

朱文达 编著

计算机语言

**A
P
L**

**A
P
L** 及其应用

**A
P
L**

**A
P
L**



科学出版社

计算机语言 APL 及其应用

朱文达 编著

科学出版社

1987

内 容 简 介

APL 语言是当今世界上广泛流行的一种计算机语言,其算法新颖、结构严谨、语句丰富、语义清晰。本语言便于进行复杂的科学计算。由于 APL 语言还是一种会话型语言,包括了较强的行处理和字符处理能力,因此除科学计算外,它还被广泛地用于系统设计、教育、商业、管理等领域。

本书由浅入深全面地介绍了 APL 语言的各种成分和特点,系统地叙述了其主要的功能和用法及其应用领域。书中通过较多的例子对上述进行了说明。每章末均附有习题,以便于读者思考和练习。

本书可供从事计算机应用的科技人员阅读,也可供具有中等以上文化程度的人员自学。

计算机语言 APL 及其应用

朱文达 编著

责任编辑 那莉莉

科学出版社出版

北京朝阳区门内大街 137 号

中国科学院印刷厂印刷

新华书店北京发行所发行 各地新华书店经售

1987 年 1 月第 一 版 开本: 787×1092 1/32

1987 年 1 月第一次印刷 印张: 11 1/4

印数: 0001—5,500 字数: 252,000

统一书号: 15031·770

本社书号: 4843·15—8

定价: 2.10 元

8150178

前 言

APL 语言是继 FORTRAN, ALGOL60, COBOL 之后出现的计算机语言,它是当今世界上流行的计算机语言之一。

与其它流行的语言相比, APL 语言具有设计新颖、结构严谨、句法简单、语义清晰等优点。由于这一语言的语句采用了数学表示法,因此,用它进行程序设计,表达、计算、检查均方便,并易于书写和阅读。

语言中有着众多的数学运算和数学函数,除了可用标量作自变量和结果外,还可用向量、矩阵和数组作自变量和结果值。这样,当遇到较为复杂的数学问题时,就可以用简单的方式把它准确地表示出来。除此之外,本语言中还包括了较强的行处理和字符处理能力。因此,除了科学计算外, APL 还被广泛地用于系统设计、教育、商业、行政管理等领域。

APL 语言是一种会话型语言,它具有较多的操作命令,只要配上相应的命令解释系统和分时工作的子系统,就可使多个用户同时在各自己的终端上使用 APL 语言解决自己的问题。

由于 APL 具有较强的计算能力和通信能力,所以把它用到数据库管理和数据库应用方面更能显示其威力。

APL 是英文“A PROGRAMMING LANGUAGE”的缩写,即一种程序设计语言。该语言是 Kenneth E. Iverson 在 1957 年提出的。当初只限于用它来描述各种数学应用中的算法。它发展成为计算机程序设计语言是在 1962 年以后的事。二十年来, APL 的应用越来越广,其结构越来越合理,功

能越来越完善。为了表彰 Iverson 发展 APL 语言所作出的卓著贡献,在 1979 年的 ACM 年会上把图灵奖授予了 Iverson,由此足见 APL 语言在计算机领域中的价值。

本书全面地介绍了语言的成份和特点,系统地叙述了语言的主要功能和用法,并通过较多的例子进行说明。书中还对语言的工作环境、系统命令及程序设计方法作了简略的介绍。每章末附有思考和练习题,便于读者练习和深化本章的内容。

本书是在编者“APL 语言使用”的培训班所写讲义的基础上几经补充、修改、编辑而成的。在本书写作过程中,得到了有关领导和同志的鼓励和支持。在写成初稿后,承蒙方钟杰同志审阅,并提出了宝贵意见。全部稿件完稿后,陈东岳同志作了详尽的审阅,并提出许多修改意见。书中的部分程序由罗勋和张玉爱等同志在计算机上调试通过。对上述人员在此一并表示感谢。

