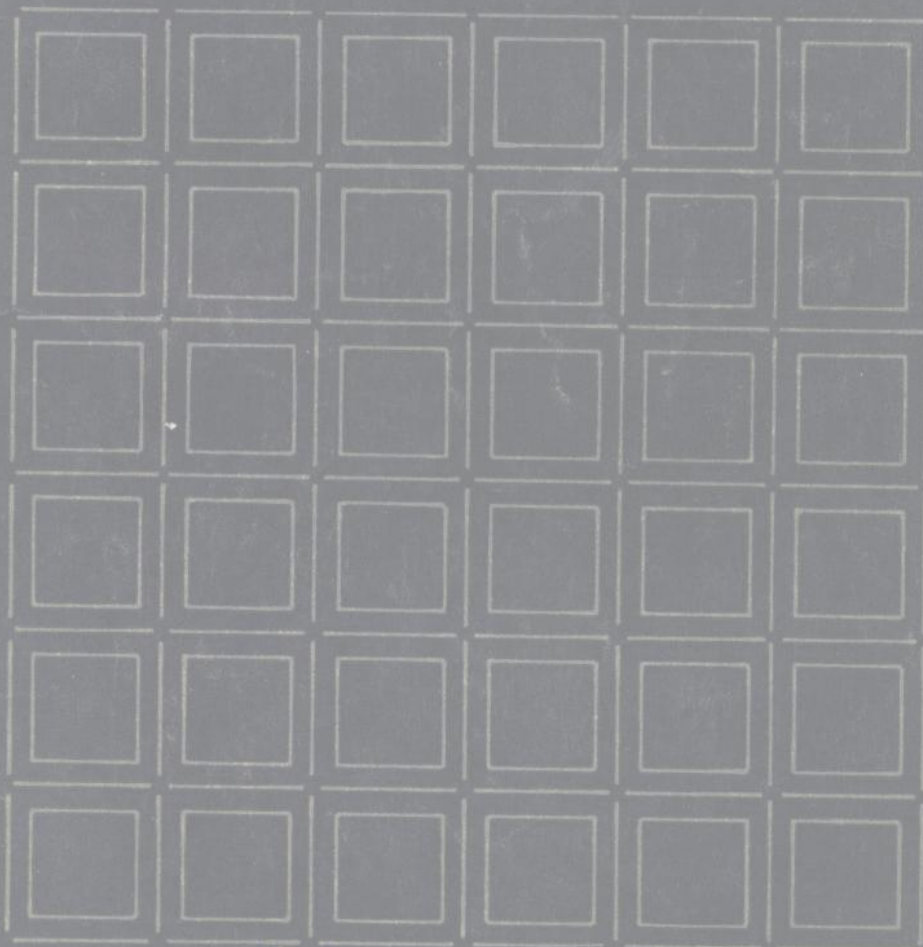


FORTRAN

语言程序设计教程

洪兴山 乐秀章 著



科学技术文献出版社

72.87221

3/5

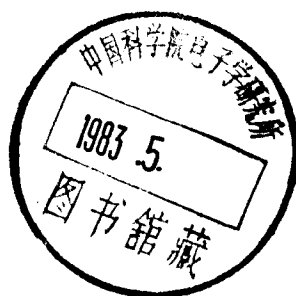
FORTRAN语言程序设计教程

洪兴山 乐秀章 著

张宗信 李秀锦 译

屈慰双 白葆林 校

沈祖恩 校



科学技术文献出版社

1983

1110649

内 容 简 介

本书的特点：语言简洁，深入浅出，所有的概念都用例题加以说明；对不具备计算机知识和较深数学基础的读者，也可通过对本书的学习了解计算机，并掌握用FORTRAN进行计算机程序设计。所以，它对初学者、自学者和改行搞计算机的人尤其适用。

全书包括十一章：一、二章介绍计算机外部设备和计算机程序设计的基础知识；三至十一章讲解ANSI标准FORTRAN；书末附有词汇表，可作为第一、二章的补充，也可作为有关计算机术语的参考工具。

