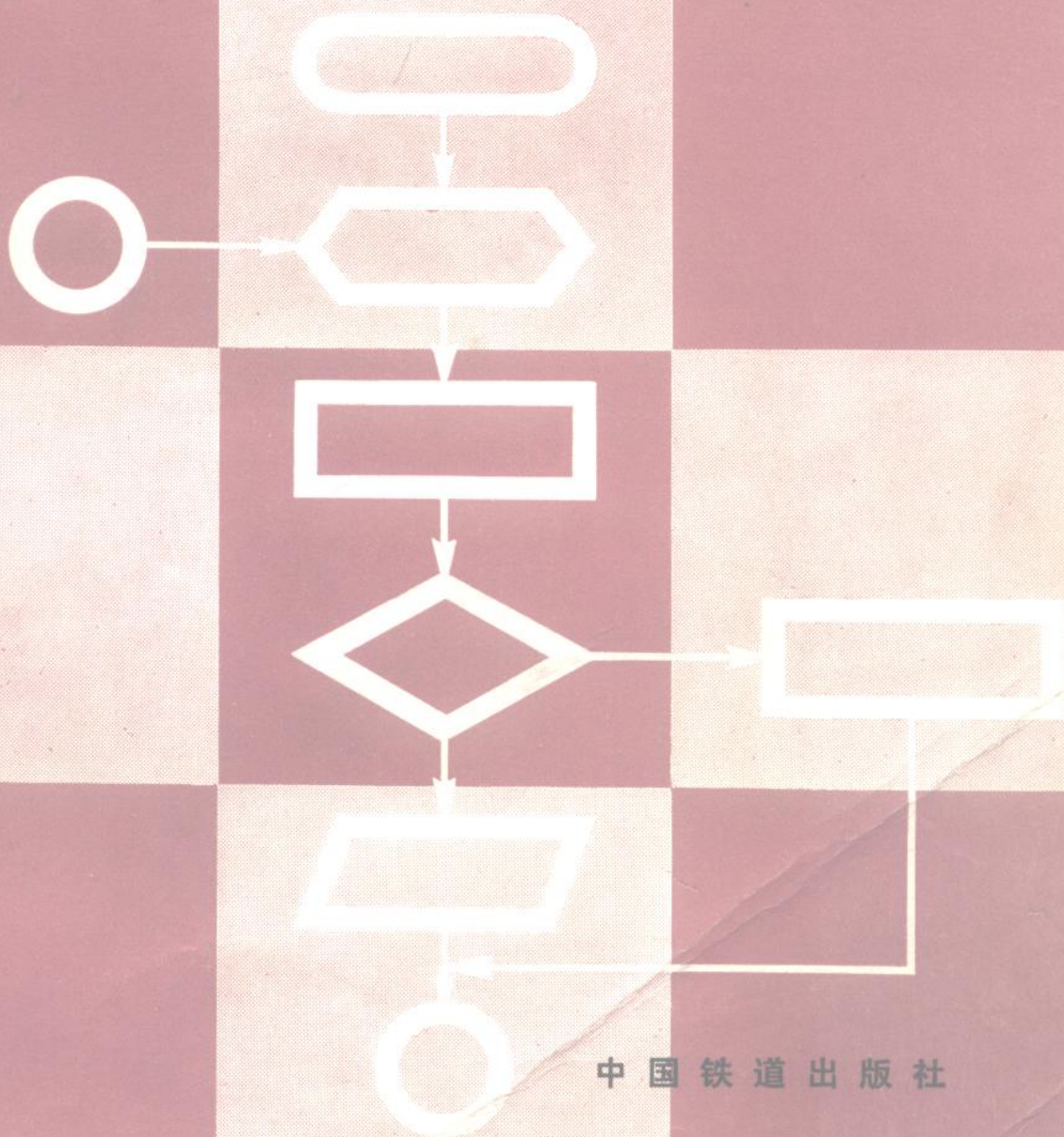


FORTRAN IV 自学指南

[美]约瑟·弗里曼 菲利普·格林伯 阿伦·M·霍夫伯 著

项斯循 陈允功 译 吴庆宝 校



中国铁道出版社

FORTRAN IV 自学指南

[美]约瑟·弗里曼 菲利普·格林伯 阿伦·M·霍夫伯 著

项斯循 陈允功 译 吴庆宝 校



中国铁道出版社

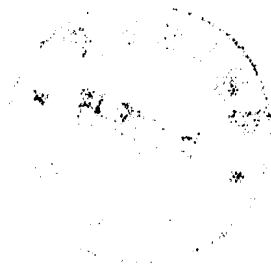
1983年·北京

内 容 简 介

本书深入浅出的介绍了FORTRAN 语言的基本知识, 提供了范围相当广泛的各种题目, 供对FORTRAN 语言感兴趣的各方面的读者学习。本书不仅对初学者是一本很好的自修书, 而且也适合已经学过FORTRAN语言的读者。

