



电子计算机介绍

怎样使用计算机

(DJS-6 机算法语言及使用说明)

北京工业大学计算站 编



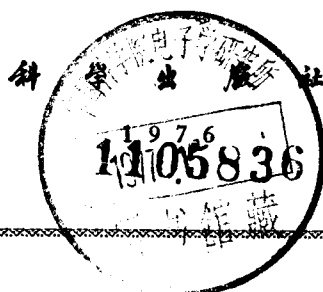
科学出版社

73.872
142

怎样使用计算机

(DJS-6 机算法语言及使用说明)

北京工业大学计算站 编



内 容 简 介

本书以 DJS-6 机作为典型进行剖析,对计算的各个步骤均有详细论述,是一本深入浅出、通俗易懂的普及读物。

全书包括三大部分。第一部分是算法语言,介绍用算法语言编写程序的方法,还结合实例指出编写程序时应注意的问题;第二部分是上机操作,介绍源程序和数据的书写格式、穿孔技术、上机操作步骤以及查寻错误等;第三部分包括习题解答及诸多表格,供使用参考。

本书可供各行各业从事计算的人员参考。

怎样使用计算机

(DJS-6 机算法语言及使用说明)

北京工业大学计算站 编

*

科学出版社出版

北京朝阳门内大街 137 号

北京新华印刷厂印刷

新华书店北京发行所发行 各地新华书店经售

*

1976 年 11 月第 一 版 开本: 787×1092 1/32

1976 年 11 月第一次印刷 印张: 7 3/4 插页: 1

印数: 0001—116,500 字数: 175,000

统一书号: 13031·518

本社书号: 760·13—1

定 价: 0.55 元

前 言

目前电子计算机已在工农业生产和科技领域得到广泛的应用, 程序设计语言正越来越多地为广大工农兵和工程技术人员所掌握。

为了适应日益发展的形势, 几年来先后为工农兵学员、工人师傅、解放军战士、生产管理人员和工程技术人员举办了数十期算法语言推广班, 本书就是在推广过程中经过几次总结与修改而写成的。

本书可以作为学习 DJS-6 机 ALGOL 语言的教材或自学读物, 对象是没有接触过电子计算机和算法语言的广大读者。因此, 编写时力求做到: 深入浅出, 条理清楚, 例子详尽, 上机操作具体。

本书按经过我站修改过的二版编译程序编写, 内容共分十章。前九章介绍用 DJS-6 机 ALGOL 语言编写程序的方法, 第十章是上机操作。由于 DJS-6 机 ALGOL 语言与 ALGOL 60 基本相同, 因此, 本书的前六章与第九章亦可作为学习 ALGOL 60 或其他与 ALGOL 60 接近的语言之用。

在编写本书和修改二版编译程序的过程中, 京字 115 部队的《DJS-6 机算法语言介绍》和修改纸带为我们提供了宝贵的资料, 各兄弟单位给了我们多方面的支持, 在此表示衷心的感谢。

书中难免存在错误和不足之处, 希望读者提出意见与建议, 共同为推广使用计算机而努力。

目 录

绪言	1
第一章 基本概念	5
§ 1 基本符号	5
§ 2 数	7
§ 3 标识符、简单变量与类型说明	9
§ 4 标准函数	13
§ 5 源程序的结构	14
习题	16
第二章 简单语句	17
§ 1 赋值语句、简单算术表达式	17
§ 2 条件语句、条件算术表达式	26
§ 3 转向语句、空语句	33
§ 4 复合语句	40
习题	45
第三章 数组、循环	47
§ 1 下标变量与数组说明	48
§ 2 循环语句	53
§ 3 循环语句的进一步讨论	63
习题	72
第四章 布尔变量、开关	74
§ 1 布尔变量与布尔表达式	74
§ 2 开关说明与命名表达式	79
习题	84