本书可作大专院校计算机专业师生的参考书，也可供计算机学科的专业人员以及初学者、自学者和爱好者阅读。

FORTRAN语言程序设计教程

张宗信 李秀锦等译

科学技术文献出版社出版

中国科学技术情报研究所印刷厂印刷

新华书店北京发行所发行 各地新华书店经售

*

开本：787×1092¹/₁₆。 印张：21.5 字数：544千字

1983年1月北京第一版第一次印刷

印数：1—12,364册

科技新书目：38—65

统一书号：15176·587 定价：2.65元

译 者 话

FORTRAN是一种效率高、用途广泛的程序设计语言。它是从事科学、工程和数据处理工作人员的一种重要的工具语言。在国内亦有广泛的流行。近年来有关FORTRAN语言的书籍虽然已有不少，但以标准文本式的为多，不能满足广大学习FORTRAN语言读者尤其是自学和初学者的要求。为此，我们翻译了香港中文大学洪兴山、乐秀章先生编著的《FORTRAN语言程序设计教程》一书。该书不仅语言简洁，内容由浅入深，而且所有的概念都用例题加以说明，有助于读者对概念的理解。同时每章之后都附有一定数量的练习题，可供读者练习。所以，这本书既可作为广大不具备计算机知识和较深数学基础的读者较好的自学书，又可作为大专院校有关计算机方面的教师和学生较好的参考书。

因译者水平所限，书中错误之处恳请读者提出宝贵意见。

一九八一年十月

前 言

FORTRAN程序设计是计算机导论课程中的一个传统科目。这主要是由于高效能的FORTRAN编译器已得到广泛应用,由于FORTRAN语言有强大的解算问题的能力,以及在一般应用中FORTRAN是最早的高级语言。

这些年来出版的FORTRAN教课书大约已有数百种之多,那么为什么还要再编写这本书呢?本书的目的在于向初学者介绍那些基础的知识,以便使他们了解计算机并掌握用FORTRAN进行计算机程序设计。特别应指出的是,本书是专门针对香港学生的需要而写的。现有的许多FORTRAN入门书一般有两个缺点:书中没有足够的习题供初学的学生做练习用,而所列出的习题又常常要求学生有坚实的数学基础。而本书却不要求学生具有计算机科学方面的知识,在数学基础方面的要求也是很低的。我们希望本书将会引起那些在课堂上学习或自学FORTRAN的学生们的兴趣。

本书是根据最近几年在香港中文大学讲授FORTRAN导论课所使用的讲稿和习题写成的。书中所有的例题和程序都具有适当的难度并由学生们做过。在组织本书材料时,我们确信一本容易理解的程序设计教科书所应具有的基本前提条件是,教材的编排要合乎逻辑,阐述要清晰。这种思想再清楚不过地表现在书中各章次序的合理安排上,其着眼点是使学生逐步地有顺序地理解并实践计算机程序设计的各项技巧。全书中,在讲解每一个专题和FORTRAN程序设计概念时,都根据我们以前的教学经验,考虑了中等程度的学生能接受的难度。在这里,简单是为了便于理解,而不是为了敷衍了事。我们使用了一种分三步开展的分析方法:(1)文字描述和解释,(2)例题,(3)对例题用逻辑框图进行图解。

本书中的所有概念都用例题(超过200个)加以说明。之所以大量地使用例题是因为在讲授计算机科学导论课时,通过对学生学习情况的观察,我们深信,即“一个好的例题比一千句话还顶用”。作者确信:只有经常实际练习才能学会FORTRAN程序设计。这本书在内容安排上几乎一开头就让学生写完整的、有实际意义的程序。

第一章和第二章向学生介绍计算机外部设备和计算机程序设计的基础知识。第三章到第十一章讲解ANSI标准FORTRAN。对每种语句都做了透彻的讲解并至少用一个例子加以说明。在每一章的最后都总结了本章所学习语句的一般格式并再举一些例题。在附录中包括有一个词汇表,这有两种用途:(1)作为第一章和第二章的补充,(2)作为计算机科学和数据处理方面所使用术语的一个简便参考工具。

在每一章的后面,我们列出了许多问题和学生练习题,这些问题有针对性地概括每章的重点,它们都是经过反复推敲的。这些问题和练习分属两种不同类型,但我们有目的地使它们掺杂在一起。有一部分题目是为了突出本章的重点,另外一些是具有探讨性质的题目供给那些有兴趣的、积极性较高的学生。在各种场合我们都尽量采用这样的程序设计习题,它们要求学生能引伸和灵活运用FORTRAN程序设计的主要概念和各种关系。

这些练习题在深度和广度上都能满足那些有能力的学生的要求,但我们也注意到不使它们超过一般学生的理解水平。

在编写本书的过程中许多作家、教师、学生的意见和建议使我们受益非浅,他们的人数是这样多以至于无法一一表示感谢。许多人的建设性的建议已被尽可能适当地采用。

作者对本书原稿的审阅者邓杨先生表示感谢。我们还特别感谢史坦斯劳斯·胡博士，他对本书的内容以及材料安排上都做了不少贡献。亦特别感谢约瑟夫·杜先生，他为本书配制了所有的框图并检验了所有程序的正确性。作者还向香港中文大学计算机科学系所有工作人员致意，感谢他们的帮助与合作。

