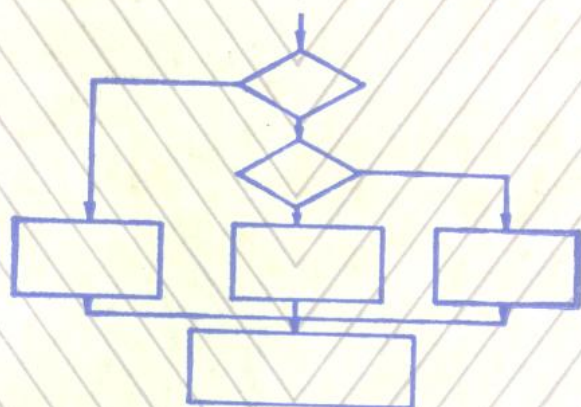
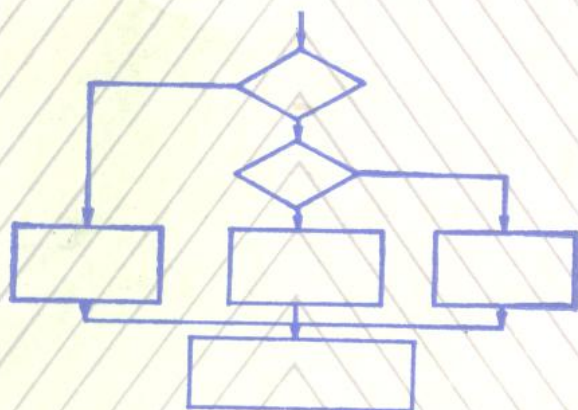




COBOL程序设计

廖艾迪编著



信息与逻辑丛书

COBOL程序设计

廖艾迪 编著

天津科学技术出版社

本丛书编委会名单

主编：张锦文

编委：陈景润、王树林、王 联、王建方、张宏裕、邵品瑞、
赵沁平、徐书润、廖艾迪

信息与逻辑丛书

COBOL程序设计

廖艾迪 编著

责任编辑：黄立民

天津科学技术出版社出版

天津市赤峰道130号

天津新华印刷一厂印刷

新华书店天津发行所发行

开本 850×1168毫米 1/32 印张 10.75 字数 272,000

一九八六年八月第一版

一九八六年八月第一次印刷

印数：1-4,000

书号：17212·17 定价：2.45元

编 者 的 话

在现实世界与人类社会，信息的传输、存取与加工的重要性和价值是无法估价的。由于科学技术的进步、工农业生产的发展、军事活动、气象与交通运输、商品交流、文化生活等领域都在不断地提供大量的信息，这就要求迅速而准确地对信息进行传输、存取与加工。信息是通过各种形式表达的，有自然语言（汉语的、英语的等等）形式，数字形式，符号形式和物质形式。信息的内容是多种多样的，有些可以公开传输，有些又必须加以保密，秘密传输。当今世界，无线电秘密传输信息是困难的，它必须设法使预定传输的有规则的、含有重要内容的信息加以扰乱，使接受了这些信息的敌方无法识别它所表达的内容，而又必须使接受到它的我方把扰乱的信息迅速地恢复它的原始含意。有些信息无长期保存的价值，而有些不仅要长期存贮，而且还要能够随时提取。有些信息必须进行大量的加工，提取有用的部分。无线电通讯中的编码、加保护码、译码与破码也是一例。合理地使用信息，有时比合理地使用物质更为重要。在这个意义下，我们可以说，现在是信息的时代，具体地说是信息量巨增、信息的意义与价值巨增、信息的研究与利用巨增的时代。人们不仅借助于物质器件传输、存取与加工信息，而且还要借助于数字、符号与语言的工具才能迅速、准确地处理信息。不仅需要设计与制做传输、存取与加工信息的硬件，而且还必须设计与制做相应的软件。现阶段，在一定意义下，设计与制做软件是更为重要的，因此，人们称八十年代为软件的时代。