本书共分十二章。第零章介绍了数据处理的术语和概念; 第一至三章介绍了全部 FORTRAN语言的基本内容; 第四至十一章在前三章的基础上, 介绍了 FORTRAN语言的变量、输入/输出、数组的用法、控制语句和辅程序的使用等。

JS 190/26



FORTRAN IV

Self-Teaching Guides

Jehosua Friedmann, Philip Greenberg, Alan M. Hoffberg

John Wiley & Sons, 1975

FORTRAN IV 自学指南

〔美〕约瑟·弗里曼 菲利普·格林伯 阿伦·M·霍夫伯 著

项斯循 陈允功 译 吴庆宝 校

责任编辑 郭宇 封面设计 王毓平

中国铁道出版社出版

新华书店北京发行所发行

各地新华书店经售

中国铁道出版社印刷厂印

开本: 787×1092 1/32 印张: 20.75 字数: 475 千

1983年4月 第1版 1983年4月 第1次印刷

印数: 0001—13,000 册 定价: 2.15元

译者序

计算机应用的普及与否，是衡量一个国家现代化程度的一项重要标志。而推广计算机应用的一个关键问题，就是编制程序。由于程序是为各行各业服务的，如果各行各业的专业人员都能掌握程序编制的规则，并能够自己动手编程序，必然会使计算机的应用得到迅速发展。本书的翻译出版，将对编制程序的人员起到引导作用。

编制程序用的计算机语言，比较常用的有FORTRAN语言、BASIC语言、ALGOL语言、COBOL语言、PL/I语言等。其中FORTRAN语言是一种通用的计算机语言，目前世界上绝大多数的计算机都配备了它。由于FORTRAN语言使用了常见的英文单词，加上本书作者深入浅出的讲解各种语句的细节，并利用许多例句、图解说明其功能和编写方法。所以，本书读者只要有一般高中毕业的数学和英语水平，就能学会FORTRAN程序的编制方法和规则。为了便于读者自学，每章的末尾都附有习题，并给出习题答案，可供读者边学边练习，使所学的内容得到巩固。

正如本书作者所说：“FORTRAN语言已不再是专用于科学技术中进行程序编制的语言了，愈来愈多的商业、企业认识到用FORTRAN语言编程序既简单效率又高”。所以，本书的内容对各个专业的科学计算、数据处理及商业、企业管理的程序编制，都是普遍适用的。

编写正规的FORTRAN程序，应书写在专用的程序纸上（注意各列的间距与语句之间的距离），并按FORTRAN语言的格式书写程序。读者在练习时，可以写在普通的纸上，但也要注意使用正确的书写格式，以养成良好的习惯。

本书第七、八、九三章由陈允功翻译，其他各章均由项斯循翻译。

本书的校阅工作得到了第四机械工业部一九一五所领导的重视和支持，在此表示感谢。

由于水平有限，译文中的错误和不妥之处，请读者批评指正。

译者

一九八一年十二月于北京

致读者

虽然FORTRAN是FORmula TRANslation (公式翻译) 的简称, 但它已不再是专用于科学技术中进行程序编制的语言了。愈来愈多的商业、企业认识到用 FORTRAN 语言编程既简单又效率高。在这本自学指南中, 我们提供了范围相当广泛的各种题目供 FORTRAN语言感兴趣的各方面的读者学习。读者只要有一些基本的高中代数知识就可以了。

由于有一些读者对数据处理还没有任何基础, 所以我们编写了第零章, 即计算机概述。通过这一章对数据处理的简单介绍 (它的术语和概念), 使以后几章中的许多术语不致于感到生疏难懂。

我们希望你尽快开始编写 FORTRAN 程序, 所以在这以后三章中讨论了全部FORTRAN语言的基本内容。第一章使你熟悉一个FORTRAN 程序及其编码形式。同时学会用流程图方法求解一个简单计算机的题目。在第二章中, 我们告诉你算术运算方法和在计算机存储器中传送数据的情况。在你读完第三章中关于输入/输出的基本概念后, 就会编写简单的但是比较完整的FORTRAN程序了。

不过, 这时你还不会满足, 因为你还要控制程序的执行, 即根据程序中判别的不同条件或重复执行某几条指令, 或跳过另一些指令。因此, 我们在第四章中介绍了基本的控制语句。