限于编者的经验和水平,本书难免存在错误和不当之处,敬请读者批评指正。

编著者

引 言

人们根据自己的意图创造并研制了计算机，然后人们又要利用计算机来实现自己的意图。那么，人们如何把自己的意图告诉计算机，计算机又怎样识别它、处理它、实现它，并把实现的信息返回给人呢？以上这些问题正是计算机的工作原理及其过程所要回答的。

人可识别的信息是文字、语言和图象，而计算机能识别的信息一般是二进制码。为了实现两者之间的通信，首要的任务是需完成对信息的转换。这一工作要由人和计算机共同完成。目前人们正在为实现直接用图象、文字、声音同计算机通信而进行努力，并取得了一定的成效。这样，越来越多的信息转换工作就可由计算机完成。但是目前广泛使用的还是一些传统的转换方法，即人们将一系列计算机可识别和接受的符号和数字按规定的格式写出，并将这些信息穿孔在纸带、卡片上，或录制在磁介质上，然后通过相应的装置送入计算机；计算机则把处理完的信息通过打印机、穿卡机、磁介质等设备打印、穿孔、录制在相应的介质上传递给人。我们把用于向计算机表示人们意图的一系列指令称为程序，编写程序所用的语言称为计算机程序设计语言。程序设计语言种类繁多，有相应于特定机器、由指令代码构成的机器语言和汇编语言，有相应于实现人的特定意图的高级语言，如 FORTRAN, COBOL, ALGOL, BASIC, PL/I 等。这些高级语言各有所长，它们是目前人们所广泛使用的一些程序设计语言。在用机器语言和汇编语言实现人的意图的过程中，需要人做大量的信息转换

工作，而高级语言则是把人在低级语言中所要做的信息转换工作移交给机器去做。随着面向问题语言的出现和模式信号识别装置的使用，需要人去做的信息转换工作越来越少了，这是广大用户所希望的，同时，也正是广大计算机工作者的奋斗目标。

用户热切希望的还有一点，这就是能在自己的办公室、实验室中如意地使用计算机来帮助自己的工作，解决用户想要解决的问题。如果每个办公室均装备一台计算机，这从经济观点和空间观点以及目前的计算技术来衡量当然不是不可能的，但从共享数据这一角度来看，这就是不可能的。计算机网络和终端的出现，较好地解决了这个问题。如在办公室里将一台终端同中心计算机连上，那么用户就不需要为一个小题而延误许多时间，也不需要一次又一次地跑到计算中心去送程序、取结果。此时只要打开终端，用户就可随意使用中心计算机及存取数据。

本书介绍的 APL 语言就是一种接近于面向问题的解释执行类型的高级语言，同时它又是一种在终端上使用的人机交互式语言。因此，APL 语言广泛地用于工业、科学技术、行政管理、企业管理、文化教育等领域。

APL 语言具有功能强、结构简明的特点，程序员可使用其大量的函数直接按习惯的数学形式来表达自己的思想和编制程序。这是 APL 语言的特色之一。特色之二是此语言对数组计算十分方便。在 APL 中可用单个语句实现的功能，在其它语言中必须用许多语句才能实现。因此，APL 语言缩短了编写程序的时间，并减少了许多错误源。特色之三是此语言提供的功能易于输入输出语句的书写。特色之四是如发现程序设计错，可给出简单明了的信息。所有这些特性使 APL 成为能适合于分时交互使用的较为理想的语言。

APL 语言还在不断发展中,结构也越来越趋于合理,功能越来越完善.现有 APL 语言的各种文本,如 TTPL, APL/360, VSAPL, APLSV 等.它们之间存在着某些差异,但只要掌握其中一种,其余的也就迎刃而解了.本书着重介绍 APL 的一般功能,不涉及各文本间的差别.

本书共分十一章.第一章主要介绍 APL 语言的工作环境及其资源,通信过程的建立和撤销,通信信息的传送、存贮、处理及保护等一般概念.

