

程序设计语言PL/I



國防工業出版社

程序设计语言 PL/I

〔美〕C. P. 莱 特著

《程序设计语言 PL/I》编译组编译

国防工业出版社

内 容 简 介

程序设计语言在使用和发展电子计算机中起着很重要的作用。本书是根据〔美〕C. P. 莱特所著的《程序设计语言 PL/I》一书编译而成。它较详细地介绍了国际上较为通行的一种程序设计语言 PL/I。

全书共有七章，三个附录。第一章为引论；第二章是对 PL/I 中一些基本要素作出定义；第三章讨论语句；第四章讨论用以刻画数据特征的各种表征；第五章对程序结构、子程序、函数等作了详细的讨论；第六章专论输入与输出；第七章讨论异步操作与表处理。附录中，对内部构造函数、伪变量、中断条件、形象指明表、精确度法则以及转换法则等进行了讨论。

本书是供电子计算机领域中的程序工作者使用的参考书。同时，对于从事电子计算机的研究、制造与使用的广大工作人员与大专院校师生，也有一定的参考价值。

本书系根据1968年美国McGRAW-HILL图书公司出版的《THE PROGRAMMER'S PL/I》(A Complete Reference) (Charles Philip Lecht 著) 一书编译而成

程序设计语言 PL/I

〔美〕C. P. 莱特著

《程序设计语言 PL/I》编译组编译

国防工业出版社出版

北京市书刊出版业营业许可证出字第074号

新华书店北京发行所发行 各地新华书店经售

国防工业出版社印刷厂印装

787×1092¹/₁₆ 印张 107/8 245千字

1972年12月第一版 1975年7月第二次印刷 印数：8,501—24,500册

统一书号：15034·1287 定价：0.95元

毛主席语录

洋为中用。

学习有两种态度。一种是教条主义的态度，不管我国情况，适用的和不适用的，一起搬来。这种态度不好。另一种态度，学习的时候用脑筋想一下，学那些和我国情况相适合的东西，即吸取对我们有益的经验，我们需要的是这样一种态度。

我们不能走世界各国技术发展的老路，跟在别人后面一步一步地爬行。我们必须打破常规，尽量采用先进技术，在一个不太长的历史时期内，把我国建设成为一个社会主义的现代化的强国。

译者的话

程序设计语言在使用与发展电子计算机中起着重要的作用。为了使电子计算机能够更好地在我国社会主义建设事业中发挥作用，我们根据毛主席“洋为中用”的教导，编译了该书，供电子计算机领域中的同志们参考。

PL/I 语言是国外比较流行的一种程序设计语言，它可以兼用于科学计算、商业、实时控制等领域。

全书共分七章。第一章为引论。其中，对 PL/I 语言及使用 PL/I 语言编写程序时所用的一般法则作了介绍，并给出了一个综合性的例子（样本程序）；第二章为定义。其中，给出了 PL/I 中一些基本要素的定义；第三章讨论语句。其中，对 PL/I 中的各种语句都给出了详细的定义；第四章讨论用以刻画数据特征的各种表征；第五章对程序结构、子程序、函数等作了详细讨论；第六章专论输入、输出；第七章讨论异步操作及表处理。此外，在书末还给出了三个附录。分别对内部构造函数、伪变量、中断条件、形象指明表、精确度法则以及转换法则进行了讨论。

原书有些内容过于烦琐，还有些内容在一定程度上反映出资本主义社会里在科学技术领域中的实用主义。在阅读时，我们必须采取批判的态度。为此，我们对原书作了部分删节。

为便于阅读，我们把第三章与第四章的内容按性质作了重排，在个别地方还加了译注。但限于水平，译文虽经多次修改，仍会有不少缺点和错误，恳请读者批评指正。

目 录

第一章 引 论

一、PL/I 语言的定义与结构	7
定义	7
结构	7
基本符号	8
界限符	8
基字	9
一般的语句形式	12
PL/I 语句的类型	13
二、PL/I 程序的编写	13
PL/I 程序的形式	13
编写法则和技巧	14
标点符号	16
注解	17
三、PL/I 程序一例	18