对我们的家人为我们完成此书，坚韧地承担了各种压力，在此一并表示谢忱。

洪兴山 乐秀章

1978年6月于香港

目 录

前 言

第一章 计算机简介	(1)
1.1 计算机简史	(1)
1.2 计算机的最新进展	(3)
1.3 计算机的特点	(4)
1.4 计算机的分类	(4)
1.5 计算机的应用	(5)
1.6 计算机系统的基本组成部分	(5)
1.7 输入/输出 装置	(8)
1.8 外存贮器	(9)
1.9 数据准备设备	(10)
1.10 硬件和软件	(10)
1.11 习题	(11)
第二章 计算机程序设计简介	(13)
2.1 解题步骤	(13)
2.2 程序框图	(13)
2.3 程序的编写	(19)
2.4 程序的调试和检测	(23)
2.5 例 题	(24)
2.6 FORTRAN程序设计简介	(26)
2.7 习题	(27)
第三章 算术运算	(30)
3.1 字符表	(30)
3.2 符号名字	(30)
3.3 语 句	(30)
3.4 编写FORTRAN语句	(32)
3.5 常 量	(33)
3.6 变 量	(35)
3.7 算术表达式	(36)
3.8 FORTRAN标准函数	(42)
3.9 算术赋值语句	(45)
3.10 语句小结	(47)
3.11 习题	(47)
第四章 基本的输入和输出	(52)
4.1 输 入	(52)
4.2 输 出	(57)

4.3	文字输出	(62)
4.4	格式语句的有关规则	(64)
4.5	FORTTRAN程序的结尾	(65)
4.6	一个完整的程序实例	(65)
4.7	语句小结	(68)
4.8	习题	(70)
第五章	条件控制语句	(76)
5.1	无条件GO TO语句	(76)
5.2	逻辑IF语句	(77)
5.3	循 环	(80)
5.4	算术IF语句	(82)
5.5	计算GO TO语句	(85)
5.6	例 题	(88)
5.7	语句小结	(104)
5.8	习题	(104)
第六章	DO循环	(115)
6.1	DO语句	(115)
6.2	CONTINUE语句(继续语句)	(122)
6.3	嵌套DO循环	(124)
6.4	例 题	(128)
6.5	语句小结	(142)
6.6	习题	(143)
第七章	数 组	(149)
7.1	一维数组	(149)
7.2	DIMENSION(维数)语句	(151)
7.3	类型语句	(152)
7.4	下标的形式	(153)
7.5	实例程序	(154)
7.6	二维数组	(159)
7.7	三维数组	(162)
7.8	例 题	(164)
7.9	语句小结	(171)
7.10	习题	(171)
第八章	输入输出的补充	(180)
8.1	E型区域描述符	(180)
8.2	行式打印机的间隔控制	(182)
8.3	区域分隔符	(184)
8.4	一组格式描述符	(185)
8.5	输入输出中的隐含DO循环	(186)
8.6	整个数组的输入和输出	(190)

8.7	格式控制与输入输出表的相互配合	(191)
8.8	例 题	(195)
8.9	语句小结	(199)
8.10	习题	(199)
第九章	字符处理	(209)
9.1	何勒内斯字符和常数	(209)
9.2	用H型描述符读取字符数据	(209)
9.3	用A型格式描述符来输入/输出字符串	(211)
9.4	数据初始赋值语句	(217)
9.5	字符串的比较	(220)
9.6	执行时间的格式语句	(223)
9.7	语句小结	(226)
9.8	习题	(227)
第十章	子程序	(236)
10.1	子例行程序子程序	(236)
10.2	函数子程序	(244)
10.3	子例行程序和函数的比较	(249)
10.4	子程序的优点	(249)
10.5	有关子程序的补充例子	(250)
10.6	算术语句函数	(259)
10.7	语句小结	(262)
10.8	习题	(264)
第十一章	子程序的其他特点	(274)
11.1	COMMON BLOCKS (公共程序块)	(274)
11.2	在执行期间数组维数的表示	(280)
11.3	外部语句	(281)
11.4	在各程序单元间语句的次序和关系	(283)
11.5	语句小结	(283)
11.6	习题	(284)
附 录: 词 汇		(288)

第一章 计算机简介

虽然许多学生很快就学会了怎样使用计算机,但实际上他们对这种现代化的机器了解甚少。在这一章中,我们简要地介绍一下计算机的发展历史,并介绍一些计算机的重要概念和术语。我们认为有关计算机的一般知识能够帮助学生了解,借助于这些机器到底能做些什么工作。

1.1 计算机简史

最近二十年来计算机的飞速发展是一个惊人的创举,它说明在科学知识和工艺技术能力都已成熟并且社会又对此提出需要时将会发生什么事情。但计算机的发展也和科学方面的其它进步一样,在其后面有很长的一段历史。