然后, 在第五章中进一步介绍输入/输出, 而其它的输入/输出则在后面第九章中介绍。但在学习时请勿把它提到前面来学, 因为中间插进的几章内容按学习进程是必须的。

第六章中叙述了附加的控制语句。它能使你编写出极有效的程序, 使用少量的语句便可得到大量的数据处理。这样的控制语句, 后来由第七章介绍的多维变量所利用。第八章对其他类型的变量一一作了图解说明。

第九章是根据前面几章的内容引伸出来的, 进一步讨论了输入/输出的特点。

当你编写了许多 FORTRAN 程序以后, 你会理解到, 在一个程序或一串程序中, 会重复出现顺序完全相同的 FORTRAN 语句。这时你就会对第十章的内容感到兴趣。这里提出的函数和辅程序, 使你能把 FORTRAN 编码模块化并重复使用, 从而避免经常地反复书写它。

第十一章介绍了使用磁带和磁盘设备提供了专门的输入/输出语句的资料, 以及其他一些有价值的FORTRAN程序编制技巧。

在附录中介绍了一些有用的参考资料, 包括算术操作的精确度问题, 机内FORTRAN程序库 (可以重复使用的 FORTRAN 编码) 的说明, 以及为提高编写程序效率的几点意见。

约瑟·弗里曼

菲利普·格林伯

阿伦·M·霍夫伯

一九七五年三月于纽约

怎样使用本书

要学习FORTRAN程序，读者只需要有这本书、一支铅笔、一些草稿纸和FORTRAN程序纸，以及一个安静的工作环境。既不需要有事先学过数据处理的经验，也不需要接触过计算机（虽然这并没有什么坏处）。但是，倘若你有机会使用一台特定的计算机装置，你就找到该机的FORTRAN语言手册，弄明白这台计算机系统的特殊性能是什么。

由于本书具有自学的特点，你会应用学到的每一个新概念或新技巧积极地钻研学习FORTRAN程序。本书内容的各小节都编以号码。每一小节都告诉你关于FORTRAN语言的一些新的内容，然后，要求你回答一个问题或编写一小段程序。当你答完问题或写完程序后，拿你的答案与书中虚线下面的答案比较。我们要求你在写完你的答案以前，用纸把本书的答案盖上。这些精心编写的题目是为了引起你注意说明和例题中的要点，并帮助你学习运用已说明了的内容。假使你的答案对了，就意味着你已理解了这些内容，而且可以继续学下一小节。若是你的答案跟本书的答案不一致，就应复习前面的小节。在确实理解其内容以前，不要急于往下进行。

在每一章的末尾，有一个自学测验题，它帮助你检查自己对这一章的内容学得如何。答案就在每个测验题的末尾，并附有有关小节的编号以供你复习具体问题参考。所以你可以在每读完一章之后，立即自我测验一下。你也可以在读完一章之后，暂停一下，把这些测验题作为在开始读下一章之前的复习。

每一章开始时，列出了学习本章的目的，即你在学完本章之后应会做的事情。如果你事先已具有一些使用FORTRAN语言的经验，且对这章的目的比较熟悉，你就可以将测验题作为对这一章的复习，又作为你应从何处开始继续学习本书的指导。在读每章之前，试做一下自学测验题。测验的结果若是较好，你就只需要读几个答得不好的有关小节。若许多题你都没答对，就需从头学习这一章。

目 录

怎样使用本书	1
第零章 计算机概述	1
什么是计算机	1
主存储器	2
辅助存储器	4
虚拟存储器	5
输入输出设备	6
I/O介质的编码系统	8
ALU (算术和逻辑运算部件)	8
控制器	9
计算机语言	9
数据组织——文件、记录、域	13
第一章 FORTRAN程序的编制	15
流程图符号: 启动和停止	15
流程图符号: 输入和输出	16
流程图符号: 数据处理或计算	16
FORTRAN字符组	17
手写FORTRAN代码	18
流程图符号: 连接符	19
流程图符号: 判断点	20
在FORTRAN程序中识别各种操作	22
在程序内编制文件	26
准备FORTRAN程序纸	26
习题	31
第二章 变量和算术语句	34
变量和常量	34
算术操作符	39
括号	40
赋值语句	45
习题	51
第三章 基本的输入和输出	54
READ语句	54
语句标号	57
WRITE语句	57
FORMAT语句	59
I格式代码	61
STOP和END——快速访问	68