第二章 定 义

标识符	22
名	22
外过程	22
外名	22
内分程序	22
内名	22
变量	23
常数	23
实算术常数	23
虚算术常数	24
字符行常数	24
字位行常数	24
数组	25
数组名	25
下标名	25
数组的断面	25
结构	25
结构名	25
受限名	27
结构数组	28
下标受限名	30
函数引用 (或函数调用)	30
表达式	31
算术表达式	32

行表达式	33
比较表达式	35

第三章 语 句

一、语句形式索引	37
二、语句的详细描述	40
赋值语句	40
转向语句	46
如果语句	46
空语句	48
循环首部语句	48
调用语句	54
入口语句	56
返回语句	59
说明语句	60
中断语句	61
恢复语句	62
发信号语句	63
显示语句	64
开语句	64
关语句	67
取入语句	68
送出语句	69
读语句	71
写语句	72
重写语句	73
定位语句	74
启封语句	75
删除语句	75
格式语句	76
开始语句	76
结束语句	76
过程语句	77
分配语句	82
解除语句	84
等待语句	85
延迟语句	86
出口语句	86
停语句	86

第四章 表 征

一、表征索引	87
二、表征对标识符的应用	87

上下文说明的应用	88
缺省法则的应用	88
文件表征的应用	89
三、详细的表征格式	90
二进位 (BINARY) 或十进位 (DECIMAL)	
表征	90
实 (REAL) 或复 (COMPLEX) 表征	91
定点 (FIXED) 或浮点 (FLOAT) 表征	91
形象 (PICTURE) 表征	92
精确度表征	93
字符 (CHARACTER) 或字位 (BIT)	
表征	94
标号 (LABEL) 表征	95
任务 (TASK) 表征	95
事件 (EVENT) 表征	95
指示型 (POINTER) 表征	96
内部构造 (BUILTIN) 表征	97
相似 (LIKE) 表征	97
初值 (INITIAL) 表征	97
维数表征	99
内部 (INTERNAL) 或外部 (EXTERNAL)	
表征	100
静态 (STATIC)、自动 (AUTOMATIC)	
或受控 (CONTROLLED) 表征	101
带指示变量的受控 (CONTROLLED(p))	
表征	101
次级 (SECONDARY) 表征	103
文件 (FILE) 表征	103
顺序 (SEQUENTIAL) 或直接 (DIRECT)	
表征	103
缓冲 (BUFFERED) 或无缓冲	
(UNBUFFERED) 表征	104
输入 (INPUT)、输出 (OUTPUT) 或听用	
(UPDATE) 表征	104
流式 (STREAM) 或记录 (RECORD)	
表征	104
打印 (PRINT) 表征	105
反向 (BACKWARDS) 表征	105
专用 (EXCLUSIVE) 表征	105
信息附标 (KEYED) 表征	106
外圈环境 (ENVIRONMENT) 表征	106
入口 (ENTRY) 表征	107
返回 (RETURNS) 表征	108
同族 (GENERIC) 表征	109
正常 (NORMAL) 或异常 (ABNORMAL)	
表征	110
用值 (USES) 表征	111
置值 (SETS) 表征	112
区域 (AREA) 表征	113
单组 (CELL) 表征	114
定义 (DEFINED) 表征	115

列式 (ALIGNED) 或包式 (PACKED)	
表征	116

第五章 程序结构、子程序、函数

一、分程序的等级	117
分程序的动用与结束	118
动态后继关系	118
名的作用域	119
存貯分配	121
控制转移	123
条件前缀的使用	124
二、子程序与函数	124
参数	125
变元	126
ENTRY 表征的使用	126
参数的分配	127
把结果传送给动用分程序	127

第六章 输入/输出组织

一、流式传输	129
数据表	129
格式表	131
表式传输	136
数据式传输	138
编排式传输	140
二、记录式传输	140
缓冲文件	141

第七章 异步操作与表处理

一、异步操作	142
任务名与事件名	142
任务任选	143
优先级与延迟语句	143
任务的结束	144
二、表处理	144
有基变量与指示变量	144
未被分配的有基变量的使用	144
已被分配的有基变量的使用	148
有基变量与缓冲输入/输出	149
具有自定义的界及长度的有基结构	149

