



实用COBOL 程序设计



殷广济 编

电子工业出版社



实用COBOL程序设计

殷广济 编

電子工業出版社

内 容 提 要

本书不以完整描述COBOL文本为重点,而是突出实用,以数据处理过程中经常遇到的输入输出变换、分组合计、检索、更新、核对处理、表处理等为线索、按照结构化程序设计的原则,结合实例引导读者在较短时间内,通过学习本书就能写出一个完整的COBOL程序,进而,读者结合自己使用计算机所提供的COBOL用户手册,就能编出实际运行的程序。

JS363/18

实用COBOL程序设计

殷广济 编

责任编辑 王惠民

电子工业出版社出版(北京海淀区万寿路)
新华书店北京发行所发行 各地新华书店经营
通县曙光印刷厂印刷

开本: 787×1092毫米1/16 印张: 13.625字数: 314千字

1987年7月第1版 1987年7月第1次印刷

印数: 1~8,000册 定价: 2.75元

统一书号: 15290·475

序 言

COBOL (COMMON BUSINESS-ORIENTED LANGUAGE) 是专门为处理事务数据而建立的计算机程序设计语言,目前在国际上COBOL拥有最多的用户。它的特点是能处理大批量的数据和报表,并提供了多样化的数据结构,使之可以对数据格式进行各种改变。因此,COBOL语言被广泛地应用于政府部门、银行、商业系统、档案系统以及数据库系统等方面。

近年来COBOL语言在国内也日渐受到重视,并开始在一些计算机上得到应用。现在,高档的微型计算机上也大多能运行LEVEL II COBOL。随着我国计算机事业的发展,随着计算机应用的普及,COBOL语言在我国也必将会得到广泛的应用。

本书是在为中日软件中心系统工程师学习班的COBOL语言程序设计的课程编写的讲义基础上改编而成的,除个别部分提及是作为实习用的ACOS计算机以外,基本原理与机型无关。力求内容广泛,深入浅出,通俗易懂。完整的描述COBOL语言文本不是本书的重点,本书的重点在于以进行数据处理时经常碰到的输入输出变换、分组合计、检索、更新、核对处理、表处理等为线索,按照结构化程序设计的原则,引导读者在较短的时间内通过学习本书能写出一个完整的COBOL程序来。读者可结合你所使用的计算机提供的COBOL用户手册,编出实际运行的程序。

本书适宜做COBOL程序设计学习班的教材,也适宜编写COBOL程序的广大科技人员参考。

由于本人水平有限,时间仓促,书中缺点错误在所难免,敬希读者斧正。

殷广济

1985.6

目 录

第一章 概要	(1)
1.1 COBOL语言的特点	(1)
1.2 COBOL语言的历史描述	(2)
1.3 数据和数据处理	(3)
第二章 COBOL 语言 基础	(6)
2.1 COBOL程序结构	(6)
2.1.1 标识部分	(6)
2.1.2 设备部分	(6)
2.1.3 数据部分	(6)
2.1.4 过程部分	(7)
2.2 COBOL基本字符	(7)
2.3 COBOL字	(8)
2.3.1 保留字	(9)
2.3.2 系统名	(10)
2.3.3 用户定义的字	(10)
2.4 常数	(11)
2.4.1 数字常数	(11)
2.4.2 非数字常数	(11)
2.4.3 表意常数	(11)
2.5 COBOL程序的正确写法	(11)
2.6 几点约定	(13)
第三章 标识部分和设备部分	(14)
3.1 标识部分	(14)
3.2 设备部分	(14)
3.2.1 功能和组成	(14)
3.2.2 配置节	(15)
3.2.3 输入输出节	(16)
3.2.4 顺序文件的指定	(17)
第四章 数据部分	(19)
4.1 数据部分概述	(19)
4.2 文件节	(20)
4.2.1 标号子句	(21)
4.2.2 块长子句	(22)
4.2.3 DATA RECORD子句	(23)