• i •

第五章 分程序	86
§ 1 分程序概念	86
§ 2 标识符的作用域	87
§ 3 引入分程序的优越性	91
习题	94
第六章 过程	97
§ 1 一般过程	97
§ 2 函数过程	120
§ 3 关于过程说明和过程调用的一些问题	131
§ 4 几个常用算法	136
习题	147
第七章 标准过程	149
§ 1 输入	149
§ 2 输出	150
§ 3 其它	160
习题	161
第八章 代码语句	162
§ 1 代码语句的一般形式	162
§ 2 代码语句的入口与出口	165
§ 3 要注意的一些问题	166
§ 4 例子	167
第九章 DJS-6 机算法语言语法规则	172
§ 1 语法规则	172
§ 2 与 ALGOL 60 的区别	176
第十章 操作使用	179
§ 1 源程序的编制	179
§ 2 纸带穿孔	185
§ 3 上机操作——基本操作	193

§ 4	上机操作——调试	220
§ 5	上机操作——特殊情况下的操作	212
§ 6	控制台变量与已知地址量	213
§ 7	鼓的利用	220
附表 1	55 型电传机编码(修改后)与字符对照表	224
附表 2	语法错误分类表	225
附表 3	运算错误类型表	228
附表 4	操作命令一览表	229
	习题解答	231

绪 言

一 DJS-6 机简介

DJS-6 机(D, J, S 分别是电子、计算机和数字三词汉语拼音的第一个字母)又称 108-乙机,是我国自行设计、制造的一台中型晶体管通用数字计算机。运算方式为二进制并行运算,指令为单地址指令。可作定点半字长运算与浮点全字长运算,或双倍字长的乘除运算。定点运算平均每秒七万二千次,浮点运算平均每秒五万四千次。

DJS-6 机的主要部份的功能与规模如下:

1. **输入器** 有光电机输入与电传打字机输入两种。前者的功能是将穿孔纸带上代表程序或数据的信息送入机器的存贮器,速度为 800~1000 个字符/秒;后者一般用于从控制台上输入信息,输入速度很慢。

2. **存贮器** 它是机器的主要部分,用来存放程序或数的装置,有内存贮器与外存贮器两部份。

DJS-6 机配有两组大磁芯体作为内存贮器,它们分别装在两个机柜里。每组磁芯体含有 2×16384 个半字长单元(以下谈到的容量,都是指半字长单元),因此,共有 $2 \times 2 \times 16384$ 个单元。每个单元都是由 24 颗磁芯串成的一个磁芯串。由于每颗磁芯有两种不同的状态,因此,可用以表示二进位的两个不同的数字 0 和 1。

DJS-6 机配有四台磁鼓作为外存贮器,编号为 0, 1, 2, 3 鼓,每鼓容量为 2×16384 个单元,鼓的转速为 1500 转/分。有

些机器还配有磁带贮存器,最多可装八台磁带机,编号为0,1,2,3,4,5,6,7台.采用1吋(25.4 mm)的具有16条信息道的磁带.每条磁带的容量为 $32 \times 2 \times 16384$ 个单元.外存贮器的特点是容量大,但是,由于存、取的速度较慢,一般算题都利用内存贮器,外存贮器只用作存放机器所需的固定信息(如编译程序,程序库等)或当计算大型题目时弥补内存贮器的不足.

这里要特别指出的是,存贮器有一个重要的性质:从它的每一个单元中取出信息后,该单元仍保留其原有信息.但是,当送入一个新的信息后,原来的信息即被破坏,而被新送入的信息所取代.

3. **控制器** 它的功能是对程序中每条指令进行分析、判别,决定机器要执行那一种操作,然后对有关部份发出命令,是指挥机器各部份的“神经中枢”.

4. **运算器** 各种运算操作在运算器内进行.

5. **输出器** 一般有窄行快速打印机、电传打字机和宽行打印机三种.快速打印机打印速度为15行/秒;电传打字机可以输出52种字符,输出速度为400个字符/分钟;宽行打印机每行打印160个字符,每秒钟打印5—10行.有些机器还配备X-Y绘图仪和快速穿孔输出机.

二 什么是算法语言?