一个简单而完整的程序	68
程序设计者的控制卡片	69
X格式代码	70
F格式代码	73
习题	75
第四章 基本的控制语句	83
无条件GO TO语句	83
算术IF语句	86
STOP语句	100
PAUSE语句	101
END语句	102
习题	102
第五章 输入和输出 (续一)	108
托架控制字符	108
/格式代码	112
重复因子	116
记录长度	121
文字型数据——H格式代码	122
文字型数据——另一种方法	123
A格式代码	127
FORMAT语句的重复	129
习题	132
第六章 控制语句 (续)	138
DO循环	138
DO循环的增量	144
DO循环的控制变量	144
DO循环的结束	146
DO循环的缩进书写法	147
DO循环的流程图	148
DO循环的终结	149
嵌套DO循环	151
逻辑IF语句	154
计算GO TO语句	167
赋值GO TO语句	170
习题	172
第七章 数组的妙用	176
一维数组	176
交换分类	180
DIMENSION语句	185
多维数组	190
数组是如何存入存储器的?	192
数组的用途	193
习题	200
第八章 再谈变量	204

E型表示法	204
双精度常数	208
双精度变量	209
类型语句	212
逻辑赋值语句	216
IMPLICIT语句	219
DATA语句	220
EQUIVALENCE语句	223
习题	224
第九章 输入和输出 (续二)	230
数组的输入和输出	230
隐DO循环	233
输出用的E和D格式说明符	236
输入用的E和D格式说明符	238
比例因子 (P型格式说明)	241
L型格式代码	242
习题	246
第十章 辅助程序的混合使用	250
语句函数	250
虚元和变量	253
函数辅程序	258
子程序辅程序	267
EXTERNAL语句	270
COMMON语句	275
带标志的COMMON语句	278
数据块辅程序	280
程序库函数	280
习题	280
第十一章 输入和输出 (续三)	295
目标时间FORMAT语句	295
磁带和磁盘	299
无FORMAT语句的输入和输出	300
END及ERR的选择	301
磁带命令	302
磁盘命令	304
习题	311
附录一 算术操作和赋值的精确度	317
附录二 FORTRAN程序库函数	318
附录三 提高编程序质量的几点想法	320

第零章 计算机概述

在这一章中，我们将向你介绍计算机的构造以及操作计算机的概念。你也将学到一些必要的基本术语以便正确理解这本自学指南。

通过对本章的学习，你可以掌握下列各点：

- a. 通用数字计算机；
- b. 计算机的整机系统；
- c. 输入和输出设备；
- d. 机器语言；
- e. 高级程序设计语言；
- f. 程序设计的句法和语义；
- g. FORTRAN语言；
- h. 其他方面的计算机术语。

倘若你已经懂得计算机系统是如何操作的，便可跳过本章的第一部分，而从第16节中“介绍FORTRAN语言”开始读。

什么是计算机

1. 计算机是一种电子机器，它能处理得到的信息（数据或指令），并将它变为输出信息或数据。正如推土机能够处理泥土和石头一样，我们可以把计算机看成是一架高速符号处理机，它能够作算术运算，读、写、存储信息甚至可以作出判断。

计算机是一架“哑巴机器”，即它只能作人们通过“指令”让它作的事。这些指令的编写过程叫作“编码”或“程序设计”。为求解某一特定问题而编写的指令，组成一个“程序”。编制这些指令的人，叫作程序设计者。

“数据”是根据程序设计者所提供的“处方”（程序），由计算机处理的实体。总之，程序是以数据或信息为操作对象，而将其读入计算机的一组指令。

计算机有两大类：模拟计算机（它以连续的形式表示测量数据）和数字计算机（它以数字形式的量值表示数据）。凡是模拟计算机能做的工作，数字计算机都能做到。但是，反过来并不尽然。此外，数字计算机能够得出较精确的结果。本指南只涉及数字计算机。

“通用数字计算机”能接受很多不同类别的程序，因而能够在新程序送入后，一个接一个地解题。而“专用数字计算机”是按只能使用在其内部已建立好的程序重复地作指定的操作设计的。在本指南中，凡用到“计算机”这一术语时，是指通用数字计算机而言。

用“对”或“错”回答下列问题：

(a) 导弹制导计算机是专用计算机。

(b) 能计算工资单，登记存货清单，同时又能诊断疾病的计算机是通用计算机。

(a) 对； (b) 对； (c) 错。

(c) 模拟计算机通常比数字计算机精确。

2. 一个典型的计算机系统包括六个基本组成部分：

控制部件；

中央存储部件或主存储部件；

算术运算部件及逻辑部件 (ALU)；

辅助存储部件；

输入部件；

输出部件。

前三项往往合并成一个设备，叫做中央处理部件 (CPU)。CPU 是直接用于处理信息或数据的。

信息进入计算机系统，称为“输入”。信息一旦进入，就被计算机程序所处理。作为处理结果所产生的信息，称为“输出”。上面所述计算机系统的后三部分，包括把信息输入到存储器的输入过程和把存储器中的数据处理后产生信息的输出过程。