算盘 (古代)

算盘是最早用来进行计算的装置之一,它由串在若干根棍上的许多珠子构成,而这些棍安装在一个长方形的框架里。一个水平杆把这些珠子分成上、下两部分,上边的每个珠子代表五个单位,下边的每个珠子表示一个单位,每个棍代表一个数位,比如说十位、百位等等。在算盘上计数时,把棍上的一定数量的珠子拨向水平杆。如果在某一棍上所要表示的数超过9时,则下一个数位(即10)的表示法是:在左边的棍上进位,并使原来棍上的所有珠子拨离水平杆,这表示零。算盘可用来做加法、减法、乘法(垒加)和除法(垒减)。算盘是数字型计算装置的一个例子。

对数和对数计算尺 (1614, 1630)

计算尺是模拟型计算装置的例子。在这里,数是用尺上的一段距离表示的,而这段距离的长度由此数的对数确定。例如,两个数相乘是通过把它们在尺上所对应的距离相加而完成的。

机械式计算装置 (1642, 1672)

法国哲学家和数学家波莱斯·巴斯噶创制了世界上第一台机械式加法装置。在他的机器里,齿轮上的每个齿分别代表0到9这十个数中的一个,再把这样的齿轮连成一组。当一个齿轮转过了代表9的那个齿时,左边的那个齿轮自动地转动一齿(或一位)。换句话说,第一个齿轮上的十个齿可用下一个齿轮的一个齿替换。这是第一台能够把十进位到下一位的计数装置。因为齿轮既可正转又可反转,所以原则上它既可用于计算加法,又能计算减法。

1672年德国数学家哥特弗雷德·莱布尼兹对巴斯噶机器作了改进,他设计的机器能用垒加的方法计算乘法,用垒减的方法计算除法。巴斯噶和莱布尼兹的工作在历史上第一次证实自动数字计算这一伟大思想,即认为算术运算是可以用机器进行的。

巴贝奇的分析机 (1812—1834)

英国剑桥大学数学教授查尔斯·巴贝奇的工作使计算机的发展向前推进了一大步。1833

年巴贝奇为分析机设计了方案。机器有一个保存数的存贮器，一个用作中央算术装置用来进行计算的机械装置，和一个指挥机器操作的操作员。在这个机器里计算顺序由穿孔卡片控制，计算结果输出在一块铜板上，答案还可以自动地反馈到机器中，这样，不需要操作员的参与，就可继续进行进一步的计算。在采用前推或后退一定数目卡片的简单设计方法中，初步包含有转移这一概念。数的精确度为五十位，而机器的存贮装置可容纳50,000个数位。在机器的说明中指出了机器的运算速度，加法计算用一秒钟，两个50位的数字相乘大约需要一分钟。不幸的是，这个机器一直没有做成，因为他的设计超出了当时的技术水平。尽管如此，第一次提出数据存贮、用编程序进行顺序控制以及自动读数等思想的荣誉应归于巴贝奇，这些思想是现代计算机的基本原则。

布尔代数 (1854)

英国数学家乔治·布尔创造了用数学符号表示逻辑语句的方法，使用这些符号和若干规则就可以确定一个语句逻辑上是真是假。他的方法当时并未被广泛接受，一直到本世纪他的这种思想才被应用到计算机的设计上。

可供机器阅读的穿孔卡片 (1880—1890)

1886年，美国统计学家和发明家赫尔曼·霍尔瑞斯在加工1880年的“美国人口普查”中的数据资料时，创建了用穿孔卡片代表数据的方法。霍尔瑞斯所制作的能阅读卡片的机器用电来感知卡片上的孔，这样就能读出统计资料中的数据。以后，其它人建造一些能够复制穿孔卡片、分类、排序，甚至能制表的机器。用可拆卸的插头板控制这些机器的操作，一台机器所能做的工作由这个插件上的导线的连接方式来确定。

开关逻辑 (1938)

克劳茨·先农用布尔代数系统地表示复杂的开关网络。先农这一工作使后来在电子计算机中所遇到的那些电路设计的教学和研制工作大大简化。

Mark I (1937—1944)

从1937年起，哈佛大学教授霍华德·艾肯开始建造一个应用当代技术和穿孔卡片相结合的自动计算装置，在一些研究生和国际商业机器 (IBM) 公司工程师的帮助下，这一工作于1944年完成。做成的这台机器被称作Mark I 数字计算机，机器的内部操作由电磁继电器自动控制，算术计算器是机械式的。由此看来，Mark I 不是电子计算机而是一个机电式计算机。Mark I 的计算精确度达到二十三位数，除了能够进行四种基本算术运算之外，它还能查阅由前面的计算结果所构成的表。加法和减法运算恰好用0.3秒的时间，乘法大约需要4秒钟，除法最多需要16秒钟。Mark I 大约有5吨重，长51英尺，高8英尺。在许多方面我们可以说Mark I 实现了巴贝奇的梦想，并是自动电子计算机的先驱。

