9.3.1 文本坐标系	212
9.3.2 图形坐标	213
9.3.3 实坐标例程序	216
9.4 可用字型	222

9.5 使用字体	222
9.5.1 初始化字体	222
9.5.2 设置字体和显示文本	223
9.6 字体示例程序	224
第十章 使用对话框	226
10.1 使用资源编辑器设计对话框	226
10.1.1 设计对话框	226
10.1.2 设置控件属性	233
10.1.3 包含文件	234
10.2 编写对话框程序	234
10.2.1 初始化并激活对话框	234
10.2.2 对话框反馈例程	236
10.3 对话框函数	238
10.4 对话框控件	239
10.4.1 控件索引	240
10.4.2 对话框控件的可用索引	241
10.4.3 指定控件索引	242
10.5 使用对话框控件	243
10.5.1 概述	243
10.5.2 静态文本	244
10.5.3 编辑框	244
10.5.4 分组框	245
10.5.5 复选框和单选框	246
10.5.6 按钮	247
10.5.7 列表框和组合框	247
10.5.7 滚动条	251
10.5.8 设置返回值和退出	251
第十一章 混合语言编程	253
11.1 概述	253
11.2 混合语言问题	253
11.2.1 调整混合语言中的调用约定	254
11.2.2 调整混合语言编程中的命名约定	260
11.2.3 定义 FORTRAN 中过程的原型	264
11.3 混合语言编程中的数据交换和访问	265
11.3.1 在混合语言编程中传递参数	266
11.3.2 在混合语言编程中使用模块	268
11.3.3 在混合语言编程中使用公共外部数据	269
11.4 处理混合语言编程的数据类型	273
11.4.1 处理数字、复型和逻辑型数据类型	273

11.4.2	处理 FORTRAN 90 数组指针和可分配数组	274
11.4.3	处理 DIGITAL FORTRAN 指针	275
11.4.4	处理数组和 Visual FORTRAN 数组描述符	277
11.4.5	处理字符数组	279
11.4.6	处理用户定义类型	282
11.5	Visual FORTRAN/Visual C++ 混合语言编程	283
第十二章	高级主题	285
12.1	高效使用数组	285
12.1.1	数组整体操作	285
12.1.2	使用列为主的数组访问和存储	286
12.1.3	尽量使用 FORTRAN 90 内在数组过程	287
12.1.4	多维数组维的宽度	287
12.2	使用 IMSL 数学和统计库	288
12.2.1	从 Visual FORTRAN 中使用 IMSL 库	288
12.2.2	库命名约定	290
12.2.3	在混合语言环境中使用 IMSL 库	291
12.3	使用本国语言支持例程	293
12.3.1	概述	293
12.3.2	单字符集和多字符集	294
12.3.3	本国语言支持库例程	294
12.4	创建多线程应用程序	300
12.4.1	多线程的基本概念	301
12.4.2	编写多线程程序	301
12.4.3	编译和连接多线程程序	306
附录	Visual FORTRAN 语言简表	308

第一章 Visual FORTRAN 初步

本章介绍 Visual FORTRAN 的一些基本知识。包括对 FORTRAN 语言本身发展简史和新版本的标准特性的分析和它与其他一些工具的比较, Visual FORTRAN 简介和 5.0 版本的安装, Microsoft Developer Studio 开发环境的介绍, 如何有效使用在线帮助, 最后介绍在 Developer Studio 中对互联网的访问。

1.1 重新认识 FORTRAN

随着 FORTRAN 语言标准的沿革, 现在的 FORTRAN 90 标准与原来的各个标准有了很大的区别: 现在的 FORTRAN 已经是一种功能强大、尤其适合进行科学计算的现代语言, 我们对古老的 FORTRAN 语言应该有全新的认识。

1.1.1 FORTRAN 语言的发展简介

一套特定的编译计算机指令的规则称为编程语言。FORTRAN 是世界上最早出现的高级编程语言。FORTRAN 是 FORMula TRANslate (公式翻译) 的缩写, 所以 FORTRAN 的主要用途是做科学计算。FORTRAN 语言的思想首先由 John Backus 于 1953 年在纽约提出, 称为 FORTRAN I。第一个 FORTRAN 程序运行于 1957 年 4 月。随后, FORTRAN 的应用迅速广泛传播, 形成了许多不同的版本。较为重要和流行的是 1958 年提出的 FORTRAN II, 它较 FORTRAN I 增加了大量重要的扩充, 例如引进子函数等概念。从 1958 年至 1963 年期间, FORTRAN 语言在许多计算机上得以实现, 期间又出现了较为流行的 FORTRAN IV。但是 FORTRAN II 和 FORTRAN IV 并不兼容, 从而使 FORTRAN 语言的标准化工作提上日程。