附录 I 内部构造函数与伪变量

一、内部构造函数	151
算术函数	151
行函数	154

处理数组函数	155
中断条件函数	156
表处理函数	157
杂函数	157
二、伪变量	157

附录 I 中断条件

计算条件	159
输入/输出条件	160
表处理条件	160
程序校验条件	160
程序员命名的条件	162
系统动作条件	162

附录 II 形象指明表、转换法则与精确度法则

一、形象指明表	163
数值形象指明表	163
字符形象指明表	164
利用漂移符号编排数值数据	164
二、转换法则	165
算术转换	165
类型转换	166
三、精确度法则	166

附：中英名词对照表

第一章 引 论

这是一本供程序工作者使用的书。

本书对在相当广泛的计算机系统中实现的“PL/I 语言”作了详细的描述。但是，已经实现了很多 PL/I 编译程序，它们的使用设施（指令组等等）和本书中所提供的不应该有本质上的不同。

本书的目的是作为参考之用。它并不是一本“自学”读物。

一、PL/I 语言的定义与结构

定义

PL/I 是一种可以用来编写计算机程序的语言。

PL/I 是程序设计语言 I (Programming Language I) 的缩写。这个简单而又非常一般的名称，指的是一种具有通用目的的语言。它可以兼用于商业、科学以及实时等方面。并且，不管程序员的经验多么丰富，所用的技巧有多高，这种语言都是可用的。

PL/I 是一组符号与一组法则。与此相关的是该语言的描述（其形式与使用）所特有的一组定义。

有一个和 PL/I 语言相关联的计算机程序，这个程序称为“PL/I 编译程序”。所有用 PL/I 语言写成的程序，在作为目标程序执行以前，都必须由 PL/I 编译程序进行处理。

因此，准备使用 PL/I 语言解题便包含着了解本书中所描述的该语言的各个组成部分的形式与使用。此外，必须有一个可用的 PL/I 编译程序，它把任何用 PL/I 写的程序翻译成用机器码（计算机本身的语言）写的相应的程序。

本书并不讨论用来翻译 PL/I 程序的各种 PL/I 编译程序（例如，它们如何工作等等），那一部分内容作为资料汇编在别的书中，以及汇编在计算机制造者所出版的手册中。

结构

PL/I 语言的结构和用 PL/I 语言所写的程序的结构不同。本节讨论前者，而在第一章二“PL/I 程序的编写”中讨论后者。

为了讨论的目的，我们把语句取作语言的最大单位，一个 PL/I 语句和一个英语句子类似。这就是说，它是一个语法上和语义上完备的结构。

PL/I 语句由一些标识符、常数以及界限符组成。

PL/I 标识符分成两类：

PL/I 定义的基字

程序员定义的标识符

（基字是标识符的特例，这是因为一个程序员定义的标识符，可以等同于一个基字，而二者的区别仅在于它们在语句中的位置不同。）

关于构造程序员定义的标识符及常数的一些法则，包含在第二章“定义”中。第二章还包含了一些构造表达式的法则，其中又利用了标识符与常数。

界限符或者指明一个初等动作（例如，加），或者提供标点符号，或者兼而有之。

界限符与基字都在本章下文中来定义。

上面所指的每一个语句元素，都是由一个或多个基本符号组成的，这些基本符号总起来构成PL/I语言的字符集。下面给出语言的基本符号。

基本符号

1. 字母字符——A B C D E F G H I J K L M N O P Q R S T U V W X Y Z 加上@ # \$ ●

字母字符用来构造标识符和行。

2. 数字——0 1 2 3 4 5 6 7 8 9

字位（二进位数字）——0 1

数字用来构造数、标识符和行

字位用来构造二进位数和行

注：29个字母字符与10个数字总起来被看作字母数字字符。

3. 特定字符

符 号	名 称	符 号	名 称
=	空格（表示成□） ^①	%	百分比号
+	等号或赋值号	;	分号
-	加	:	冒号
*	减	¬	非
/	星号或乘号	& ^②	与
(	斜竖号或除号		或
)	左括号	>	大于
,	右括号	<	小于
.	逗号	_(下方划线)	下破折符_(用于标识符中)
'(或'')	十进位(或二进位)小数点或句号	?	问号
	引号		