ENIAC (1943—1945)

电子数字积分器和计算器 (ENIAC) 是宾夕法尼亚大学穆尔电力工程学院的 J. 普列斯普尔·艾克特和约翰·W. 英希莱设计的，于1945年完成。ENIAC大约重30吨，占地大约1500平方英尺，它使用18,000个电子管来代替Mark I 所使用的电磁继电器。由于用电子管

代替了动作缓慢的电磁继电器，ENIAC每秒钟可进行300次乘法运算，它一天就可以做完用手工300天才能做完的工作。但是，这个机器没有内存贮器。它必须通过开关和插头从外部接受指令，这是一个很大的限制。ENIAC主要被用来计算弹道表。它通常被认为是第一台电子计算机，但是把它看做一个过度型的机器可能更合适些。

存贮程序计算机（1945—1951）

1945年，普林斯顿大学高级研究中心的著名数学家封·诺伊曼在一篇论文中提出：（1）在计算机的制造中可以使用二进制编码系统；（2）计算机指令以及由它加工的数据可以存贮在机器的内部。这两条建议成了计算机设计原则的基本部分。体现这种想法的第一台计算机是电子延迟存贮自动计算机(EDSAC)，它于1949年在英国的剑桥完成。EDSAC的指令存贮在机器内部，并以程序形式提供执行，于是存贮程序计算机就问世了。1952年，宾夕法尼亚大学一研制组的成员完成了另一台存贮程序计算机——离散型变量电子自动计算机(EDVAC)。第一台在商业上使用的计算机是通用自动计算机(UNIVAC I)。与以前的计算机不一样，它用来对非科学方面的数据进行加工。UNIVAC I是在1949年由莱明顿·兰德制成的，它使用磁带进行输入和输出，而在这之前，计算机是用比这慢得多的穿孔卡片和纸带进行输入输出的。UNIVAC I还是第一台除能接收和加工数字数据外，还能接收和加工文字数据的计算机。第一台UNIVAC I计算机于1951年安装在美国联邦调查局内。

1.2 计算机的最新进展

电子计算机在持续不断地向前发展。计算机可分成几代，“代”这个词的含意是，每一代计算机与上一代计算机相比较，都在设计上发生了重大变革。

第一代计算机从1951年延续到1959年，这一代计算机使用电子管，机身庞大，需要大量的空气调节，并且耗电量很大。它们的计算速度比较慢，用毫秒度量。第一代计算机主存贮器容量很小（2000—4000字），它们是为科学计算而设计的。

第二代计算机是从1959年到1964年。这一代计算机的特点是以晶体管代替了电子管。以晶体管代替电子管的理由是晶体管体积小，价格低，几乎不发热，耗电量很小。这样一来，第二代计算机大大地变小了，操作时需要的功率较小。并且比第一代计算机更可靠。第二代计算机的速度用微秒来度量。对第二代计算机来说，已能使用超过30,000字的内存贮器。与早期的计算机不一样，第二代计算机在设计时还考虑到加工商业数据的需要。

第三代计算机是从1964年到1970年，这一代计算机以使用集成电路为特点。集成电路的主要优点是它比晶体管电路体积小，所以数据的加工可在很小的面积内进行，并且以更快的速度传递。集成电路能够做比它大数百倍的晶体管电路所做的工作，集成电路可减小计算机的体积。辗转之间，使用集成电路制成了小型计算机，以及大型高速计算机。第三代计算机的内部运算速度以毫微秒度量，它的主存贮器可存贮数十万个字。第三代计算机一般来说比相当的第二代计算机价格低。它的另外一个特点是，大部分的第三代计算机都是通用计算机，它们被设计得既能解决面向科学的问题，又能解决面向商业的问题，而第一代和第二代计算机只能解决科学方面的问题或只能解决商业方面的问题。

第四代计算机是从七十年代开始的。这一代计算机的特点是广泛使用微电子学原理以达到更高的集成度。已研制成功了大规模集成电路(LSI)，生产出了以微处理器为基础的计算

机,现在使用了半导体存储器,并且磁泡存储器亦被使用,第四代计算机的内存存储器中,可存储一百万或更多的字,它们的运行速度更快,却较便宜。

当前,科学家们在计算机的设计方面继续取得进展,他们的目标是制成更好、用途更广、更强有力的计算机,这种计算机有更快的运算速度,能存储更多的信息,要求小的功率,占据更小的空间,有更高的可靠性,并且成本更低。

