

国际标准化组织推荐文本

程序设计语言COBOL



科学出版社

73.8721
7

国际标准化组织推荐文本

程序设计语言 COBOL

陈胜凡 朱致远 译

吕义忠 校

科 学 出 版 社

1980

内 容 简 介

本书是国际标准化组织推荐的程序设计语言 COBOL 标准文本,它反映了国际数据处理界的要求,能描述种类繁多的数据处理问题,也能用于科学计算。

读者对象是计算机软件工作者,计算数学工作者及大学有关专业师生。

ISO Recommendation R 1989 PROGRAMMING LANGUAGE COBOL

Switzerland, 1972

国际标准化组织推荐文本 程序设计语言 COBOL

陈胜凡 朱致远 译

吕义忠 校

*

科学出版社出版

北京朝阳门内大街 137 号

石家庄地区印刷厂印刷

新华书店北京发行所发行 各地新华书店经售

*

1980 年 2 月第 一 版 开本: 787×1092 1/32

1980 年 2 月第一次印刷 印张: 13 1/2

印数: 0001—80 字数: 309,000

统一书号: 13031·1170

本社书号: 1636·13-1

定 价: 1.38 元

简 史

国际标准化组织 (简称 ISO) 推荐的 R1989 号文本——《程序设计语言 COBOL》——是由该组织技术委员会“计算机和信息加工”(ISO/TC97)起草的。该技术委员会秘书处现由美国国家标准学会主持。

有关这方面的工作导致采用 ISO 推荐文本 No. 1989 草案。1970 年 7 月, 该草案曾在 ISO 全体成员国中巡回审阅过。

除了有少量的编辑方面的修改外, 这个草案已经被下列成员承认:

澳 大 利 亚	德 国	葡 萄 牙
奥 地 利	爱尔兰	西班牙
比 利 时	以色列	瑞 典
巴 西	意大利	瑞 士
捷克斯洛伐克	日 本	土耳其
丹 麦	荷 兰	英 国
法 国	新西兰	美 国

本 ISO 推荐文本草案用通讯方式递交 ISO 委员会 后者决定接受它作为 ISO 推荐文本。

前 言

经过六年时间的发展,这本 ISO 推荐书终于反映了国际数据处理界的要求。本书的主要目的是要反映一种足够丰富的语言,以便描述种类繁多的数据处理问题,同时也能准确地反映各会员国的需要。我们尽可能地做到对正在发展过程中的各国的 COBOL 标准作一个统一的解释。

COBOL 是一种工业语言,它不是哪一个公司或哪一个公司集团的私有物,也不是哪个组织或集团组织的私有物。因此,不论是 COBOL 委员会还是对 COBOL 作出了贡献的人都没有对这个语言的正确性和功能以及它的程序设计系统提出明确的或含蓄的保证。当然,他们也不会在这些方面承担任何责任。

目 录

第 一 部 分

第 一 章	引言	1
第 二 章	各级核心和各级模式中的 COBOL 元素表 (甲表)	8
第 三 章	各 COBOL 元素所属核心级和模式级的表 (乙 表)	34
第 四 章	定义	53
第 五 章	总的语言考虑	76
第 六 章	语言造句格式表	113

第 二 部 分

第 七 章	一级核心	135
第 八 章	二级核心	191
第 九 章	一级表处理	233
第 十 章	二级表处理	238
第十一章	三级表处理	242
第十二章	一级顺序存取	251
第十三章	二级顺序存取	271
第十四章	一级随机存取	293
第十五章	二级随机存取	305
第十六章	一级排序	317
第十七章	二级排序	327
第十八章	一级报表打印	334

第 十 九 章	二级报表打印	358
第 二 十 章	一级程序分段	383
第二十一章	二级程序分段	388
第二十二章	一级程序库	393
第二十三章	二级程序库	396
索引		399
英中词汇对照表		413