数字与语言都可以用符号、符号串和它们的集合所表达，处理信息也就是处理符号串和相应的集合（即符号语言或形式语

言)。信息来源是广泛的、系统的，并且常常是相互关联的。这样，形式语言、形式系统也是丰富多采的。

数据库、知识库、专家咨询系统、专家系统、自动程序设计、自动证明数学定理、机器人和各种智能工程等领域的研究和应用已成为当前科学技术的重要课题。这一系列课题中不仅需要已有的逻辑知识，而且在不断地发展新的逻辑工具。应用逻辑方法是上述各个课题的共性。众所周知，数学方法是一切自然科学、技术科学和工程技术的重要工具之一，然而在信息系统与智能工程中，逻辑方法不仅比一般的数学方法应用更广泛，而且它是更为基本的方法。一个加法运算必须分解为一系列基本的逻辑步骤，每一次加法运算本身都是通过一条很长的逻辑链条而完成的。所以，数学运算都是通过逻辑运算的链条而完成的。正如由微小的力量可以聚集成巨大的力量一样，由微小的逻辑步骤也可以以各种结构形式聚集成巨大的逻辑步骤，实现巨大的智能活动，高速的电子元件为这种琐碎的、冗长的步骤提供了可能。把数学运算和其它的逻辑过程（判断、推演）组成有机的部分，再把各个部分组成各种系统，由若干小系统组成大系统。大系统的组成以及工作的过程和环节，都存在着逻辑的联系、逻辑转换和逻辑过程。这不仅运用了来自人类的思维过程，特别是数学的思维过程中发展起来的数理逻辑理论与方法（它起源于莱布尼兹，经过布尔、施莱德、德·摩根、弗雷格、罗素、希尔伯特和哥德尔形成独立学科。近五十年来它又有了新的重大的发展，解决了一批引人注目的数学难题，还使一批数学难题获得重大进展，产生了一批令人发省的数学结果），而且又为数理逻辑提供了新概念、新方法和实际背景，使纯逻辑的某些研究有了新的生命力。在这个意义下，我们可以说，计算机、信息系统和智能工程促进了数理逻辑的研究与发展，可能为数理逻辑的研究与发展开辟新的前景。

本丛书的宗旨是阐述信息系统、计算机、智能工程与数理逻辑

辑的联系，借以促进它们的发展。一方面从信息系统、计算机和智能工程的角度阐述数理逻辑的若干重大的逻辑成果（某些纯数学的成果如连续统假设、群论、字的判定问题等除外）和基本方法，借以促进前者的发展，也阐述一些逻辑问题，希望寻求计算机解决它们的方法，或者证明不存在计算机解决的方法。另一方面是阐述信息系统、计算机和智能工程等领域的进展，特别是逻辑方法，借以促进它们的发展与推广，并进而为数理逻辑研究提供新的资料和问题。任何一门学科，当它能够不断地提出大量的研究问题，它就充满着生命力，而问题的缺乏则预示着独立发展的衰竭与中止。我们的领域是不断提出问题，充满生命力和蓬勃发展的研究领域。

我们希望通过本丛书的十数卷本能够阐明我们的研究领域的各项专题及它们的联系，使本丛书能成为这一领域的广大科学技术工作者、教师和学生较系统的由浅入深的读物，从而对读者有所裨益。

本丛书的撰写和出版过程中，我们得到学术界许多部门和许多朋友的多方支持，得到许多学者、专家与权威的大力支持和指教，得到天津科技出版社的热情有力支持。借此机会表示由衷的感谢。

《信息与逻辑》丛书编委

1984年10月

目 录

绪论	(1)
第一章 COBOL程序概述.....	(6)
§1 COBOL程序结构	(6)
§2 COBOL程序的书写格式	(7)
§3 COBOL的基本概念	(9)
习题一	(15)
第二章 标识部与环境部	(18)
§1 标识部	(18)
§2 环境部	(19)
习题二	(24)
第三章 数据部	(25)
§1 数据部的组成与有关概念	(25)
§2 工作存储节	(33)
§3 PICTURE (形象) 子句	(36)
§4 USAGE (用法) 子句	(54)
§5 JUSTIFIED (对齐) 子句	(56)
§6 VALUE (初值) 子句	(58)
§7 BLANK (遇零置空) 子句	(60)
§8 REDEFINES (重定义) 子句	(60)
§9 RENAMES (重新命名) 子句.....	(62)
§10 OCCURS (重现) 子句	(63)
§11 SIGN (符号) 子句和SYNCHRONIZED(同步)子句	(68)
习题三	(71)