请添上下面的空白

进入计算机系统的信息称为_____。计算机把输入的信息进行处理后产生_____。

输入；输出。

下面几节介绍二进制、十六进制和八进制系统。如果你对 these 数制并不熟悉，请别着急，且先读下去。要读懂本书的其余部分，只要对数制有些基本概念就可以了，若要多学些这方面的知识，可参考 R·阿希雷著的《计算机世界的数学基础》和马丁·哈利斯著的《数据处理引论》。

主 存 储 器

3. 计算机是按照指令处理数据的。这些指令和有关的数据存储在计算机存储部件中，这种部件称为主存储器（也称做内存存储器）。大多数计算机的内存存储器是由很小（相当于针尖般大小）的磁芯组成，并排成一个个磁芯平面（磁芯板），然后再水平地或垂直地组合起来。按计算机的设计，把程序读入内存存储器。这时每个磁芯就很容易被磁化成两种可能的状态，即磁化或退磁；或者磁化成两个方向，即顺时针或逆时针。这两种磁化状态可以用一个灯泡的“亮”或“灭”来表示。二进制的数字“0”和“1”，也常常用来代表这两种状态。因此，一个磁芯的状态也被称为一位。

用“对”或“错”回答下面的问题：

(a) 二进制的位，能毫不费力地磁化成两种可能的状态。

(b) 磁芯磁化的方向是由二进制数字的 0 或 1 来表示的。

(c) 一位是一小块塑性物体。

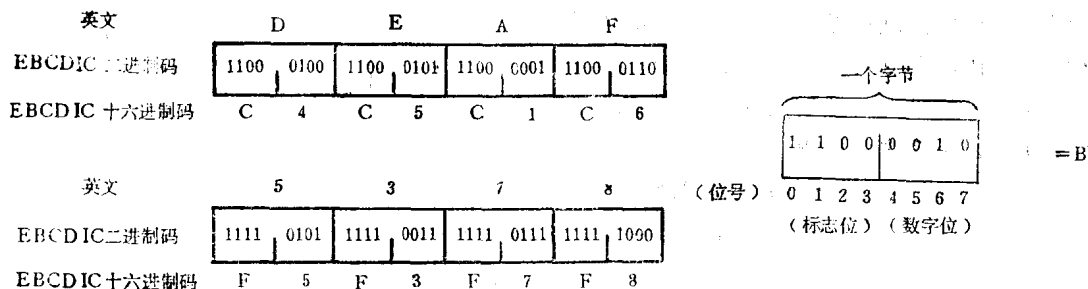
(a) 和 (b) 是对的，(c) 是错的。

4. 就像莫尔斯电码那样，把“点”和“划”组合成组来表示字符，位（磁芯）也是这样组成组的。许多计算机的主存储器，是把相邻八位分成一组，通常称做字节。每位的磁化状态，是计算机的一个信号，而八个连续磁芯（一个字节）就能有 256 种不同的“1”或“0”的组合。所以，根据所用的组合，我们可以对一个字节给出多达 256 种不同的意义。这样，

一台计算机就能够用位的特殊组合和特定的符号相联系。例如,某些计算机以字母B和11000010相联系。一个字节中“1”与“0”的各种组合,就是所谓的EBCDIC码(Extended Binary Coded Decimal Interchange Code——(扩展二进制编码的十进制变换码)。在这种编码中,字母B表示为:

为方便起见,我们把一个字节分成两半,即标志部分和数字部分。

四个二进制数位相应于一个十六进制数位,所以一个字节也可以用两个十六进制数位来表示。下面是一组代码的例子:



别的代码也一样。例如,在BCD(Binary Coded Decimal—二进制编码的十进制)中,六个连续位可以表示64个不同的字符,包括26个英文字母,数字和专用字符。ASCII码(American standard Code for Information Interchange—美国信息交换标准代码)则采用七个连续位以表示各种符号,最多可以表示128个不同的符号。ASCII—8码是在使用八位编码的计算机中用的一种ASCII码的八位变型。ASCII码用在通讯和数据处理工业中,以便在各类型计算机和外围设备之间获得高效率的通讯联系。

(a) 什么叫做字节?

(b) 一个字节有多少种不同的0和1的组合?