第二、四、五、七章介绍以各种不同信息类型作为自变量的所有 APL 函数的定义、功能、规则及其用途.其中第二和第四章介绍以标量或矢量为自变量的标量函数.第二章中涉及到的标量函数包括算术函数和超越函数.第四章涉及到的标量函数包括关系函数和逻辑函数.第五章介绍以标量或矢量为自变量的所有 APL 的混合函数.第七章介绍以二阶和二阶以上的数组作为自变量的所有标量函数和混合函数.

第三章主要描述定义函数的结构、功能和书写、编辑、修改定义函数的过程及方法.为了照顾上下文的内在关系,把随机输入输出函数、执行函数和格式函数也列为本章的内容.

第六章叙述定义函数中实现转移、循环的方法,介绍各类中断、跟踪、停止操作在定义函数中的作用及实现方法.第八章介绍如何利用 APL 所定义的函数实现结构化程序设计,并通过实例说明程序设计的方法.第九章讲述几个典型的应用实例.第十章介绍系统变量、系统函数、共享变量的句法和功能.

第十一章逐一介绍各系统命令的定义、格式和功能,同时也介绍了内存活化区和库存区的管理方法.

目 录

引言.....	ix
第一章 概述.....	1
1.1 APL 分时系统.....	1
1.2 终端.....	2
1.3 内存工作区和程序库.....	3
1.4 字符集.....	5
1.5 通信过程.....	6
1.6 通信信息.....	9
1.7 变量.....	15
1.8 赋值.....	17
1.9 函数.....	19
1.10 表达式计算顺序.....	20
1.11 诊断信息.....	22
1.12 改正打字错误.....	23
1.13 信息保护.....	23
思考和练习.....	24
第二章 标量函数之一——算术函数和超越函数.....	26
2.1 双目函数和单目函数.....	27
2.2 标量之间的双目函数.....	27
2.3 独元素和矢量之间的双目函数.....	28
2.4 矢量之间的双目函数.....	30
2.5 单目函数.....	31
2.6 定义域限制条件.....	32
2.7 下归整与求小数函数.....	33
2.8 上归整与求大数函数.....	34

2.9	上归整与下归整的舍入误差及比较容差	35
2.10	求绝对值与残数函数	36
2.11	阶乘与组合函数	37
2.12	随机函数及随机分配函数	38
2.13	下归整、上归整及残数的性质比较	40
2.14	归并算子	40
2.15	扫描算子	43
2.16	算术函数一览表	43
2.17	超越函数	45
	思考和练习	48
第三章 定义函数		51
3.1	编写函数	51
3.2	说明信息	54
3.3	编辑修改命令	55
3.4	全局量和局部量	62
3.5	注释行	75
3.6	随机数据输入	76
3.7	随机数据输出	80
3.8	执行函数和单目格式函数	81
3.9	执行函数的应用	84
	思考和练习	86
第四章 标量函数之二——关系函数和逻辑函数		91
4.1	关系函数	91
4.2	从属函数	94
4.3	关系函数的比较容差	95
4.4	关系函数一览表	96
4.5	压缩函数	96
4.6	逻辑函数	100
4.7	扩展函数	103
	思考和练习	105

第五章 混合函数	107
5.1 规格和构造函数	107
5.2 指标检索元素函数	110
5.3 指标生成函数	112
5.4 空矢量	114
5.5 连结和拆散函数	117
5.6 筛选和筛舍函数	120
5.7 递增和递减函数	123
5.8 旋转和倒置函数	124
5.9 元素检索指标函数	126
5.10 内积算子	127
5.11 外积算子	128
5.12 译码函数	128
5.13 编码函数	130
思考和练习	134
第六章 转移、中断及跟踪	137
6.1 无条件转移	137
6.2 行标号	139
6.3 条件转移	140
6.4 循环	145
6.5 嵌套定义函数	147
6.6 函数的中断及状态指示	152
6.7 强制中断和错误中断	155
6.8 跟踪和停止	157
6.9 函数的保密性质	160
思考和练习	160
第七章 数组	165
7.1 数组的术语	166
7.2 规格和构造函数	169
7.3 轴算子	172
7.4 归并和扫描算子	173