1962 年 5 月, 美国国家标准学会 (American National Standard Institute, 简称 ANSI) 成立了相关机构来进行 FORTRAN 语言标准化的工作。1966 年 ANSI 正式公布了两个美国标准文本:

- 美国国家标准 FORTRAN (ANSI X3.9-1966), 相当于 FORTRAN IV
- 美国国家标准基本 FORTRAN (ANSI X3.10-1966), 相当于 FORTRAN II

其中美国国家标准基本 FORTRAN 是美国国家标准 FORTRAN 的一个子集, 从而实现了语言的向下兼容。1972 年国际标准化组织 (International Standard Organization, 简称 ISO) 在 FORTRAN 66 的基础上公布了 ISO FORTRAN 标准 (ISO R1539)。它描述了三种级别:

- 基本级, 接近于 ANSI X3.10-1966

- 中间级，介于基本级和完全级之间
- 完全级，接近于 ANSI X3.9-1966

美国国家标准 FORTRAN（通常称为 FORTRAN 66）建立之后受到了国际上的广泛接受，统治了几乎所有的数值计算领域。但是随着计算机软硬件技术的迅猛发展，尤其是在结构化程序设计（Structure Programming，简称 SP）方法提出以后，FORTRAN 66 日益显得不能满足需要。这是因为，FORTRAN 66 不是结构化的语言，尤其以 GOTO 语句为标志，没有很好地实现顺序、分支和循环这三种基本结构的语句。因此许多计算机厂商对 FORTRAN 66 进行了不同程度的扩充。针对这种情况，美国国家标准学会（ANSI）于 1976 年对 1966 年公布的 ANSI X3.9-1966 进行了修订，基本上把各厂商的有效功能都吸收进来，并且还增加了许多新的内容。1978 年 4 月美国国家标准学会正式公布了新的美国国家标准 ANSI X3.9-1978《程序设计语言 FORTRAN》，这就是通常称为 FORTRAN 77 的 FORTRAN 语言标准。FORTRAN 77 向下兼容 FORTRAN 66。1980 年，FORTRAN 77 被 ISO 正式采纳成为国际标准 ISO 1539-1980，该标准同样分为全集和子集。

FORTRAN 77 公布之后在国际上得到了广泛应用，多数计算机系统都配备了 FORTRAN 77 编译程序，国内外关于 FORTRAN 77 的教材和参考资料也相当丰富。但是，FORTRAN 77 还不是完全结构化的语言。国际上为了推动 FORTRAN 语言的结构化和现代化作了不懈的努力。经过长时间酝酿，1991 年 5 月通过了研制期间称为 FORTRAN 8x 的 FORTRAN 90，美国编号为 ANSI X3.198-1991，ISO 编号为 ISO/IEC 1539:1991。新标准中增加了许多新的特性和功能，其中值得一提的是由我国计算机和信息处理标准化技术委员会程序设计分会提出的多字节字符集数据类型及相应的内部函数。这一数据类型的增加为非英语国家在使用计算机方面提供了极大的方便。之后不久又出现了 FORTRAN 95，并且更新的标准又在积极准备之中。本书应用的主要是 FORTRAN 90 标准。

由此可以看出，FORTRAN 语言虽然历史最为悠久但仍在不停地改进和发展。这也是 FORTRAN 语言保持旺盛的生命力的原因之一。

1.1.2 FORTRAN 90 语言标准的新特性

FORTRAN 90 采用仅第一个字母是大写，而 FORTRAN 77 及 FORTRAN 66 等语言的名称中只使用大写字母。FORTRAN 90 没有沿用这一传统的书写方式。

FORTRAN 90 对 FORTRAN 77 做了较大的扩充和完善，显著的扩充主要有如下七个方面：

- （1）引入数组运算；
- （2）提高了数值计算的功能；
- （3）内在数据类型的参数化；
- （4）用户定义的数据类型；
- （5）用户定义的运算与赋值；
- （6）引入模块数据及过程定义的功能；
- （7）引入指针概念。