(a) 字节是能表示一个符号的一组八个相邻位;

(b) 一个字节有256种不同的组合。

5. 我们可以把计算机的存储器与邮局的信箱相比较,每个信箱都有一个用以识别的号码。我们希望你同意信箱的内容和它的地址(号码)是分开的,而且是不一样的。同样,计算机把唯一的地址(号码)分配给每个内存单元,这和该内存单元中包含的数字或字母则是另外一回事。程序设计者在编写一个程序时,对每一个内存单元赋以一个符号名(或地址)。计算机就用一个对照表,使这个名同唯一的编号地址联系起来,从而减轻了程序设计者的繁重劳动。怎样选择名,将在后面讨论。这里请记住,符号单元名和存储单元地址号码是一样的,它和存储单元中的内容是不同的(见图0—1)。

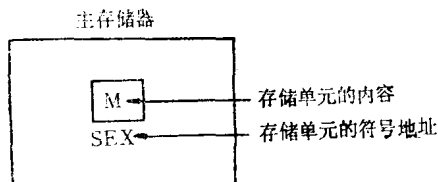


图 0—1

习惯上,我们用英文字母K表示一组1,000个可编址存储单元(精确些说, $K=1,024=2^{10}$)。典型的计算机可能有64K个存储单元。大多数计算机的存储容量从32K到1,000K以上。

用“对”或“错”回答下面的问题:

(a) 一个存储单元可以包含数字、字母或其他可识别的符号。

(b) 每个存储单元是由一个称做“地址”的单独的存储单元来识别。

(c) 计算机对每个存储单元赋以一个符号地址。

(d) 符号地址与该地址的内容是一样的。

(a) 和 (b) 是对的; (c) 和 (d) 是错的。

6. 计算机装有可以把信息读入主存储器或从主存储器中把信息写出来的电子线路。当我们把一个信息读入某个存储单元时, 新读入的信息取代了原先存储在该单元中的信息。而当信息从某个磁芯单元中取出时, 该单元的内容仍保留不变。实际上我们获得了该信息的副本。因此, 一旦信息存储在内存中, 它就可以多次取用。在这个意义上, 信息类似于音乐或声音被存储在磁带上, 然后用录音机重放。我们用图 0—2 说明这一概念。

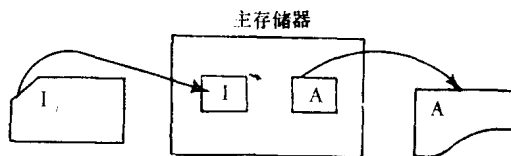


图 0—2

用“对”或是“错”回答下列问题:

(a) 一个磁芯单元一经磁化, 就能保留它的磁性。

(b) 当我们把信息输入到某磁芯单元时, 该单元的原有内容保持不变。

(c) 许多计算机能够在不到百万分之一秒或更少的时间里把信息输入到内存储器去或从内存储器中输出 (我们还未把这一点向读者说过!)。

(a) 和 (c) 是对的; (b) 是错的。

7. 有些计算机用一个字节 (用它自己的地址) 作为信息的基本单位。一般用一个字节来表示信息的一个单位。信息的单位称作字符, 通常由英文字母、数字、或象 \$ 这样的特殊符号组成。通常把字符组合成字, 它是存储在一连串字节中的。这样的单位通常就表示一个数字。计算机字的长短 (位数) 决定了存储器可以表示数的量。

一个字有一个单独的地址, 即第一个字节的地址。FORTRAN 语言采用“变量” (即符号名) 去标示地址。所以, 当我们在 FORTRAN 语言中提到符号地址 (即变量) 时, 计算机就自动存入或取出一个字。例如, 变量名 AMT 是指一个特定的区域, 或是内存储器中的一个地址, 它的内容是一个字或一个信息单位。若要存储较大的数字, 我们可以采用双字长作为一个单位来编址 (关于这一点, 将在第八章中详细论述)。

假设某台计算机采用的每个字节有八位, 每个字有四个字节, 试将下面 A 列中的项与 B 列中的一项相匹配。

A	B
(a) 字节	(1) 64 位
(b) 半字	(2) 8 位
(c) 字	(3) 16 位
(d) 双字	(4) 32 位

(a) 2; (b) 3; (c) 4; (d) 1。

辅助存储器

8. 存储器有两种, 主存储器和辅助存储器。

主存储器用来放置或存储输入信息并按需要消去输出信息。这些信息可以是:

数据（用于输入，计算机处理，或输出）；

指挥计算机操作的指令；

在计算机程序执行过程中所需的参考数据表格代码和常数）。

由计算机处理的所有信息都必须经过主存储器，所以在执行程序中主存储器必须能够保留所需的指令和数据。在实际应用中，往往有许多程序超出主存储器的容量，需要由辅助存储器提供附加的存储器。

辅助存储器（也称为二级存储器）有两种型式：

（1）顺序存取。它包括所有各种磁带存储介质。为了找到一个记录，必须按顺序读出所要记录前面的全部记录，我们将这种按顺序读出并记录的方法，称做顺序存取。这恰恰像家用的磁带录音机一样，我们必须从磁带的开头一直转到我们要听的部分。