第四章 过程部 (74)

§1 过程部的结构 (74)

§2 接收语句与显示语句 (75)

§3 传输语句 (MOVE) (81)

§4 算术运算语句 (87)

§5 条件语句 (IF) (99)

§6 转移语句 (GO TO) (111)

§7 修改语句 (ALTER) (113)

§8 执行语句 (PERFORM) (114)

§9 出口语句 (EXIT) (129)

§10 检验语句 (INSPECT) (132)

§11 字符串连接与分解语句 (141)

§12 停机语句 (STOP) (154)

习题四 (155)

第五章 表处理 (161)

§1 表的描述 (161)

§2 表处理语句 (169)

习题五 (182)

第六章 文件处理 (185)

§1 基本概念 (185)

§2 环境部中文件的描述 (190)

§3 数据部中文件的描述 (196)

§4 文件处理语句 (206)

习题六 (241)

第七章 子程序 (245)

§1 使用子程序的优越性 (245)

§2 子程序的编写及其调用方法 (245)

第八章 排 序	(252)
§1 排序过程	(252)
§2 排序语句 (SORT)	(254)
§3 排序实用程序	(261)
习题八	(265)
第九章 程序实例	(266)
§1 卡片校验程序	(266)
§2 建立库存文件程序	(267)
§3 建立帐目文件程序	(270)
§4 简化的工资处理程序	(273)
§5 计算并打印工资报表程序	(279)
§6 师资管理程序——计算工作量且打印报表	(284)
§7 学生成绩档案管理程序	(301)
附录	(321)
一 文件区分符的写法	(321)
二 I-100机COBOL编译命令	(322)
三 保留字	(325)
四 字符的ASCII编码	(333)
参考文献	(334)

绪 论

一、什么是数据处理

所谓数据处理，顾名思义，就是对数据进行加工处理。可以毫不夸张地说，数据处理工作几乎充满着整个人类社会。只要我们稍微留心地观察一下周围的实践活动，就会发现大量的数据处理问题存在。同时也会看到大批的人们在从事解决各种各样的数据处理问题。比如，工资的发放就是一个典型的数据处理问题。会计人员要处理每名职工的姓名、编号、应发工资金额、附加费（如粮贴、夜班补助……）、扣除费（如房租费、水电费、托儿费……），根据这些数据计算出每名职工的实发工资金额，并制定一份工资清单，以便向职工发放工资。这就是数据处理工作的一个典型实例。此外，我们还可以举出大量的数据处理的例子。如：企业管理、行政管理、银行业务处理、仓库的管理、人口调查统计、图书与档案资料的管理和检索、列车的编组与调度、招生考试分数统计……。总之，数据处理在机关、商业、邮政、交通、工厂、银行、学校等部门都有着极其广泛的应用。

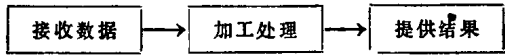
与工程科学计算不同，数据处理有其自己的特征。工程科学计算是对少量的初始数据进行复杂的、反复的多次运算，所得结果寥寥可数。而数据处理则是对大量的初始数据进行极其简单的计算或逻辑判断，所得结果数据却是大量的。如工资发放问题，某单位有2000名职工，每月会计人员在处理这2000名职工的工资时，显然每人的工资计算并不复杂，即：

实发工资 = (应发工资 + 附加费) - 扣除费，然而初始数据和结果数据却是大量的。

其次，工程科学计算中所处理的数据都是数值化的，因此，

这种计算是数值计算，而数据处理所处理的数据不仅仅是数，还可以是人名、书名、地址、产品名、产品型号、列车到站……，即数据处理所处理的数据可以为一串字母，也可以为一串字母数字或专用符号，可见数据处理所处理的数据称为信息更为合适，因此，有时也把数据处理叫做信息处理。而数据处理的任务就是对信息进行收集、存贮、传递、分类、计算以及打印各种报表等。