7.5	数组计算	181
7.6	内积和外积算子	185
7.7	连结和加轴连结函数	192
7.8	筛选和筛舍函数	200
7.9	压缩和扩展函数	203
7.10	指标检索元素函数	208
7.11	倒置和旋转函数	213
7.12	转置函数	217
7.13	译码和编码函数	224
7.14	元素检索指标函数	229
7.15	从属函数	233
7.16	矩阵除	234
7.17	格式函数	238
	思考和练习	242
第八章	程序设计方法	249
8.1	等价于条件转移的控制结构	249
8.2	用循环结构求多项式的值	251
8.3	嵌套循环	253
8.4	逐次近似法	254
8.5	等价于循环的控制结构	259
	思考和练习	262
第九章	应用实例说明	264
9.1	数据分类	264
9.2	数据检索	282
9.3	数据的图形表示	292
9.4	计算机辅助教育	300
	思考和练习	307
第十章	系统变量、系统函数及共享变量	311
10.1	系统变量	311
10.2	系统函数	314

10.3 共享变量	317
第十一章 系统控制命令.....	326
11.1 终端控制命令	328
11.2 活化区控制命令	330
11.3 程序库控制命令	334
11.4 询问命令	335
11.5 工作区管理	337
附录1 错误报告.....	341
附录2 故障报告.....	344
参考文献.....	346

第一章 概 述

本章对 APL 语言作一般性的介绍, 涉及到的概念将贯穿以后各章. 如什么叫分时系统? 分时系统所需的工作环境及资源, 通信过程的开始和结束, 信息的传递、存贮、保护, 数组加工处理的规则和方法, 出现错误后的诊断信息和改正办法. 通过对诸如此类问题的介绍, 我们将逐步把 APL 的全貌揭示给读者.

1.1 APL 分时系统

在 APL 计算机系统中, 假定用户通过终端键盘按交互方式使用 APL 程序设计语言. 那么, 用户通过键盘把 APL 语句和数据打入计算机, 然后, 计算机自动地把响应后的执行结果打在打印机纸上或显示在阴极射线管的屏幕上; 另一方面, 有时计算机则主动提出某一问题送至终端, 要求用户作出回答. 人与计算机的这种一问一答的工作方式称为会话方式.

所有的 APL 会话语言都包括两部分: 一是语言本身, 二是系统控制部分. 构成语言部分的主要是 APL 语句, 构成系统部分的主要是系统命令.

对于小型计算机来说, 同一时间内只能一个人使用 APL; 在大型计算机中通常按分时方式运行 APL, 此时, 可有许多用户同时使用 APL. 对于这两种计算机所配置的 APL, 其语言部分完全是相同的, 只是系统控制部分存在着某些差异. 在分时工作方式中, 主机 (CPU) 以时间片轮转工作, 为一个用

户服务几毫秒(几毫秒或几十毫秒不等,视系统不同而异)以后,就迅速转移去为另一个用户服务。如此轮转,APL 在同一时间内可为几个用户或几十个用户服务。从微观上(从计算机的角度)看,计算机是在轮流为用户工作,而从宏观上(用户的角度)看,计算机是同时为许多用户服务。这样就使得使用 APL 的每一个用户都感觉到他在独自使用计算机,而觉察不到别人也在使用计算机。按如此方式工作的计算机系统称为**分时系统**。

在多个用户同时工作时,APL 为每个用户设置一个用户号(或叫用户帐号),以此区别不同的用户。用户还可设置口令,保护其信息,以免遭别人窃取。如果用户的信息愿意向其它用户公开,则可不设置口令,这样如其它用户需要,就可利用该用户的信息为自己服务。用户之间还可互相通信,以便协调工作。