（2）直接存取或随机存取。它包括磁鼓、数据胞和磁盘存储介质。在这些介质中，任一记录都可以直接存取。这就好比我们可以把一个留声机的唱针直接移到我们所要听的那一段音乐上去，而无需从唱片开头听起。

主存储器比辅助存储器的价钱要贵得多，但是速度也快得多。所以，它是在程序执行时用来存储指令和一些需要的数据的。主存储器和辅助存储器的作用，如图 0—3 所示。

把下面 A 列中的一项与 B 列中的一项或几项相匹配：

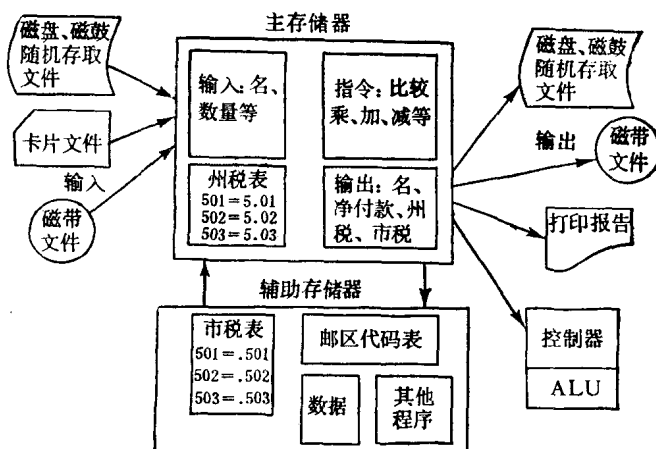


图 0—3

A

- (a) 主存储器
(b) 辅助存储器

B

- (1) 二级存储器
(2) 磁芯存储器
(3) 磁盘
(4) 磁带
(5) 磁鼓

用“错”或“对”回答下面的问题：

(c) 辅助存储器是在程序执行期间用以保持所需数据和指令的。

(d) 主存储器是在程序执行期间主要用以存储并不立即需要的数据的。

(a) 2; (b) 1, 3, 4, 5; (c) 和 (d) 都是错的，与它相反的倒是对的。

虚拟存储器

9. 只有当程序和指令存储在主存储器中时，一台计算机才能正常工作。所以，程序设计者必须肯定他的程序和所需的数据都能适当存入内存存储器时才能运行程序。在内存存储器容量太小而不足以容纳全部程序时，我们就采用一种叫做程序分段的技术。这个技术需要在全过程中一直把该程序的主段放在主存储器中，而把程序的其余部分分为许多小段（见图 0—

4)。把第一段放入主存储器靠近主段的地方，其余各段放在辅助存储器内。当第一段执行完后，就把另一段移入主存储器中，以取代第一段。用这一方法对辅助存储器中的其他各段重复处理，直至程序运行完。

近年来，有些计算机厂商采用了一种称为虚拟存储器的新概念。其基本想法是由计算机本身自动地把程序分成小段，而不是由程序设计者去分段。这些小段被分

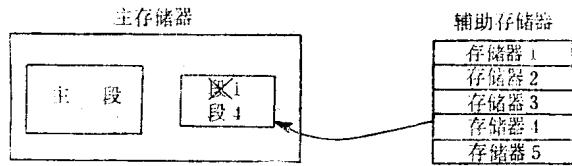


图 0-4

散在整个主存储器和辅助存储器中。计算机使用表格和索引掌握各小段的行踪，并在需要时把某一段移入主存储器中。这样一来，程序设计者不必关心存储器的管理，因为可利用的存储器就没有数量的限制了。因此把它称为虚拟存储器。

现在请回答，什么叫虚拟存储器？

虚拟存储器是计算机使用的一种技术，用来管理程序存储单元分段，这样好像使计算机为程序设计者提供了一个几乎没有数量限制的存储器。

输入输出设备

10. 卡片的输入输出设备往往是由同一个部件组成的。卡片阅读机就是一般的输入设备。往往把输入信息用键盘穿孔机（这很类似于一台打字机。如果你想学习如何穿孔，就要找到一台可以使用的键盘穿孔机，或阅读键盘穿孔机的说明书）在卡片上穿孔。图 0-5 所示是一张带有一组标准穿孔符号的80列典型穿孔卡片。

穿孔卡片上的信息可由卡片阅读机传送（输入）到存储器中去。同样，也可以从计算机中由卡片穿孔机把信息穿孔在卡片上（输出）。典型的卡片阅读机每分钟可以处理 1,000 张卡片。但与执行指令用微秒计算的计算机相比，简直是太慢了。