1.3 计算机的特点

上一节叙述了计算机的发展过程,我们多次使用了“计算机”这个专门名词,但实际上还没有给它定义。在开始深入讨论计算机之前,明确一下这个专门名词的基本含意是很重要的。

严格地说,计算机是任一进行计算的装置。计算机这一名字来自拉丁文 *Computare*,其意义是计数或计算,计算机这个词一方面指现代计算机,也同样可用于算盘或台式计算器,但我们现在说“计算机”指的是具有下列特点的一种特殊类型的计算装置。

电子的 计算机用电子脉冲的运动而不是靠内部部件的机械运动得到计算结果,电子计算机的运算速度取决于电子线路,其速度只受电的传播速度限制(电的传播速度与光速相等)。

内部存储 电子计算机以电子形式把数据和指令存放在内部记忆部件里。这点不但使处理速度加快而且还构成了存储程序的基础。处理数据的速度以微秒(10^{-6} 秒)或毫微秒(10^{-9} 秒)来度量。

内存程序 数据被按照指令进行处理,指令可以是算术型的,逻辑型的或输入输出型的。为解算一个问题在计算机上所要执行的指令所构成的序列或由指令构成的一些相关序列叫程序。在运算过程中,数据和程序必须都存在存储器中。所存储的程序允许计算机从若干个可供选择的序列中选出一个指令分支来执行,还允许计算机在必要的时候重复或修改若干条指令。

自动的 计算机在没有人参与条件下,其所存储的程序指挥计算机进行一系列的操作,在这种意义下,计算机是自动的。

总的来说,计算机是一个自动的电子装置,它以高速进行操作,具有内部存储能力,在执行程序过程中存有一个由指令组成的程序。

1.4 计算机的分类

现在,成千上万台计算机在运转,执行着多种多样的功能。有许多方法对它们进行分类,其中之一是按照它所能处理的数据类型来分类(数字式和模拟式),另一种方法是按照它的用途来分类(专用型和通用型)。

数字式和模拟式计算机

计算机基本上可分成两类:数字计算机和模拟计算机。数字计算机以计数方式进行,它直接对以通常的十进制数位或其它数制的数位表示的数进行运算,它的输入和输出是用穿孔卡片上的孔、磁带上的磁迹或纸上的印刷符号等等表示的数字、字母或一些特殊符号。数字

计算机能达到很高的精确度，它们一般既用于商业数据处理也用于科学计算。大部分的现代计算机是数字式的，我们说“计算机”通常指的就是数字式计算机。

模拟计算机与数字计算机相反，它不直接对数字进行运算，而是测量连续的物理量（例如压力，温度，电压，电流，转动角度，长度等），这些物理量表示或模拟进行运算的量。例如，汽车的速度表就是一个模拟计算机，它把轮子的转动转换成指针所指的数字。模拟计算机用在科学、工程和过程控制等方面，比如说，模拟计算机可用来调节阀门以控制管道中液体的流量，或用来调节一个温度装置以控制烘炉的温度。数字计算机的计算精度比模拟计算机高，但模拟计算机处理数据的速度比数字计算机快。

在某些情况下（例如，模拟一个导弹系统或模拟新型飞机的设计），我们把模拟计算机和数字计算机的优点结合起来而制成混合型计算机。

专用和通用计算机

计算机可被设计成专门的或通用的。一个专用计算机，正象它的名字所指出的那样，是为执行一种特殊的任务而设计的。指令程序安置在机器内部。因为这计算机是专用的，所以执行这特殊任务时效率很高。但缺点是沒有灵活性，不能用来执行其它的任务，例如，专为石油提炼厂设计的专用计算机不经大改是不能转作它用的。

通用计算机，正象名字所指出的那样，是为执行各种各样的任务而设计的。它所以具有这种能力是因为它能在内存贮器中存贮各种不同的指令程序。简单地说，程序存贮概念使机器成为通用设备。它的通用性质使其能在这分钟计算工资，而在下一分钟却作库存编目控制操作。遗憾的是，这种可执行多种任务的能力是以速度和效率的减低为代价的，但在大多数情况下，可执行多种任务这一灵活性使速度和效率方面的损失成为可以接受的。