数据处理的过程大致分成三步：



二、数据处理的发展过程

数据处理经历了人工数据处理，机械数据处理和电子数据处理三个阶段。

人工数据处理就是人借助于简单的工具，如算盘、手摇计算机等，对数据进行手工操作、整理数据、进行计算、制作报表等。显然效率低、速度慢，准确性差，而且花费大量的人力。因此，这是最原始的方法。

机械数据处理，所使用的工具是卡片制表机，包括穿孔、分类、计算和打印等。1890年制作了第一台卡片制表机，当时用于编制人口普查表。人们把要处理的数据在80列卡片上穿孔，再用卡片分类机进行分类和计算，最后将结果再穿在卡片上或打印在纸上。这种处理方法，相对手工操作来说，无论速度还是效率都有了一定的提高，但由于机械速度的提高受到一定的限制，因此效率必然不会很高。

电子数据处理，是目前最先进的数据处理方法，其主要工具是电子计算机及其辅助设备。电子计算机有惊人的速度，可靠的准确性，特别是20世纪60年代以来，存贮数据介质的迅速发展，诸如，磁鼓、磁带、磁盘等大容量的快速存贮介质，为数据处理提供了极为有利的条件。因此，使用计算机进行数据处理，不但

速度快、效率高，同时准确性也得到可靠的保证。

三、COBOL语言的产生和发展

COBOL语言是进行数据处理的高级语言，是当前应用最广泛的计算机语言之一。全称是“COMMON BUSINESS ORIENTED LANGUAGE”，意即商业通用语言。

开始使用电子计算机是靠手编程序(用机器指令编写程序)，如用DJS-183机指令编写如下程序：

01 02 13——传送 即： $(R_2) \rightarrow (R_3)$

06 02 13——加法 即： $(R_2) + ((R_3)) \rightarrow (R_3)$

随后出现了符号语言，用符号语言编写上述程序可写成：

```
MOV R 2, @R 3  ↙
ADD R 2, @R 3  ↘
⋮
```

无论是用机器语言，还是用符号语言编写程序都是一件非常麻烦的事情，要完成一件数据处理任务，需要花费大量的人力编程序及可观的机器时间调试程序。

二十世纪五十年代以来，电子计算机在数据处理领域里越来越广泛地被应用，而手编程序的这种状态，阻碍了数据处理的发展。此时FORTRAN、ALGOL等高级语言已为工程科学计算成功地应用并产生了可喜的效果。这促使人们考虑能否用类似的方法来解决数据处理的程序设计困难，即设计一种适用于数据处理问题的程序设计语言。1959年5月，由美国国防部主持，有企业用户、政府机关用户、计算机厂家等代表参加，举行了一次会议。这次会议专门讨论了数据处理通用语言的必要性、可能性等问题。会后成立了常设机构，研究数据处理语言，并于1959年12月提出了世界上第一个COBOL语言文本，来年由美国政府印刷局发表，这就是所谓的CODASYL(即Conference on Data System Language——数据系统语言会议)COBOL-60文本。

此后,CODASYL对COBOL语言一直不断地发展和完善,陆续地发表了COBOL—61,扩展COBOL—61, COBOL—65, COBOL—68, COBOL—69, COBOL—70, COBOL—73 等一系列文本。因此CODASYL的COBOL语言成为其它COBOL 语言文本的依据。COBOL语言得到了广泛地应用,出现了 功能大小不同的各种COBOL语言。为使COBOL语言具有通用性,对COBOL语言进行了标准化。最早的标准化工作由ANSI (American National Standards Institute——美国国家标准协会) 进行,于1966年8月提出第一个标准COBOL语言文本,经过广泛征求意见,于1968年8月23日经标准化会议正式通过成为美国国家标准COBOL语言,其后ANSI又公布了ANS COBOL—1974。

本书以PDP—11与I—100计算机为背景介绍的COBOL 语言就是ANS COBOL—1974的子集。

COBOL语言按数据处理的主要内容划分为十二个功能块,每个块又按处理能力的强弱分成2—3个等级。于是抽取不同的功能块及其不同等级,便可构成不同规模的COBOL 语言文本。这就为计算机厂家在各种不同规模的机器上,为不同的数据处理业务,实现COBOL编译系统提供了极大的方便。