4.3 记录描述	(24)
4.3.1 层号子句	(25)
4.3.2 PICTURE子句	(26)
4.3.3 用法子句.....	(31)
4.4 工作存贮节	(33)
第五章 过程部分	(35)
5.1 过程部分概述.....	(35)
5.2 语句的形式和种类.....	(36)
5.2.1 无条件语句	(36)
5.2.2 条件语句.....	(36)
5.2.3 编译指示语句	(36)
5.3 面向文件的输入输出语句	(37)
5.3.1 OPEN语句	(38)
5.3.2 CLOSE语句.....	(39)
5.3.3 READ 语句.....	(39)
5.3.4 WRITE语句.....	(40)
5.4 面向数据的输入输出语句	(41)
5.4.1 ACCEPT语句	(41)
5.4.2 DISPLAY语句.....	(43)
5.5 算术运算语句.....	(44)
5.6 STOP语句.....	(46)
5.7 简单的PERFORM(执行)语句	(47)
5.7.1 PERFORM语句(格式1)	(48)
5.7.2 PERFORM语句的嵌套.....	(49)
5.7.3 PERFORM语句的UNTIL选择项	(50)
5.8 EXIT语句.....	(51)
5.9 GO TO语句.....	(52)
5.10 MOVE语句	(54)
第六章 输入输出变换	(57)
6.1 框图介绍	(57)
6.2 例题1(输入变换)	(59)
6.3 IF语句	(62)
6.4 垂直行距控制.....	(64)
6.5 例题2(输出变换)打印简表.....	(67)
6.6 数据编辑	(72)
6.7 例题3(输出转换)作分组合计	(78)
第七章 文件组织及存取方法	(88)
7.1 存贮介质	(88)
7.2 文件编制	(93)

7.3 文件处理的种类和方式.....	(94)
7.4 存取方式.....	(94)
7.5 索引文件.....	(95)
7.5.1 组织形式.....	(95)
7.5.2 文件的处理.....	(98)
7.6 索引编制文件的定义	(103)
7.7 索引编制文件的输入输出	(103)
7.8 相对编制文件和直编制文件	(107)
7.9 例题4(索引编制文件的更新)	(110)
第八章 条件和条件语句	(116)
8.1 引言	(116)
8.2 关系条件.....	(117)
8.3 类别条件.....	(118)
8.4 符号条件.....	(119)
8.5 条件名条件.....	(119)
8.6 复合条件.....	(122)
8.7 条件语句的嵌套	(123)
8.8 行数(LINAGE)子句.....	(125)
8.9 算术运算符的选择部分.....	(128)
8.10 例题5(核对)	(129)
第九章 顺序文件的批处理	(136)
9.1 引言	(136)
9.2 主文件的维护.....	(138)
9.3 活动率与文件维护	(138)
9.4 数据顺序与文件处理.....	(139)
9.5 文件维护程序设计逻辑.....	(139)
9.6 例题6(更新,顺序存取).....	(140)
第十章 表处理	(147)
10.1 表定义.....	(147)
10.2 OCCURS(重现)子句及其应用.....	(149)
10.3 常数表的定义.....	(153)
10.3.1 对表赋相同初值	(153)
10.3.2 重定义(REDEFINES)子句	(153)
10.3.3 对表赋不同的值	(155)
10.4 表处理.....	(156)
10.4.1 下标法	(157)
10.4.2 指标法.....	(157)
10.4.3 指标数据项.....	(159)
10.5 SET语句.....	(159)

10.6 表检索.....	(161)
10.6.1 顺序查表法.....	(161)
10.6.2 二分查表法.....	(165)
10.6.3 PERFORM 语句的检索功能	(167)
10.7 例题7(表操作的利用).....	(169)
10.8 例题8(表的定义和查表)	(177)
通过 DAO 调用,通 的保留字表	(183)
写它们的数据库应IC字符代码对照表.....	(187)
附录3 语句格式总结.....	(188)
附录4 习题.....	(193)
参考文献	

第一章 概 要

1.1 COBOL 语言的特点

COBOL (COmmon Business-Oriented Language) 语言是出现较早、应用最广泛的一种面向商业、事业的数据处理的程序设计语言, 经过二十多年的发展, 其功能不断得到完善扩充, 现已为当今世界上广大计算机用户所接收的标准化了的语言。