另外，还包括了其它一些扩充。例如，改进了源程序的书写形式、引入了更多的控

制构造和递归过程、新的输入/输出功能及动态可分配数组等。

FORTRAN 90 的先进性，体现在以下几个方面：

- 增加了许多具有现代特点的项目和语句，用新的控制结构实现选择分叉与重复操作，保证了程序的结构化。
- 增加了结构块、模块及过程调用的灵活性，使源程序易读易维护。
- 吸收了 C、PASCAL 语言的长处，淘汰或拟淘汰 FORTRAN 77 中过时的语句，具有现代语言的特色。
- 在 FORTRAN 77 数值计算的基础上，进一步发展了数值计算的优势。新增了许多先进的调用手段，扩展了 FORTRAN 77 的操作功能。
- 增加了多字节字符集的数据类型及相应的内在函数，允许在字符数据中选取不同种别，在源程序字符串中可以使用各国文字和各种专用符号，对非英语国家使用计算机提供了更大的支持。

FORTRAN 77 程序仍能在 FORTRAN 90 编译系统下运行，即具有对 FORTRAN 77 的向上兼容性。

1.1.3 FORTRAN 语言与其它语言的比较

前面提到，从名称就可以看出 FORTRAN 语言是一种用途非常专一的语言。而随着其它高级语言和软件的发展，许多都能够用来作科学计算，其中较为典型的是 C/C++ 语言和 Matlab 语言。

C 语言的诞生背景是人们希望能够找到一种集高级语言特性（可读性和可移植性好）和低级语言特性（可以直接对硬件操作）于一身的语言。C 语言是由贝尔实验室的 D.M.Ritchie 于 1972 至 1973 年间在 B 语言的基础上设计出来的。随后 C 语言又作了多次改进，但主要还是在贝尔实验室内部使用，直到 1975 年 C 语言才引起人们的普遍注意。C 语言简洁、紧凑，使用方便灵活，书写格式自由，语法限制不严格，运算符和数据类型丰富，同时还是结构化的理想语言。C 语言成功地实现了它的设计初衷，从而在系统设计及应用程序设计方面得到广泛应用。

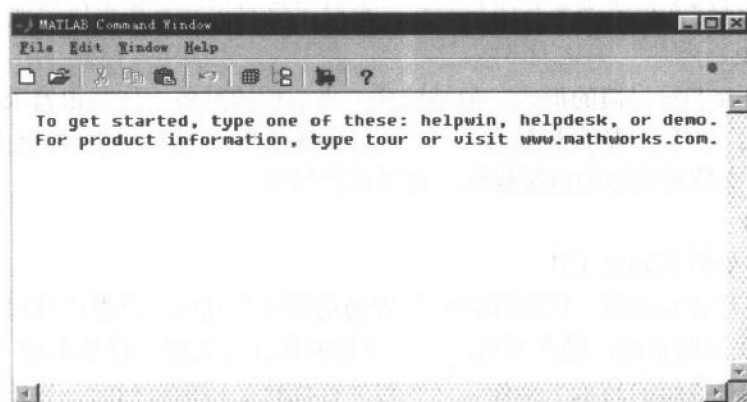


图 1.1 Matlab 5.2 的界面

Matlab 是“Matrix Laboratory”(矩阵实验室)的缩写。开始是由 Cleve Moler 于 1967 年用 FORTRAN 编写的,可以方便地调用 LINPACK EISPACK 公司所设计开发的矩阵分析软件的各种过程。现在 Windows 下的 Matlab 是 Mathworks 公司用 C 语言编写的。它是一个高度集成的系统,由于操作(尤其是对矩阵)方便,容易掌握,所以很受美国大学生的欢迎。近年我国对 Matlab 的研究也很活跃。现在 Matlab 的最高版本是 5.x 版。Matlab 5.2 的工作界面如图 1.1 所示。