1.5 计算机的应用

现在计算机大量用在科学计算和商业处理上。经验证明，最适于使用计算机的是具有一个或多个下列特点的操作。

大容量的输入 需要处理的数据量越大，使用计算机比使用其它方法就越经济。

重复工作 计算机处理任务的准备工作需花很多时间、精力和资金，所以用计算机处理重复的题目往往是最经济的。

要求必须高速处理 需要直接访问的信息量越大，或需要的周转（对输入进行加工然后给出结果）的速度越高，使用计算机比使用其它较慢方法的价值就越大。

要求必须高精度 如果解算问题的准备工作适当，计算机处理能达到很高的精度。

复杂的计算 在某些情况下计算非常复杂，这时沒有别的方法可代替计算机。

1.6 计算机系统的基本组成部分

一个计算机包括有五个主要组成部分，每一部分都为整个计算机系统的操作出一份力。

图 1-1 是计算机的组成部分的示意图，还表明了它们怎样连接在一起。正如从图中所看到的那样，计算机的组成部分为：

控制器 整个系统的控制中心。

存储器 用来存贮程序和待处理的数据。

算术和逻辑运算装置 在这里执行算术和逻辑运算。

输入装置 它是计算机接收信息的部分，接收指令和数据，并把它们存贮在存储器里。

输出装置 计算机的这一部分从存储器接收运算结果并把它们输出给用户。

控制器、存储器与算术和逻辑运算装置在一起构成计算机系统的主机部分，称做中央处理装置或简写做CPU。整个看来，CPU处理由输入装置传送来的数据，再把处理的结果传送给输出装置。

现在让我们来更详尽地探讨计算机的每个组成部分。

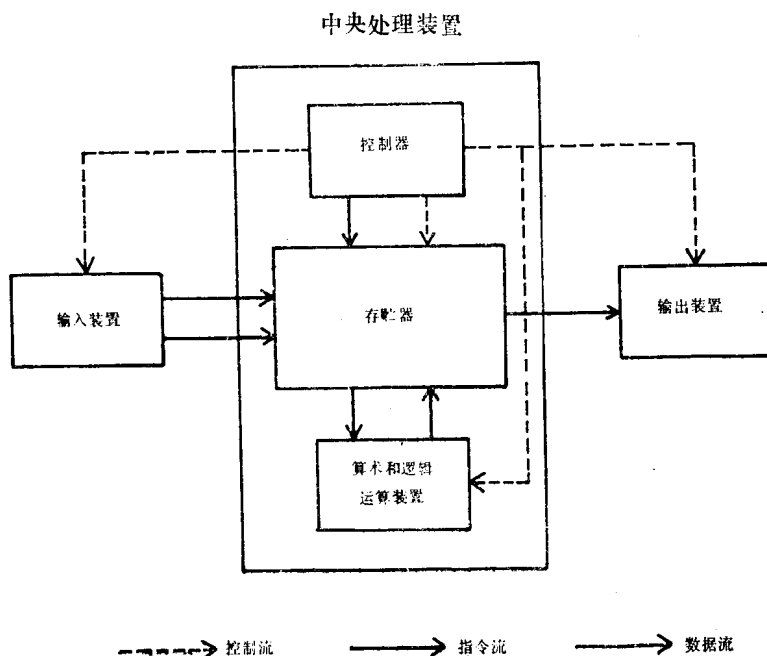


图 1-1 计算机的构成

控制器

控制器的基本功能是管理所有其它部分的运行和控制信息在各部分间的流动。存贮在计算机存储器中的程序指示控制器应该做什么，它每次从存储器中选出一条程序指令，加以翻译，然后发出适当的电控制讯号使计算机的其它部分执行这条指令。在计算机系统内，数据流动路线有三条：（1）由输入装置经控制器到存储器，（2）由存储器经控制器到算术和逻辑运算装置，再经控制器回到存储器，（3）由存储器经控制器到输出装置（参看图 1-1）。这样看来，控制器在整个计算机系统中起中枢神经系统的作用，但是它不对数据进行实际运算，这些运算应由其它装置执行。

算术和逻辑运算装置

所有的算术运算和所有的逻辑比较都在这一装置中进行。在控制装置的管理下，数据按照需要由记忆装置传送到运算和逻辑装置，经运算后再返回记忆装置。在存储器里是不对数据进行运算的。数据在这两个装置中间可能要往返若干次，一直到计算最后完成。当计算完

成以后，计算结果从存贮器被传送给输出装置。

记忆装置

存贮器（有时称做内存贮器或主存贮器）有四个功能，其中三个功能与被处理的数据有关。它们分别是：第一，数据从输入装置被传送给存贮器，在它被处理前一直存放在这里。第二，存贮器中的工作区不但存放被处理的数据，而且存放处理的中间结果。第三，存贮器把数据处理结果一直保留到它被传送给输出装置。除了刚刚提到的三个与数据有关的功能外，第四个功能是存放由输入装置或外存贮器介质（参看 1.8 节）传送来的程序以便按此程序对数据进行处理。