第一部分

第一章 引言

1.1 范围

这本 ISO 推荐文本 (以后简称 ISO COBOL) 建立了 COBOL 程序的格式和解释, 旨在倡议一种高度可互换的 COBOL 程序, 用于各种自动数据处理系统之中。

1.2 语言的结构

为了更有效地使用 COBOL 语言, 在这一推荐文本中我们已经将 COBOL 说明书的历来编制——标识部分、设备部分、数据部分以及过程部分——加以校正 (关于这些部分的一般讨论见 5.4—5.7 段)。新的编制方向是围绕功能处理模式的概念的。定义了一个核心 (关于内部处理的) 和七个功能处理模式。这些功能处理模式是: 表处理, 顺序存取, 随机存取, 排序, 报表打印, 程序分段和程序库。

核心分为两级。低的一级包含了基本的内部处理所需要的元素, 而且是高的一级的真子集。每一功能处理模式有两个或更多的级。在所有的情况下, 较低的级是同一模式的较高的级的真子集。此外, 某些功能处理模式以空集作为它们的最低级。

这种编制方法提供了裁剪说明书所必需的灵活性, 使之

适应于多种不同数据处理应用的需要。同时,这种编制,在确定一个给定的编译系统所需要的标准元素上的能力,自然地要比以往的编制来得强一些。

1.3 ISO COBOL 的编制

本书分为两部分。第一部分包括引言,各级核心和各级模式中的 COBOL 元素表(以下简称甲表),各 COBOL 元素所属核心级和模式级的表(以下简称乙表),定义,总的语言考虑的讨论以及语言造句格式表。第二部分对于核心和功能处理模式的每一级都有一章。各章均包含了 COBOL 的详细技术说明,它是以 1965 年出版的 CODASYL COBOL 中的说明为依据的。

1.4 怎样使用本书

应当注意本书是从几个不同的角度来加以考察的。除目录和索引外,上述甲表和乙表都起着说明的作用。甲表用来确定任一级的内容。

甲表有 ISO COBOL 中每一元素的详细分析,并且是按级来编排的。此外有一个正文参照号,指出在本书的什么地方能找到这个元素的说明。例如,为确定高级表处理的内容,就要按甲表来引用那个模式。在“总的语言考虑”里,列出了所有的 COBOL 元素,还有设备部分,数据部分描述体以及关于表处理的过程部分的动词。由于级是一个套一个的,为了确定最高级的内容,就必须考察整个模式。如果要得到一个特定元素的更详细的信息,可以按该元素右边列出的正文参照号去找正文的技术说明部分。

为了确定一个特定的语言的特征在哪一级或哪些级出

现,就要用乙表。乙表详细指出了 ISO COBOL 的全部元素以及它们是在哪些级中出现的。此外,每出现一个元素,就附有正文参照号,对于那种并不完全包含在某一级中的元素,则充分详细地规定每一子元素的位置。如果读者还需要知道关于任一级的特定元素的使用的更为详细的信息,右边的正文参照号可以引导你去找技术说明部分。例如,为指出 SELECT 子句出现在哪里,就要用到乙表。我们将看到 SELECT 子句出现在低级的顺序存取和低级的随机存取的模式中。由于 SELECT 子句的某些选择项仅仅在高级顺序存取和随机存取模式中出现,它的子元素是分开列出的。对于每一个子元素,也有一个正文参照号。

对于一个元素或者它的一部分,如果它不止在功能处理模式的一个级中出现,我们都作了双份说明。

当用甲表来确定某一级的内容时,就需要确定在正文中别的什么地方某一具体的元素也用到了,便可以参照乙表,如果必要的话,可从那里援引详细的技术说明。

涉及总的语言考察或概念的一般信息,可从目录和索引中寻找正文。

为最后确定一个 COBOL 编译系统的内容,要用到后面的框图。框图是一种图表表示,它把 COBOL 分为各种功能处理模式和两级核心。并且,框图显示了每一功能处理模式内部的级别,以及核心内部的级别。每一个级,都有一个章号。总之,甲表,乙表,目录,索引以及框图为学习 ISO COBOL 提供了几个途径。