在西方一些经济发达的国家, 计算机在国民经济各个领域得到了广泛的应用, 使用 COBOL 语言的用户远比使用其它语言的为多, 大约占60%左右。COBOL 语言使得计算机的应用与人类社会的活动发生了十分密切的联系, 它在推进计算机工业的发展以及开拓和扩大计算机的应用领域中起着极为重要的作用。以下表为例, 可见(日本)各种语言的使用情况。

语言种类 使用百分比 统计年分						
	汇 编	ALGOL	FORTTRAN	COBOL	PL/I	其 它
1973	33.0	0.2	7.5	44.6	1.7	12.7
1974	27.0	0.3	8.0	44.8	2.4	17.5
1975	26.4	0.1	7.5	50.6	2.7	13.7
1976	21.7	0.1	8.2	52.8	3.9	12.3
1977	19.6	0.3	7.3	56.3	3.1	13.4

COBOL 语言之所以成为最普遍使用的强有力的程序设计工具, 与它所具备的数据构造功能和程序英语化这两个主要特点分不开。

COBOL 语言是专门为处理商业数据处理中所必须的大量数据而建立的, 对数据有详尽的描述:

数据类型 可以描述数值型, 字母型, 字符型, 编辑型……

数据存放形式 可以是二进制的, 十进制的, 浮点的……

数据关系 可以是独立数据项, 组合数据项, 层次结构。

文件使用 顺序组织, 索引组织, 直接编制……

COBOL 的数据表示虽具有一定的商业应用味道, 但和一般用于科学计算的语言(主要偏重于算法和公式翻译、数据结构较简单)不同, 用起来还是比较灵活的。数据的构造特性使它能将应用文件、数据库、数据通信、联机系统和各类信息系统贯穿在一起, 并能较好地将对数据处理和计算机应用中所发展起来的一些主要技术融会在语言的模块中, 加上它与操作系统以及计算机系统的发展有着不可分割的相互适应的联系, 使得 COBOL 语言在现代计算机

系统中占据了一个关键性位置,所以深入地学习 COBOL 语言也有助于我们对计算机系统整体和数据处理技术的了解。

和其它语言相比, COBOL 语言是最接近自然语言英语的一种计算机语言。为了便于非专业人员掌握和运用,它除了同一般的高级程序设计语言一样使用一些关键字外,还定义了大量的可选保留字,而选用这些保留字书写的程序很接近英语口语, COBOL 程序的英语化特征使得它容易阅读,容易理解。阅读 COBOL 程序如同阅读一般文章一样,所以用 COBOL 语言编写的程序便于交流和推广。再加上 COBOL 程序除了设备部分之外基本上不依赖于具体的计算机,从而使它成为一种很好的具有通用性和可维护性的程序文件。

学习 COBOL 语言与学习 FORTRAN 语言不同。对于 FORTRAN 语言,学习几条语句就可以开始简单的程序设计,可以边学习,边编制程序上机实习。而 COBOL 程序结构严谨,无论一个程序怎样简单也要有四大部分,即标识部分、设备部分、数据部分和过程部分,因此对于 COBOL 语言来说,非要在四大部分以及其它内容学完以后才能编写程序。一开始我们将要回避一些涉及很多“内部处理”的概念,这些问题会分散初学者的注意力,从而忽视程序设计的一些更为重要的问题。

1.2 COBOL 语言的历史描述

COBOL 是一种解决商业上的数据处理问题的计算机程序设计语言。1959 年在美国政府的赞助下,计算机厂商、高等院校的代表在一起成立了一个委员会——CODASYL (Conference on DATA SYstem Language) 数据系统语言协会,代表们一起研究了多种版本的 COBOL。那时几乎每台大计算机系统都有自己的版本。1960 年 4 月 CODASYL 提出了一个 COBOL 初稿并于当年完成 COBOL 规范,由 CODASYL 于 1961 年公布。这就是人们通常称的“COBOL-60”。62 年出版了修改后的文本“COBOL-61”。到了 1965 年编辑了更加精细的“COBOL-Edition 1965”,这个文本 1968 年 8 月为美国国家标准化协会所接收,成为美国国家标准文本——ANSI COBOL X3.23-1968,简称为 ANSI COBOL-68。1972 年国际标准化组织 (ISO) 通过了国际标准 COBOL 文本“ISO COBOL-72”,它与 ANSI COBOL-68 大同小异。该文本受到美、英、法、日、苏等 21 个会员国承认。1972 年日本制定了 JIS COBOL (日本工业规格 COBOL)。