把下面 B 列中的一项或几项与 A 列中的一项相匹配：

A 列	B 列
(a) 卡片阅读机	(1) 输出部件
(b) 卡片穿孔机	(2) 在卡片上穿孔
(c) 键盘穿孔机	(3) 输入部件

(a) 3; (b) 1,2,3; (c) 2。

11. 键盘终端（有时称做电传打字机）常常用来把指令和数据输入到计算机存储器中去。一台键盘终端和一台附加了一些完成特殊功能键的电传打字机很相似。程序设计者可以用键盘打入一个程序和数据，然后命令计算机执行程序。程序设计者的输入和计算机的输出都印在纸上，输出是自动地送到终端去。如果你有这样一台可用的终端，务必要使用它。一个 FORTRAN 程序和所得到的输出可以由一个终端来交换信息。虽然终端是速度很慢的设备，但它能与计算机系统直接交换信息，工作得很好。大部分计算机系统使用一种称为控制台的键盘终端，与数据处理员进行人机通讯。但通常在大量输入/输出（或 I/O）时，并不使用控制台。

把下面 A 列中的一项与 B 列中的一项或几项相匹配：

卡片顶部

专用字符

和

数字

行 12

区段穿孔区

数字穿孔区

80列标度-

如果只在一个数字孔，它表示一个数字。

12行中的孔和	A	B	C	D	E	F	G	H	I
1	-	-	-	-	-	-	-	-	-
2	-	-	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-	-	-
4	-	-	-	-	-	-	-	-	-
5	-	-	-	-	-	-	-	-	-
6	-	-	-	-	-	-	-	-	-
7	-	-	-	-	-	-	-	-	-
8	-	-	-	-	-	-	-	-	-
9	-	-	-	-	-	-	-	-	-

11 行中	J	K	L	M	N	O	P	Q	R
的孔和	-	-	-	-	-	-	-	-	-
	1	2	3	4	5	6	7	8	9

0 行中	S	T	U	V	W	X	Y	Z
的孔和	-	-	-	-	-	-	-	-
	2	3	4	5	6	7	8	9

已穿的孔

5-0

A列	B列
(a) 键盘终端	(1) 是比较慢的I/O设备
(b) 卡片阅读机	(2) 仅作输入设备用
	(3) 像一台电传打字机

(a) 1,3; (b) 2。

12. 最普通的输出设备是行式打印机, 或简称打印机。计算机中心的大部分输出必须以打印文件的形式交给非数据处理人员, 和(或)上级管理单位的人阅读, 所以打印机打印出的数据是便于阅读的。一般的打印机每分钟能打印1,100行, 有的能达到每分钟3,600行以上的速度。如果每页打印50行, 以每分钟3,600行的速度计算, 每分钟就能打印72页。

但是打印机有两个主要的缺点。第一, 与计算机处理数据的速度相比, 它比较慢; 第二, 它耗用大量纸张, 因而处理和存放这些纸张会造成很多困难。

用“对”或“错”回答下列问题:

- (a) 打印机是一种输入设备。
- (b) 打印机是一种比较缓慢的输出设备。
- (c) 打印机能耗用大量纸张。

(a) 错; (b) 对; (c) 对。

I/O介质的编码系统

13. 说不同语言的两个人互相交谈时, 需要一个译员。译员把一个人的话翻译为另一人所能听懂的语言。同样情况; 用人们能懂的形式记录下来的信息, 通常不是计算机所能懂的, 反之亦然。这个人与计算机间信息交换的问题, 是用I/O设备作为介质而解决的。前面我们已经介绍了输入用的卡片阅读机和输出用的卡片穿孔机及打印机, 还有I/O用的电传打字机。第十一章中我们还要讨论另外两种I/O设备, 即磁带和磁盘。

由于工艺上的限制, 每种I/O设备需要独特的介质, 并用它交换信息。卡片阅读机和卡片穿孔机使用穿孔卡片, 打印机使用连续的纸张, 电传打字机使用连续的纸张和纸带, 磁带机使用磁带, 磁盘机使用磁盘。

试将下面A列中的设备与B列中的介质相匹配。

A列	B列
(a) 卡片阅读机	(1) 磁带
(b) 卡片穿孔机	(2) 穿孔卡片
(c) 电传打字机	(3) 纸带
(d) 磁带机	(4) 磁盘
(e) 磁盘机	(5) 连续的纸张

(a) 2; (b) 2; (c) 3,5; (d) 1; (e) 4。

ALU (算术和逻辑运算部件)

14. ALU (Arithmetic and Logical Unit)是由一系列小存储部件(称做寄存器)和