与 C/C++ 语言和 Matlab 语言相比, FORTRAN 语言在科学计算方面有得天独厚的优越性。

首先, FORTRAN 语言悠久的历史是一笔宝贵的财富。在很长一段时间里, FORTRAN 语言是科技工作者唯一的选择。从 FORTRAN 语言诞生到目前近 40 年的历史过程中,在科学技术研究的各个领域积累了大量的 FORTRAN 语言程序,其中很多是经过长期实践检验证明是正确、可靠的,如果用其它语言重新编写其正确性和可靠性都还要再由实践或时间来验证。同时, FORTRAN 语言标准的历次修订都尽量保持向下兼容,这使得前人编写的 FORTRAN 程序可以不改动或只作很小的改动就可以供现在使用,这样就尽量避免重复的脑力劳动。相比之下, C/C++ 和 Matlab 比 FORTRAN 晚出现 20 年,它们所能利用和借鉴的材料是无法和 FORTRAN 语言相比的。

其次, FORTRAN 语言书写和语法要求严格,更适合严谨的科学计算领域。例如, C 语言中的数组并不提供越界检查(实际上 C 语言中数组名和指针几乎是同一个概念),这使数组的应用更加灵活。但是在科学计算方面数组的这种“散漫”的用法是相当危险的,如果允许访问到错误的内存地址,其计算的结果是不可预测的,将会在科学和工程应用上造成不可估量的损失。

第三, FORTRAN 语言可以直接对数组和复数进行运算。矩阵(包括向量)是科学计算最重要的单元,科学计算中需要对矩阵进行大量的运算。在计算机中矩阵是以数组的形式存储的。在 C 语言中要完成两个矩阵相加(如矩阵 A 加矩阵 B)要通过两个循环语句来实现,如果编成子函数调用形式也是类似

```
add_matrix (A , B , Result , matrix_line , matrix_column );
```

的有 5 个形式(相加的矩阵 A、B、存放结果的矩阵 C 及其行数和列数)繁琐的形式。在 C++ 中,可以通过定义矩阵类及其成员函数,再对运算符进行重载可以实现

```
Result=A+B;
```

这样的简洁、符合书写习惯的形式。但是这样一来前期的准备工作就很繁琐,并且对矩阵的初始化变成了对矩阵类的初始化,工作量不减反增。复数的情况也是类似的。而 FORTRAN 语言支持对数组的直接操作,可以直接使用

```
Result=A+B
```

的形式而不需作如何初始化工作。

第四,在并行计算方面, FORTRAN 语言仍是不可替代的。巨型计算机的实现,包括 IBM 的深蓝,我国的银河、曙光系列,无一不是依靠并行处理。近年来对 FORTRAN 的扩展中有很多是对数组的高水平操作,这些对并行优化是特别有利的。另外还有对共用存储器并行系统的扩展(PCF)和对串并行系统的数据并行扩展(HPF)。它们都使得 FORTRAN 在并行计算领域独领风骚。

第五, FORTRAN 语言是一种编译语言, 运行速度快。Matlab 语言之所以能够流行, 很大程度上得益于它对矩阵运算的简化。但它是类似于 Basic 语言的一种解释语言, 这使得 Matlab 语言的循环效率非常低。所以在 Matlab 中如果要大量使用循环就不得不调用 C/C++ 和 FORTRAN 程序。科学计算中普遍存在大量的循环, 这使得 Matlab 语言的应用受到很大制约, 而 FORTRAN 语言则显得得心应手。

最后, FORTRAN 语言自身仍在不断完善和发展, 功能不断增强。作为历史最长的高级语言, FORTRAN 语言自身有很多早期历史的痕迹, 如书写格式近乎刻板、不是结构化语言、输入输出简陋、命名的隐式规则等。尽管其中很多是受当时硬件条件制约的结果, 但是如果不加以改进仍会被历史淘汰。FORTRAN 语言经过几次标准修订淘汰了部分过时的语言和特性(尽管还保持兼容)并增加了符合需要的现代特性, 顺应了时代的潮流。

从这些比较中可以看出, FORTRAN 语言具有 C/C++ 语言编译语言的特点又具有 Matlab 语言对矩阵操作简洁直接的特点, 并且本身语法严谨, 底蕴丰富, 包含了现代语言的特征, 是科学计算的首选语言。