与此同时, COBOL 在继续发展,人们也在努力扩充 COBOL 语言本身的规格。自从 1959 年成立 CODASYL,发表第一个 COBOL 文本以来,纵观二十多年 COBOL 的发展史,可以说该语言的功能随着计算机技术的变化和更新而逐步扩大、增强,这也正是 COBOL 之所以能连续地保持其旺盛的生命力和赢得越来越多的用户的原因所在。从七十年代起,随着高速度大容量外存储器的出现, COBOL 从文件存取方式发展到数据库存取方式,另一方面,随着数据通信技术与计算技术结合的日臻完善, COBOL 从成批处理型态发展到联机乃至网络处理型态。1974 年 ANSI 在 ANSI COBOL-68 的基础上增加了通信、调试和程序间通信的功能,并把随机存取相关存取分为和索引存取两个模式后,发展为 ANSI COBOL X3.23-1974 文本,1978 年 ISO 宣布 ANSI COBOL X3.23-1974 文本即为 ISO COBOL-78,1980 年日本发表了修订版的 JIS COBOL。

ISO COBOL-72 由一个核心和七个功能处理模块组成 (如表 1.1 中实线所示), 1978 年

公布的ISO COBOL-78在国际上已广泛使用。现在常把具有数据库存取方式或适用于联机,网络处理型态的COBOL称为高级(ADVANCE) COBOL。国际上许多著名的计算机系统都已不同程度地实现了高级COBOL的功能,从而有力地推动了信息处理应用软件的发展。

ISO COBOL-78由一个核心和十一个功能处理模块组成(见表1.1)。核心和功能处理模块可再进行分级,1级低于2级,2级低于3级,除了空级以外,低级为较高级的真子集。

COBOL语言的实施依赖于不同的机器处理。最小的COBOL标准由一级核心(描述所允许的字符集、数据结构、基本输入输出、数据的传送、算术运算及程序间转移等操作),一级表处理和一级顺序存取模块组成。最大的COBOL标准由核心及各功能模块的最高级组成。

目前我国也已制定了一个国家标准的COBOL文本。国内已有的一些关于COBOL语言的讲义和书籍,大都是以美国国家标准ANSI COBOL-68为基础的。现阶段国外各类计算机大都采用ISO COBOL-78这个新版本。在国际上COBOL语言已为广大的应用程序员所熟知,人们运用COBOL语言这个工具已经创造和积累了大量的、极其宝贵的软件资源和信息资源。这是许多人的智慧和才干的结晶,我们应当把它很好地吸收过来,通过标准化和兼容的办法吸取别国的先进技术,使我们在软件方面的国际交往中处于较有力的地位。

表1.1 ISO COBOL 语言结构图

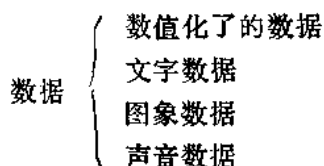
核 心	功 能 处 理 模 块										
	表处理	顺序存取	随机存取	排序	报表打印	程序分段	程序库	索引存取	调试	通信	程序间通信
2 级	3 级	2 级	2 级	2 级	2 级	2 级	2 级	2 级	2 级	1 级	2 级
	2 级		1 级	1 级	1 级	1 级	1 级	1 级	1 级	1 级	1 级
1 级	1 级	1 级	空	空	空	空	空	空	空	空	空

COBOL程序英语化的特点是通过COBOL语言的保留字和用户字与英语很好的沟通而实现的。如果我们能把COBOL语言与汉语很好地结合起来,形成我国特有的汉语COBOL(应该做得到)那我们就能更普遍地推广这种程序设计语言,推广计算机的使用,造就一支数量可观的程序设计大军,使我国应用软件设计的规模和水平、软件产品的数量和质量进入世界先进水平的行列。

1.3 数据和数据处理

COBOL 是面向数据处理的语言,那么什么是数据呢?