① 原书该符号为□，现改为惯用的□。——译注

② 原书该符号为&，现改为惯用的&。——译注

界限符

正如下面所表明的那样，对于很多特定的字符，以及它们的一些双字符组合，赋予特定的作用，以作为界限符。

1. 算术运算符

符 号	含 义	符 号	含 义
+	加法或正号	*	乘 法
-	减法或负号	/	除 法

● 原书该符号为\$，现改为惯用的\$。——译注

2. 比较运算符

符 号	含 义	符 号	含 义
>	大于	$\neq$	不等于
$\neg >$	不大于	$\leq$	小于或等于
$\geq$	大于或等于	<	小于
=	等于	$\neg <$	不小于

3. 行运算符 (字符行及字位行)

符 号	含 义	符 号	含 义
$\neg$	否("非")		或
&	与		并置

4. 分隔符及其界限符

符 号	用 途	符 号	用 途
,	供分隔表元素用	□	一般分隔符
;	供结束语句用	'(或'')	用来括住行常数
=	用作赋值语句中的赋值号	.	用来分隔受限名中的各项
:	用以结束标号及条件前缀; 还用于 维数表征	$\neg >$	用来分隔指示变量与这样受限的名
		/ *	用来表示注解的开始

为了便于阅读起见, 可以在标识符内部利用下破折符来分隔一些组成字或缩写, 例如, 代替 PAYNO 而写成 PAY_NO。在使用时, 把下破折符看作标识符的一个不可分割的部分, 亦即, PAY_NO 和 PAYNO 并不等价。

由序列/ * 开始的注解可以代替一个界限空格。

基字

基字是语言中具有固定含义的标识符。基字的识别, 不仅要根据它们的形式 (一列字母), 而且还要根据它们在语句中的位置。这就是说, 基字并不是 PL/I 中的 '保留' 字; 程序员定义的任何标识符, 即使它和一个基字相同, 也可以用于从语法上看它并不处在一个基字的任何位置上。

使用取入语句的一个简单的例子是:

```
GET FILE(FILE);
```

GET 和 FILE 的首次出现都是基字。而 FILE 的第二次出现却根据它相对于序列 GET FILE 的位置被 (被编译程序) 认作是一个程序员定义的数据文件的名, 或者简称为一个文件名。

基字可以按作用分为若干类, 简单说来, 它们是:

1. 语句标识符。
2. 语句任选——这些基字描述某些语句专有的特征。
3. 表征——这些基字出现在说明语句中, 用来描述由一个给定的标识符所代表的数
据项。

4. 条件基字——这些基字出现在中断语句，发信号语句以及条件前缀中，用来命名程序中中断条件。

5. 内部构造的函数名——由PL/I编译程序作为语言的一部分所提供的函数过程名。

为读者方便计，我们把PL/I的基字在这里列在一起，以便读者可以在第一章三的样本程序中来识别它们。

注意：接在基字之后的括号中的各项是一些可允许的缩写。

语句标识符

ALLOCATE	IF
BEGIN	LOCATE
CALL	ON
CLOSE	OPEN
DECLARE(DCL)	PROCEDURE(PROC)
DELAY	PUT
DELETE	READ
DISPLAY	RETURN
DO	REVERT
END	REWRITE
ENTRY	SIGNAL
EXIT	STOP
FORMAT	UNLOCK
FREE	WAIT
GET	WRITE
GO TO	

语句任选

BY, TO, WHILE	循环首部语句
BY NAME	赋值语句
ELSE	如果语句
EVENT	调用语句, 显示语句以及输入/输出语句
IN	分配语句及输入/输出语句
MAIN	过程语句
OPTIONS	过程语句
PRIORITY	调用语句及输入/输出语句
RECURSIVE	过程语句
REPLY	显示语句
SET	分配语句及输入/输出语句
SNAP	中断语句

SYSTEM	中断语句及过程语句
TASK	调用语句
THEN	如果语句

——下面列出的任选用于输入/输出语句——

COLUMN	KEYTO
COPY	LINE
DATA	LINESIZE
EDIT	LIST
FILE	NOLOCK
FROM	PAGE
IDENT	PAGESIZE
IGNORE	SKIP
INTO	STRING
KEY	TITLE
KEYFROM	