存贮器可分成许多叫做存贮单元的小区域。每一个存贮单元能存放一定数量的二进制位（二进制数位），其存放信息的多少，视机器的不同而不同。计算机存贮器中的特定单元由它的地址指出，地址是用来识别特定数字的。例如，在算术运算中要引用一个数值时，必须按存贮单元的地址或按依次相连存放它的单元的第二个单元的地址对其进行访问。信息在计算机中是以二进制数位的形式存贮的，一定数量的二进制数表示一个字符。这些字符可以是文字、数字或象※和\$这样的特殊符号。

在某些计算机中，一个地址访问一固定数目的存贮单元，这些单元又包含有一个固定数目的二进制数位。每组数位被看做一个单位，用来表示一个字。象这样一个地址可访问一固定数目的存贮单元的计算机是“按字编址”的，叫做固定字长计算机。

另有一些计算机，一个地址只访问一个字符，这些计算机是“按字符编址”的，它们被称做可变字长计算机。

还有一些计算机系统，既可进行固定字长运算又可进行变字长运算，因为它们以字节为单元进行编址。字节是一组二进制位，是构成计算机字的次一级单位。程序员能够使计算机的存贮器或为“按字编址”或为“按字符编址”。

IBM370 系统机是按“字节编址”的计算机，每 8 个二进制位的字节都有一个地址，四个这样的字节构成一个 32 位的固定长二进制字，这个字的地址就是它的第一个字节的地址。当一条指令访问一个固定长二进制字时，计算机就自动地存贮或读出四个字节。这种计算机还备有这样的指令，它指定两字节（半字）或八字节（双字）的固定字长。除了这些固定字长的指令外，还有可变字长指令，它们对二进制编码的字符进行操作时，指出这个字的第一个字节，同时还指出这个字所包含字节的个数。

多数计算机的存贮器是非易失型的，这意味着一个信息一旦存于一个地址，它就在那里一直保留到有另一个信息置换它为止。“破坏性写入”这个术语指的是当信息写入存贮器单元时，就取代原来存贮在那里的任何信息。

计算机的存贮器只能存贮一定数量的信息。存贮器的容量由可使用的存贮单元的总个数表示。例如 IBM370 系统可以表示有 256K 的存贮容量。在数据处理术语中，“K”表示 1024，这就说明，此计算机存贮器的实际容量是 262144 个存贮单元。

输入装置

“输入”是把信息输入计算机，输入装置是接收这些信息的设备。输入装置可包括各种各样的设备。为了使信息可被计算机理解，大多数输入装置“读出”或“感知”以编码形式出现的数据，然后按某种方式记录下来，把数据记录在输入介质上就是一种形式。例如，卡

片、纸带或磁带都是输入介质。数据用穿孔机方式记录在卡片或纸带上，用磁迹记录在磁带上，或者用特制墨水（或特种磁盘）打印的字符记录在纸制文件上。某些输入装置可不用输入介质而直接把信息输入计算机，电传打字机的键盘就是这类输入装置的一个例子。

总的说来，输入装置把数据从用户可识别的形式转换成计算机可进行处理的形式。

输出装置

“输出”指的是从计算机输出信息，输出装置就是送还这种信息的装置。输出装置包括各种各样的设备。输出信息可以操作员所能懂的形式出现或者仍保持原来的面向机器的信息形式以使输入另一个机器。各种输出装置以不同的方式执行其功能，按输出装置类型的不同它们可以：

把信息从计算机转移到穿孔卡片，穿孔纸带或磁带上；把信息按表格形式打印在纸上或事先印好的表上；产生可在远程处理网络上传送的信号；产生图形显示或缩微胶卷影象；有时甚至输出有声语言。

1.7 输入/输出装置

计算机系统经常使用多种类型的输入/输出（I/O）装置。它们的作用是在计算机与机器和/或使用计算机的人之间建立通信联系。每一个装置一般都和一种介质相联系。表1—1归纳出计算机所使用的各种介质以及输入、输出这些介质的装置。

在这一节中，我们重点研究卡片读出机和行式打印机，因为这两种装置对初学程序设计的学生来说是最基本的输入、输出装置。

表1-1 输入/输出装置和它们所使用的介质

介 质	装 置	输 入	输 出
穿 孔 卡 片	卡片读出机	×	
	卡片穿孔机		×
纸 带	纸带读出机	×	
	纸带穿孔机		×
纸	光学扫描装置	×	
	行式打印机		×
	电传打字机	×	×
	绘 图 机		×
阴极射线管 (CRT)	显 示 设 备	×	×
磁 带	磁带驱动器	×	×
磁 盘	磁盘驱动器	×	×
声 音	声音敏感设备	×	×
磁 性 墨 水	磁 性 墨 水		
	字符读出机 (MICR)	×	
缩微胶卷	记 录 器		×