我们生活在数据的海洋里,对于什么是数据这个问题可能很不以为然。通常说到数据时,我们一般的理解就是具体的数字,如83,62...等等,细心的读者会发现用¥123.56来表示人民币的钱数和通常的123.56就有了细微的差别。从数据处理的角度来看,还有一类非数值数据,如字符、图象等。我们说的数据处理中的数据应包括下述内容:



这就把数据概念大大扩充了，文字、图象、声音都成了处理对象。但是这里只处理数值化了的数据和文字数据。在 COBOL 语言中，程序处理的对象是数字和一些字符，如人名、序号、地址、库存、文章内容等等。这些都属于数据的范围，都可成为 COBOL 程序处理的对象。

在人类社会活动中有个包括人力、实物、货币等在内的流动，这是一个物理流。与此同时也伴随一个反映物理状态和特征的信息流，象发票、报表、汇兑等。人们根据这些信息才能进行生产指挥，发布命令（当然命令也是一种信息）。这些信息构成了有机整体，有人把现在的社会叫信息社会是很确切的。

数据的实质就在于表征某一事件的量值和属性，是信息的具体体现。数据经过加工以后就变成信息。

所谓数据处理就是对这些信息进行一切必要的加工，包括收集、分类、排序分析、运算、比较、传送、印刷等处理，内容广泛，工作量很大。数据处理的特点是对大量的原始数据执行极其简单的运算（有时甚至还不计算）或逻辑判断，其产生的结果数据量也很大。这和工程科学计算有本质的区别，对数据处理而言，一般占用计算机的 CPU 时间较短，大量进行的是输入输出操作。

因此，有人笼统地把数据称作信息，数据处理就是对信息进行收集、存贮、综合分析和加工处理。

在 COBOL 程序中，如何表示数据呢？这有两种形式，即变量和常数。

变量是在 COBOL 程序中不断更新其值的数据。之所以要不断更新其值其原因有二，一是输入，二是计算或加工。为了在程序中能表示一个变动的数据，我们就要给变量起名字，并按名字来引用数据。注意，起名字要根据课题中有关数据的意义来取名，这样可增加程序的可读性。

常数在 COBOL 中又称字面，字面又分数值字面和非数值字面，按习惯我们还称常数。

采取什么方法组织数据并对它加以管理和应用呢？对好的数据处理语言一个突出的要求就是它能够描述数据的原始关系，刻划数据的自然属性。即能表达出数据的本来面目但又不是简单的对应，这就要发挥计算机的特长把数据处理的应用实际和自动化综合在一起。

对 COBOL 而言，数据组织的最大单位是文件。根据不同的意义把数据组织成一个一个的文件，文件中包含有若干个记录。所谓记录，就是一组具有独立逻辑含义的，按一定规律排列起来的数据之集合。这个集合表征并刻划了某一具体事物。如一个人的工资记录，其中包括基本工资附加工资（奖金、补贴）和扣除（房租、水电、托儿费、借支）。记录是一个有机的整体，COBOL 中用一个记录描述来定义一个记录的类型、层次结构、所占内存空间的大小等。

学好 COBOL 语言的关键之一就是要弄清数据处理中有关文件、记录及数据项目等的基本概念。

COBOL 程序纸

表 原 制 表 者 制 表 日期 年 月 日

顺序编号	行号	程序	程序标识符
1	001010	IDENTIFICATION DIVISION.	
2	001020	PROGRAM-ID.	
3			
4	002010	ENVIRONMENT DIVISION.	
5	002020	CONFIGURATION SECTION.	
6	002030	SOURCE-COMPUTER.	
7	002040	OBJECT-COMPUTER.	
8			
9	003010	DATA DIVISION.	
10	003020	WORKING-STORAGE SECTION.	
11	003030	01 WK1 PICTURE IS 999.	
12	003040	01 WK2 PICTURE IS 999.	
13			
14	004010	PROCEDURE DIVISION.	
15	004020	START-RUN.	
16	004030	COMPUTE WK1 = 200 + 100.	
17	004040	COMPUTE WK2 = 200 - 100.	
18	004050	STOP RUN.	
19			
20			