表 征

——数据类型表征——

AREA	FILE
BINARY(BIN)	FIXED
BIT	FLOAT
CELL	LABEL
CHARACTER(CHAR)	PICTURE(PIC)
COMPLEX(CPLX)	POINTER(PTR)
DECIMAL(DEC)	REAL
ENTRY	TASK
EVENT	

——次表征——

ABNORMAL(ABNL)	DIRECT
ALIGNED	ENVIRONMENT(ENV)
AUTOMATIC(AUTO)	EXCLUSIVE
BACKWARDS	EXTERNAL(EXT)
BUFFERED	GENERIC
BUILTIN	INITIAL(INIT)
CONTROLLED(CTL)	INPUT
DEFINED(DEF)	INTERNAL(INT)

KEYED	SECONDARY
LIKE	SEQUENTIAL
NORMAL	SET
OUTPUT	STATIC
PACKED	STREAM
POSITION(POS)	UPDATE
PRINT	USES
RECORD	VARYING(VAR)
RETURNS	

条 件 基 字

AREA	NAME
CHECK	OVERFLOW(OFL)
CONDITION	RECORD
CONVERSION(CONV)	SUBSCRIPTRANGE(SUBRG)
ENDFILE	SIZE
ENDPAGE	TRANSMIT
ERROR	UNDEFINEDFILE(UNDF)
FINISH	UNDERFLOW(UFL)
FIXEDOVERFLOW(FOFL)	ZERODIVIDE(ZDIV)
KEY	

在附录 I 中给出了作为 PL/I 的内部构造函数名的基字，以及每一函数的完备描述。

一般的语句形式

说明语句除外的所有的 PL/I 语句都可以在前面冠以一个或多个标号以及（或者）若干条件前缀：

条件前缀：……：标号：……：语句；

条件前缀在第五章中详细描述。

PL/I 语言中有各种各样的语句格式；可以给出如下的代表形式：

语句标识符[标识符/表达式][任选表]；

方括号表示形式中的任选元素。

（上面形式中的斜竖号 / 表示二中择一；它并不是语句的组成部分。）正如在第三章“语句”中所看到的，有这种代表形式的若干变形。

一些例子是：

CALL 标识符；

DISPLAY 表达式 REPLY 标识符；

PROCEDURE OPTIONS(MAIN) RECURSIVE；

上述代表形式的一些重要的例外是：

1. 赋值语句没有语句标识符基字。

标识符 = 标识符 / 常数 / 表达式;

作为一个赋值语句, 这是根据 = 出现在上下文中的地位来识别的。赋值号左方及右方的标识符可以是变量、数组或结构名, 并且可以是可用的下标名或受限名。此外, 赋值号左方的标识符可以是一个伪变量。关于伪变量的讨论, 参看附录 I。

2. 空语句只由前面冠以一个或多个标号的分号组成:

标号:;

3. 如果语句及中断语句二者都是复合语句, 在它们内部包含另一语句。这些语句的形式分别是:

1) IF 表达式 THEN 语句; ELSE 语句;

2) ON 条件名 语句;

注意: 除了如果语句必不可少的 THEN 子句内部所包含的语句外, 任选的 ELSE 子句本身也包含另一语句。

4. 说明语句及分配语句二者都可以包含任意多个“标识符-表征”组。相邻两组之间用一个逗号隔开:

语句标识符 *id* 表征……, *id* 表征……, ……;

或者, 每一个“标识符-表征”组也可以在一个各别的说明语句或分配语句中指明出来, 在每一种情形, 其作用都完全相同。

解除语句同样可以指明任意多个标识符, 但是却没有相关的表征。

这里所讨论的语句形式并没有包括所有可能的语句, 特别是, 并没有包括属于上述诸形式之一的一些语句的特定的变形, 例如, 具有 SYSTEM 及 (或) SNAP 任选的中断语句。讨论的目的是使读者在这时熟悉一下 PL/I 语句的比较一般的形式, 以便为阅读第一章三中的样本程序做准备。