1.5 如何确定一个 COBOL 编译系统

一个 ISO COBOL 编译系统的要求可以从后面的框图中

看出来。一个 COBOL 编译系统包括了确定的功能处理模式和核心的级, 如果它们按本书所定义的办法执行, 将被认为符合于一个 COBOL 编译系统的要求。

作为一个说明, 下面我们给出最小和最全的 COBOL (指 COBOL 编译系统) 的定义:

最小 COBOL 是由每个功能处理模式和核心中的最低级组成的。由于存在着空集, 最小 COBOL 是由核心, 表处理以及顺序存取的最低级组成。最全的 COBOL 是由每个功能处理模式和核心中的最高级组成。

在 COBOL 说明书中, 有某些语言元素, 它们的实行依赖于硬部件的特殊类型。编译为一个给定 COBOL 编译系统规定所要求的最小硬件配置以及这种硬件配置所提供的硬部件。某些功能处理模式和核心包含了一些元素, 它们的实行要求特殊的硬件。如果没有这些元素就必须加以说明, 这不会使得一个 COBOL 编译系统成为非标准的。这一类元素如下:

CLOSE...REEL

CLOSE...UNIT

CLOSE...NO REWIND

OPEN...NO REWIND

OPEN...I-O

OPEN...REVERSED

USE...I-O

MULTIPLE REEL

MULTIPLE UNIT

WRITE...ADVANCING 记忆名

WRITE...INVALID KEY

开关状态

USE...REEL

USE...UNIT

ACCESS MODE

PROCESSING MODE IS SEQUENTIAL

如果系统有能力实现某些 COBOL 元素的功能, 就不必把这个特殊的元素包括在源程序中了. 一旦这个并非必要的元素确实出现在源程序中了, 它就必须被编译系统所接受. 然而, 如果这个元素并不导致产生目标代码, 则在 COBOL 源程序内就不需要任何相应的语句来实施这个功能.

有一些元素, 它们的语法和效能是系统按 ISO COBOL 定义的范围来确定的.

其语法部分地由系统确定的元素是:

ASSIGN

SOURCE-COMPUTER

OBJECT-COMPUTER

计算机名

存储器大小

开关名

设备名¹⁾

FILE-LIMIT

RERUN 子句

VALUE OF

ACCEPT 和 DISPLAY, 数据一次传送的最大长度

ENTER

USE LABEL PROCEDURE

程序库名

R 界

1) 设备名可指硬设备及系统提供的软设备, 视具体情况而定. ——译者注

其效能部分地由系统确定的元素是:

FILE-LIMIT

ACTUAL KEY

SYNCHRONIZED

COMPUTATIONAL

INDEX

REDEFINES, 用在相同数据区的不同描述

USAGE IS DISPLAY, 用于带有运算符号的数值项的情形

非数值项的比较

ACCEPT 和 DISPLAY, 数据一次传送的最大值
算术表达式

USE ERROR PROCEDURE

STOP RUN, 并非所有打开的文件都关闭

而且, 上述的语言元素或者上述的功能均不属于 ISO COBOL 中的一部分时, 不会使得一个 COBOL 编译系统成为非标准的。即使它蕴涵地扩充了系统的保留字, 并且使某些跟 ISO COBOL 一致的程序的编译超出了正常的范围, 上面的说法也是成立的。

字符替换也可以使某些跟 ISO COBOL 一致的程序的编译超出正常的范围, 有关这方面的讨论参见 5.3.1 段字符集。

1.6 简化符号

在框图中, 甲表, 乙表以及索引内部已经采用了一种简化符号, 用来指出核心或功能处理模式中任一级的级别, 以及模式划分的级数。这个简化符号由这样一些从左到右的符号组成: 一个表示级别的数字, 三个字符的记忆名以及由逗号隔开