注释区

页数

CCTS

顺序编号区 指示区

第二章 COBOL 语言基础

2.1 COBOL程序结构

先请大家看一个为了理解COBOL程序结构而写的最简单的COBOL程序。至于如何分析问题，描述算法，印出结果等细节，以后再介绍。

〔例题0〕

程序规格书

(1) 程序名: EX 0

(2) 使用的计算机: 编译、执行时均用ACOS。

(3) 处理说明: 做数值100和200的加法、减法运算后就结束该程序。

一个完整的 COBOL 程序，规定必须有四个部分组成，无论程序怎么简单也不能省，次序也不准颠倒。

2.1.1 标识部分

标识部分 IDENTIFICATION DIVISION 最简单，它必须有的部分是程序名，这是用来标识该程序的。相当于一个 COBOL 程序的标题。此外还允许有其它一些成分，用来写程序员姓名、编写日期等，主要是提供一些阅读和归档的信息。

请注意部分后面要有一个句号，句号后边要有一个以上的空格。实际上不管在什么地方用到句号，其后都必须留一个以上的空格。

英文单字之间要用一个以上的空格分开。

2.1.2 设备部分

设备部分 ENVIRONMENT DIVISION 也有人叫做环境部分。用它来说明 COBOL 源程序在编译时所使用的机器组成，执行目标程序时所使用的机器组成，即指明运行该程序所需要的计算机环境，并且在程序中使用的文件和外部设备之间建立联系，说明所使用的诸文件的属性、定义有关的开关等。

当改变计算机类型时，这部分需要改动，但改动的数量较少。换句话说，由于有关计算机环境的成分已由各处集中到该部分了，因此其它的部分就不依赖于计算机环境，这也是 COBOL 程序的一大优点。

2.1.3 数据部分

数据部分 DATA DIVISION 是 COBOL 程序中主要的部分之一，它对程序中作为处理对象所使用的数据之名称（文件名、记录名、初等项名、组合项名等）及其有关属性（结

构、位数、类型)要一一加以定义和说明。它和下面要讲到的过程部分结合起来构成COBOL程序的最重要的部分。

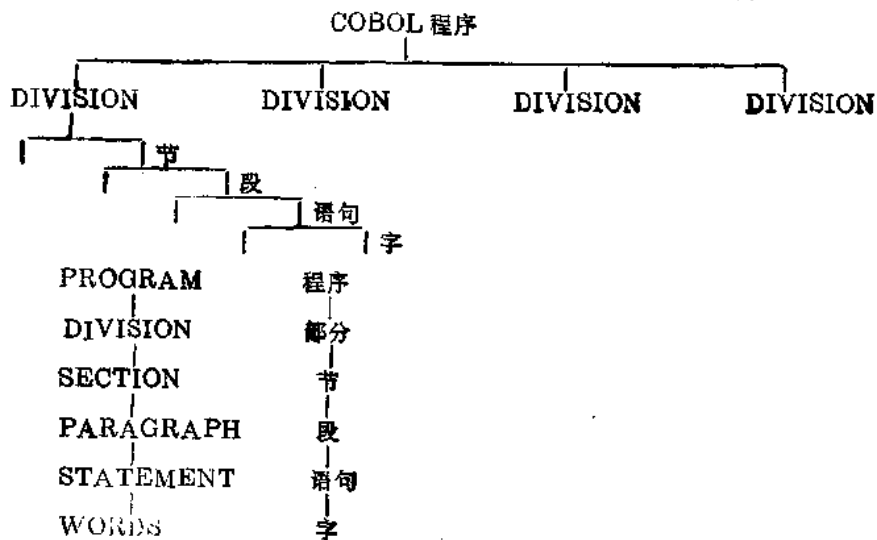
通常,这部分的描述是比较长的,约占整个源程序的 $1/3 \sim 1/2$,这也是COBOL源程序和其它语言的源程序的不同之处。

2.1.4 过程部分

过程部分PROCEDURE DIVISION也是COBOL程序的主要部分之一,它是具体的实现逻辑思维、描述信息加工过程的。例如对输入的数据和程序中的数据进行加工,并把结果输出等就是在过程部分实现的。

一个程序的执行是从过程部分顺序从上到下、从左到右进行的。但执行有转移功能的语句时除外。这里“执行”是指执行源程序经过编译后所对应的目标程序,而不是直接执行源程序。