1.2 Visual FORTRAN 简介和安装

本节介绍 Visual FORTRAN 的由来, 版本特性和安装所需的条件以及步骤。

1.2.1 Visual FORTRAN 简介

Visual FORTRAN 的前身是微软的 Microsoft FORTRAN PowerStation 4.0。成立于 1975 年的微软(Microsoft)公司是世界范围内的个人计算机软件业的领导厂商。Microsoft 的产品线很宽, 如 Microsoft DOS、Windows、Windows NT 操作系统, Microsoft Office 系列, Microsoft Visual Studio 系列等。其完整全面的产品线使 FORTRAN PowerStation 4.0 能以 Microsoft Developer Studio 为集成开发环境。FORTRAN PowerStation 4.0 的功能十分强大, 它的主要特性有:

- 全面支持 FORTRAN 90 语言标准;
- 对 FORTRAN 语言的丰富扩展;
- 丰富的在线文档;
- 完全支持与 Microsoft Visual C++ 的混合语言编程;
- 与 Microsoft Visual Basic 和 Office 协同工作;
- 支持 Windows 编程(QuickWin)。

为了将生产开发工具的主要精力集中在系统开发方面, 1997 年 3 月, 微软和 DEC (Digital Equipment Corp, 即数据设备公司)达成协议, 微软不再销售和开发 PowerStation 4.0, 并且授权 DEC 提供其后继版本 Digital Visual FORTRAN 5.0 (也可称为 Visual FORTRAN 5.0), 微软推荐 PowerStation 4.0 的用户升级到 Visual FORTRAN 5.0。

DEC 的高质量 FORTRAN 编译器在全世界享有盛誉, 并在高性能科学及工程计算方面

拥有世界领先的技术。DEC和微软的这次合作使得两家公司优势互补，其第一个产品Digital Visual FORTRAN 5.0成为继FORTRAN PowerStation 4.0之后又一个功能强大的FORTRAN编译器。

1998年1月26日，Digital公司和Compaq公司同意进行当时计算机工业史上最大的一次合并。根据协议条款，Digital公司将成为Compaq公司的全资子公司，创建一个年收入超过370亿美元的公司。于是Digital Visual FORTRAN更名为Compaq Visual FORTRAN。现在，Visual FORTRAN的最新版本是Compaq Visual FORTRAN 6.1。本书主要介绍较常见的Digital Visual FORTRAN 5.0。

Digital Visual FORTRAN 5.0分为两种版本：

- Digital Visual FORTRAN 5.0 标准版

标准版包括在Intel CPU和Windows 9x、Windows NT上运行的Digital Visual FORTRAN（DVF）编译器，DVF程序库和Developer Studio。

- Digital Visual FORTRAN 5.0 专业版

专业版除了包含标准版的所有内容以外还包括在Digital Alpha CPU和Windows NT上运行的Digital Visual FORTRAN（DVF）编译器，DVF程序库和Developer Studio。另外还包括在Intel和Alpha CPU上都可使用的IMSL数学和统计程序库。

其中的FORTRAN 90优化编译器与在Digital UNIX和OpenVMS上运行的Digital FORTRAN编译器是由同一产品派生出来的。除了包含了一般的Digital FORTRAN语言扩展之外，FORTRAN 90编译器还与Microsoft FORTRAN PowerStation兼容，并且支持FORTRAN 95的语言特性。

1.2.2 Visual FORTRAN 5.0 的特性

- 全面支持FORTRAN 90语言标准，支持大部分其他平台上供应商提供的FORTRAN语言扩展，还包含了FORTRAN 95的标准特性。
- 使用Microsoft Developer Studio集成开发环境，用户可以应用Windows的丰富特性使开发速度更快、效率更高。
- 支持COM（Component Object Model，即组件对象模型）和自动对象（模块向导）。
- 专业版包含了分析和处理科学和商业应用中统计数字的IMSL数值库。
- 支持在命令行窗口中使用命令行界面，并且用户可以定制习惯的命令行工作环境。
- 提供和运行在Digital UNIX和OpenVMS Alpha系统上的Digital FORTRAN以及Microsoft FORTRAN Powerstation 4.0兼容的语言扩展。
- 完整详尽的在线帮助系统。
- 完全支持Visual FORTRAN和Visual C++，Visual J++，Visual Basic，和Microsoft MASM（x86系统的汇编器）语言之间的混合语言编程。
- 与Microsoft Visual Basic和Office协同工作，例如可以用Visual Basic创建图形用户界面程序，用Visual FORTRAN从已经存在的FORTRAN源代码创建的动