PL/I 语句的类型

类	型	作	用
赋值		实现数值计算及逻辑运算, 并且把一个值赋给一个变量或一组变量	
控制		控制各别语句或语句组有条件地执行, 以及改变语句的正常执行顺序	
数据说明		指明一个变量所代表的值的种类	
错误控制与调试		指明中断条件的处理, 启动或模拟这样的中断	
输入/输出		实现外存与内存之间的数据传输	
程序结构		指明若干程序段或子程序的作用域以及它们之间的等级(阶序)关系	
存贮分配		在程序执行时用以得到内存的有效使用	
异步操作控制		为并行处理及实时操作提供控制设施	

二、PL/I 程序的编写

编写任何计算机程序都预先假定, 对于用数据处理运算表示的待解的问题有一个全面的了解。假定这一条件已满足, 则下面的讨论便可作为编写 PL/I 程序的导引。

PL/I 程序的形式

一般说来, 一个 PL/I 程序分为若干称作分程序的逻辑段 (语句的集合)。分程序有

两类：过程分程序与开式分程序。这些分程序分别包括在过程语句与结束语句以及开始语句与结束语句中。宜于把过程分程序简称为过程。

在分程序内部可以包括任意多个低层的分程序，包含在一个分程序内部的另一个过程称为一个内过程。开式分程序总是内分程序。这是因为它们必须包含在（在某一层上）一个过程的内部的缘故。

一个最外层的或者“不被包含的”过程称为外过程。一个完整的PL/I程序由一个或多个外过程组成，其中每一个外过程可以包含任意多个彼此分开的或嵌套的内分程序。

在一个过程或开式分程序内部所使用的变量、数组以及结构等等的名，或者是明显地定义于出现在该分程序任何位置的说明语句中，或者，在没有这样的说明时，就必须按形式或上下文隐含地定义。一般说来，一个被说明的名在它被说明的分程序内部是已知的（也就是说，是可以被引用的），并且除了它再次被说明以外，在任何被包含的分程序内部也是如此。

PL/I分程序由一些语句组成，其中任一语句都可以是另一个分程序的首部语句。过程和开式分程序的重要区别在于：过程的执行必须明显地由一个语句的执行所调用，而开式分程序则是由正常的顺序控制流程进入的。

在程序的各个语句间（甚至在语句内部）可以自由地散布一些注解，以便使源程序表能够更好地作为内部资料进行交流之用。

所有的PL/I语句都可以附以一个或多个语句标号，但是说明语句的任何标号都要被编译程序掠过。引用一个语句可以通过引用它的任何一个标号来实现。

一个标识符，或者明显地借助一个说明语句，或者按上下文由于它作为一个语句的前缀出现而被说明为表示一个语句标号。语句标号有时说成标号常数，借以区别于标号变量。标号变量是一个这样的标识符，它明显地被说明具有一个标号表征，并且可以借助赋值语句对它给出一个标号常数。第一章三中的PL/I程序，提供了一个使用标号变量的例子。

一个PL/I程序的执行是从初始外过程的第一个语句开始的，并且在分程序内部按语句依次连续进行，但是要绕过任何被包含的过程。然而，语言的某些语句却具有改变执行顺序的能力。

程序的执行正常地是在控制达到初始过程的结束语句时结束。

图1说明了一个比较复杂的（可是它是随意的）PL/I程序的结构。每个方括号代表一个过程或一个开式分程序，每一个都具有界限在其中被说明的名的作用域的能力。

试注意：图1代表一个具有两个外过程，五个内分程序，并且有三层嵌套的PL/I程序。为清楚起见，图中的每一个结束语句都明显地指明它所界限的分程序的标号。但是，这种用法却是任选的。

编写法则和技巧

PL/I程序不必写成特定的编码形式，这是因为语言的编码形式完全是自由的缘故。这就是说，在一个PL/I编码“行”中对于一个PL/I语句的不同的元素并没有固定的范围（域）。并且对语句或注解也没有固定的开始位置。

为便于按键穿孔誊录及安排程序列表格式起见，应该把一个PL/I编码行看作包含80个字符位置或80列。PL/I编码可以出现在这些列中的任何位置。