的两个数字表示的模式和核心的最低级和最高级。级数为零表示空级。例如, 2 NUC 1, 2 表示这是核心的第二级而且核心是由两个非空集的级组成的。再举一例, 2 SRT 0, 2 指出这是包含三级的排序模式的第二非空级, 它们中的最低级是空级。

用在简化符号中的记忆名如下:

记忆名	意 义
NUC	核心
TBL	表处理
SEQ	顺序存取
RAC	随机存取
SRT	排序
RPW	报表打印
SEG	程序分段(以下称分段)
LIB	程序库(以下称库)

核 心	功 能 处 理 模 式						
	表处理	顺 序 存 取	随 机 存 取	排 序	报表打印	分 段	库
2NUC1,2	3TBL1,3	2SEQ1,2	2RAC0,2	2SRT0,2	2RPW0,2	2SEG0,2	2LIB0,2
第 8 章	第 11 章	第 13 章	第 15 章	第 17 章	第 19 章	第 21 章	第 23 章
	2TBL1,3		1RAC0,2	1SRT0,2	1RPW0,2	1SEG0,2	1LIB0,2
1NUC1,2	第 10 章	1SEQ1,2	第 14 章	第 16 章	第 18 章	第 20 章	第 22 章
第 7 章	第 9 章	第 12 章	空	空	空	空	空

图 1 ISO COBOL 结构图

第二章 各级核心和各级模式中的 COBOL 元素表(甲表)

这一章列出了 ISO COBOL 中各级核心和各级模式中全部的元素。右边是正文参照号, 指出在正文相应级的哪一章中(除非另有说明)有该元素的详细说明。

一级核心 1 NUC 1,2: 第七章

	正文参照
语言概念	5.3
组成字的字符	4.2
0, 1, ..., 9	
A, B, ..., Z	
- (连字符或减号)	
标点字符	4.2
" 引号	
() 括弧	
空白或空字符	
. 句号	
编辑字符	4.2
B 空字符	
0 零	
+ 加号	
- 减号	
CR 贷方	
DB 借方	

Z	零抑制符	
*	核对保护符	
\$	货币符号	
,	逗号	
.	句号	
	允许的字符替换	5.3.1.1
	分隔符	7.2.1
	名字特征	
	一个名字的最大长度是 30 个字符	5.3.2.1.1
	数据名必须从字母字符开始	7.2.2
	名字必须是唯一的并且不能被限定	7.2.2
	过程名	5.3.2.1.2.3
	常字	
	数值常字	
	——允许 1 至 18 个数字	5.3.2.2
	非数值常字	
	——允许 1 至 120 个字符	5.3.2.2
	象征常数	5.3.2.1.2.4
	ZERO	
	SPACE	
	HIGH-VALUE	
	LOW-VALUE	
	QUOTE	
	专用寄存器(TALLY)	5.3.2.1.2.5
	保留字	5.3.2.1.2.7
	基词	
	选择的保留词	

连结词	
分隔分号和标点符号, 逗号和分号, 是不允许的	7.2.1
标准格式	
顺序号	5.8.2.1
A 界	
部分头	5.8.3.1
节头	5.8.3.2
段头, 段名	5.8.3.3
数据部分描述体	5.8.4
B 界	
段	5.8.3.3
数据部分描述体	5.8.4
行的延续	5.8.2.2
仅仅非数值常字可以被延续	7.2.4
数据描述体的写法	5.8.4
标识部分	5.4
PROGRAM-ID 段	7.3.3
AUTHOR 段	7.3.2.1
INSTALLATION 段	7.3.2.1
DATE-WRITTEN 段	7.3.2.1
SECURITY 段	7.3.2.1
REMARKS 段	7.3.2.1
设备部分	5.5
SOURCE-COMPUTER 段	7.4.1.1
计算机名	
OBJECT-COMPUTER 段	7.4.1.2