功 能 块		等 级		
核 心		1		2
表 处 理		1		2
文件处理	顺序文件	1		2
	相对文件	0	1	2
	索引文件	0	1	2
分 类 处 理		0	1	2
报 表 生 成		0	1	
程 序 分 段		0	1	2

续

源 程 序 库	0	1	2
程 序 调 试	0	1	2
程 序 连 接	0	1	2
通 信	0	1	2

上图列出了COBOL语言功能块和等级的划分，其中0级表示空集，即COBOL语言文本，可以没有此种功能。因此最小的COBOL语言文本应包含核心一级，表处理一级及顺序文件一级。

四、COBOL语言的特点

COBOL语言是一种标准化成功的语言，它具有通用性。同一个COBOL程序只需做少量的修改（主要是与硬设备有关的信息）就可以在不同型号的机器上进行编译和运行，不必重新编写程序，这就给用户带来了极大的方便。

COBOL语言是一种类似于英语的程序设计语言，从本质上来说，它接近于英语的自然语言。全部COBOL语句都是用英语词汇来书写的。如把A、B相加就用动词“ADD”。即ADD A TO B。同时COBOL语句的书写规则尽量与英语一致，这对掌握了英语的同志来说无疑将是很方便的，加上它书写和检查进度比较快，直观性强，有自我说明能力等特点，都使得COBOL语言成为一种比较容易掌握的语言。

另外，COBOL程序格式固定，结构模块化，便于存档与查阅。

COBOL语言的不足之处是书写起来冗长、繁琐，编译出来的目标程序质量较差。

第一章 COBOL程序概述

§1 COBOL程序结构

一个COBOL程序由四个部分组成，即标识部，环境部，数据部，过程部。每个部都用一个部标题开头，它们分别是：

IDENTIFICATION DIVISION. (标识部)

ENVIRONMENT DIVISION. (环境部)

DATA DIVISION. (数据部)

PROCEDURE DIVISION. (过程部)

其中标识部指出程序的有关信息，如程序名，作者名，程序的编写日期、编译日期，注解，记事等。环境部指明源计算机名，目标计算机名及源程序中的文件所使用的外部设备等。数据部对源程序中使用的数据进行必要的描述。过程部是COBOL程序的实体，它包含实际要执行的程序语句，正是靠这些语句的执行，使计算机完成各种数据处理任务。

一个COBOL程序必须而且仅须由这四部分组成，并且一定要按上述次序出现。

每个部分可以分成几个节或段，每个节又可以分成几个段。段是由若干句子(SENTENCE)组成，句子由一至若干个语句(STATEMENT)或子句(CLAUSE)组成。语句或子句是COBOL字或COBOL字与基本符号的有序集合，COBOL字又是由COBOL字符组成。因此，一个COBOL程序的组成结构可以粗略地图示如下：

标识部、环境部、数据部都有固定的节和段，只有过程部的节和段由程序员自己来定义。

源程序的修改,但对编译是无意义的,因此,往往空着不用,只要使卡片排列次序与行出现的先后次序一致即可。

2.续行区：占程序纸的第7列，可写如下一些符号：

(1) 空白：正常方式处理源程序的行。

(2) 连字符“—”：语句或子句允许从B区开始一直写到第72列。如果没写完，则允许从下一行的B区接着写。如果前一行的最后一个非空格字符与后一行的第一个非空格字符是属于同一个COBOL字，则在后一行的第7列上写上连字符“—”。

(注: COBOL字在 § 3 中予以定义)

例如SHULIANG是用户定义的COBOL字，则下述两例写法是正确的。

如果在后一例第二行的第7列写上“—”，则表示SHULI-ANGTO为一个COBOL字，这样就错了。

有一个例外，如果是非数字直接量的延续，除了在续行第7列上要写“—”外，还应该在续行的第一个字符前加一个双引号“~”。例如：

1	2	3	4	5	6	7	8	9	10	11	12	15	20	25	65	70	72	75	80
															"HOW DO				
"YOU DO" TO																			

(3) 星号“*”：第7列为“*”的行是注解行，注解行