从这个例子可以看到,每一个部分都是以部标题开始的,部分的下面分节(SECTION),节下面分段(PARAGRAPH),段下面有句子(SENTENCE),句子由语句(STATEMENT)或子句(CLAUSE)组成,句子、语句、子句都由COBOL字(WORDS)组成,COBOL字由COBOL允许字符(CCHARACTER)组成。COBOL程序是一种树形结构。



2.2 COBOL 基本字符

COBOL语言允许的字符共有51个,即:

A—Z 26个英文字母

0—9 10个数字

特定字符15个,即:

$\left. \begin{array}{l} + \\ - \\ * \\ / \end{array} \right\}$ 运算符

$\left. \begin{array}{l} = \\ > \\ < \end{array} \right\}$ 关系符

·	} 标点符号	(	} 括 号
,		)	
;		¥	} 币 符
”		␣	
()			␣ 空 格

日本产的计算机通常还有47个日语假名符号。

需要指出的是，一些特定字符有多义性，其确切含义要由上下文来定。例如“.”作为结束符号后边要有一个以上的空格，作为十进制小数点用时左右不能是空格。“=”可作为等号来看，有时又是赋值记号，按作用来分一下有如下几类：

作标点符号用	·	比较用	>
	,		<
	;		=
	”		
	(		
	)		
算术运算符用		编 辑 用	B
	+		0
	-		+
	*		-
	/		OR
	**		DB
			Z
			¥
			*
			,
			.
			/

2.3 COBOL字

一个 COBOL 字是不超过 30 个字符的字符串，由 26 个英文字母，数字 0—9 和连字符“-”组成。通常要求以英文字母开头，并且至少含一个英文字母。连字符不能出现在一个字的最前面和最后面。

COBOL字又可分成保留字 (reserved words) 和用户定义的字 (user defined words) 及系统名 (system-name) 三大类。

2.3.1 保留字

保留字是具有专门语义的字, 只能按 COBOL 语法规则来使用它, 用户不能用其作为数据名、段名、节名等。但是在 COBOL 程序中, 如果用 “ ” 把保留字括起来, 它就不再是保留字而成为非数值常数了。

COBOL中保留字有下列几种。

1. 关键字 (KEY WORDS)

这一些保留字是语句的核心, 当含有该字的格式被引用时, 必须正确地书写该字, 不能省, 所以叫关键字, 可把它们分为三种:

(1) COBOL动词, 如ADD, READ, WRITE等。

(2) 虽非COBOL动词但在格式中也不能省, 如ADD语句中的TO, GIVING等。

(3) 具有特殊意义的字如SECTION, POSITIVE等

有些关键字有缩写形式, 例如PICTURE可写成PIC, 其意义相同。

2. 任选字 (OPTIONAL WORDS)

这些字是辅助字, 在语句中只起辅助作用, 不写也可以, 写上去只是为了增加程序的可读性, 写就不能写错, 也不允许用别的字来代替。例如:

02 ABC PIC IS 9 (5) .

其中IS是任选字, 不写也可以。其功能等价于:

02 ABC PIC 9 (5) .

关键字和任选字是COBOL文本中规定的, 数量比较多, 且任选字和关键字可能是同一个字, 如TO即是一例, 其区分依赖于上下文。

3. 表意常数 (FIGURATIVE CONSTANT)

表意常数不是通过字面本身, 而是通过其含义来表达数据的, 它是一种特别的常数, 有固定的名称和意义, 单数和复数的形式等价, 单数和复数形式的运用仅仅出自英语语法上的需要。

ZERO, ZEORS, ZEROES

代表一个或者若干个0。

SPACE, SPACES

代表一个或者若干个空格。

QUOTE, QUOTES

代表一个或者若干个引号, 但不能作定界符使用。

HIGH-VALUE, HIGH-VALUES

表示一个或若干个计算机系统所规定的相比次序中, 具有最高机内编码值的字符串, 即全“1”码, 所有二进制位为1, 或16进制位为F。

LCW-VALUE, LOW-VALUES

表示一个或若干个具有最低机内编码值的字符串, 即全零码。

ALL

表示一个或若干个组成字面的字符串, 字面可以是非数值常数或者是除ALL以外的表意