1.2 终 端

在配置 APL 语言的计算机系统中,当操作员加载、启动 APL 系统后,用户即可在与中央机相连的 APL 终端上进行 APL 操作。用于 APL 的终端装置可以是一个带有 APL 符号键盘的显示设备,也可以是一个电传打字机,或同时由一个显示器和一台打字机构成。APL 可以采用不同型号的终端设备,但要求它们至少具备下列功能:

1. 要有输入和输出 APL 基本字符的能力。
2. 在编写的一个输入行(语句)发送至系统之前,要拥有能修改该输入行的控制。
3. 在编写好一个输入行之后,能发出一个信号,将输入行发送至系统,并促使终端托架或显示光标换行回车至新一行

的起始位置。

4. 要求有中断 APL 程序的能力。

5. 编辑输入行时,要有指示手段,以指明下一个要记入的字符位置。

6. 在用户需要穿孔纸带输入输出时,还要求终端具备纸带穿孔阅读器。

1.3 内存工作区和程序库

当终端用户一旦被系统操作员激活后,APL 系统就分配给用户一个称作活化工作区的内存空间,该区域分为两部分:一部分为系统内部操作使用,另一部分用于存放用户的各种信息,如变量、变量群、定义函数以及在计算过程中所产生的中间结果和最终结果。这些变量和定义函数的内容如果不是被同名变量覆盖,也没有被系统命令删除的话,则在现行终端的整个活化时间内,一直保留在内存工作区中,以供用户反复使用。

用户内存区中,现行的全部内容可借助系统命令保护到设置在磁盘上的属于 APL 的程序库中。APL 程序库分为两类:一类称为**公共库**,其库名用系统生成时就确定的数码来标识;另一类称为**用户库**。每个 APL 用户都有一个私有库,其库名就是用户的帐号。公共库可供所有的终端用户存取,私有库则只供本用户存取。系统规定在任何情况下都不允许往他人的私有库中存取信息。

用户把现行内存工作区保护到程序库的目的有三个:一是防止现行工作区被意外的事故破坏;二是内存工作区容量有限,容纳不了继续产生的新信息;三是现行内存工作区的信息需要永久地保存下来,以便供日后反复使用。

如何把现行内存工作区保护到程序库，用户应视情况而定。如为第一个目的所进行的分区保护，则可分期分段进行，其与 FORTRAN, COBOL 语言中建立故障恢复点、设置恢复点文件的方法类似，即保存现行内存工作区的同时，删除库中的老的内存工作区。如为第二个目的，则可按同读写文件相似的方法，实行分区的保护和加载。假如为第三个目的，则可按程序库管理的工作方法实行存取。

存入库中的内存工作区均有一个工作区名。一旦需要用到库存的内存工作区副本中的信息时，可全部或部分地把它恢复到现行活化内存工作区中来。当一用户知道另一用户的库名和内存工作区名时，可把另一用户的库存工作区的全部或部分内容恢复到自己的现行内存工作区中来。

分配给每个用户内存工作区的大小取决于所用的具体计算机系统。在采用虚拟存储器的系统中，其大小基本不受限制。在大小受限制的工作区中，当打入的信息和运算产生的瞬态信息超出区域时，用户将收到 WS FULL 的错误信息；系统停止计算。

在用户的内存工作区中有五个对计算产生影响的参数，这里介绍其中的三个，它们是：

1. 输出精度(输出结果的有效数字数)。
2. 输出行宽度(输出行的每行最大允许的字符数)。
3. 指标原点(确定与执行数组指标有关的操作所采用的下标起始原点是从 0 开始，还是从 1 开始)。

在任何给定的 APL 系统中，这些参数在用户内存工作区建立时就有标准值。如果用户不去改变它们，这些值在整个工作过程中固定不变。这三个值分别通过系统变量 $\square PP$ (输出精度)， $\square PW$ (输出宽度)和 $\square IO$ (指标原点)给出。通过打印这些变量名就可确知它们的值。典型的系统有：