在电子计算机产生初期,人们在使用它进行计算时,是直接根据机器指令来编制程序的,这样编出的程序通常称为手编程序.采用手编程序有很大的缺点,首先是工作量太大,机器几分钟能算完的问题往往要用几个星期来编程序,因而需要一支庞大的编程序的队伍.其次,手编程序是由一大堆枯燥的数字组成的,很不直观,不便于阅读和检查,而且非常容

易出错。最后,手编程序是用机器指令编成的,它对机器的依赖性很大,同一个题目,改用另一种型号的机器来计算时,原有的程序就不适用了,必须重新编制程序。

能否把编制程序的大部份工作让机器去完成,也就是说,能否只按通常的数学表示写出计算步骤,然后交给计算机去工作?随着计算机的迅速发展,这个问题得到了肯定的答复。近廿年来,已建立了许多接近数学表示的程序设计语言。在科学计算中最常用的有 FORTRAN 语言与 ALGOL60 两种。这里, FORTRAN 是 *Formula Translation* 的缩写‘即公式翻译之意; ALGOL 是 *Algorithmic Language* 的缩写,即算法语言之意。

本书介绍的是 DJS-6 机 ALGOL 语言,它与 ALGOL60 基本相同¹⁾。

掌握算法语言要比掌握手编程序的技巧容易得多,它大大地促进了计算机的应用。另外,国产机器大都配备了与 ALGOL60 接近的语言,因此,掌握了一种机器上的算法语言之后,也就容易掌握另一种机器上的算法语言。这就使编制的程序在很大程度上具有通用性,只要稍加修改就能在另一型号的机器上使用。

使用算法语言解题,大大减少了编制程序的时间,同时也便于人们阅读、检查、修改。例如,计算一个底面半径为 R 、高为 H 的圆柱体体积和侧面积,用 DJS-6 机算法语言来编写,就是下列形式:

‘BEGIN’

‘REAL’ R, H, V, S ;

READR (R, H);

$V := 3.141592654 \times R \times R \times H$;

$S := 2 \times 3.141592654 \times R \times H$;

1) 为了使读者在学习本语言 (DJS-6 机 ALGOL 语言的简称) 的同时对 ALGOL60 有所了解,在本书的前六章中,如无特别声明,则所叙述的内容既符合 ALGOL60 又符合本语言。若只适用于本语言,则声明“在本语言中”;同样,若只适用于 ALGOL60,则声明“在 ALGOL60”中。

OUTPUTR (V, S)

END

用普通的语言来表示,意义如下:

开始

实型数 R, H, V, S ;

(从光电输入机上通过数据纸带)读入 R, H ;

计算 $\pi R^2 H$ 的值并把结果送到 V 中;

计算 $2\pi RH$ 的值并把结果送到 S 中;

(在快速打印机上)输出计算结果 V, S

结束

这与我们平常所写的计算公式极为接近,因此,算法语言是易于为广大工农兵群众和工程技术人员所掌握的。

用算法语言编写的程序称为源程序。我们用计算机进行计算时,先要编制好源程序,把它穿孔成纸带,再输入到机器中去,利用存放在机器内的编译程序将源程序翻译成机器指令程序(称为目标程序),然后让机器进行计算。

DJS-6 机算法语言的编译程序正在不断完善,功能不断扩充,各套编译程序相继问世,为使用算法语言解题创造了良好的条件。本书介绍的 DJS-6 机 ALGOL 语言使用的是经过我站修改与补充的二版编译程序。修改与补充包括以下几个方面:

1. 增加了宽行输出功能以及宽行有故障时变窄行输出的措施;
2. 增加了数据检验功能;
3. 增加了用电传命令选择扩体方案的功能;
4. 向 ALGOL60 进一步靠拢;
5. 增加了区域溢出等方面的检查功能;
6. 其它。

第一章 基本概念

§1 基本符号

大家知道,英语是由 26 个字母和标点符号构成的,我们称这些字母和标点符号为英语的基本符号。任何语言都有它自己的特定的一套基本符号。DJS-6 机算法语言也一样,是由一些基本符号构成的,它的基本符号共有 86 个¹⁾。今后用本语言编写程序时,只允许使用这 86 个符号,在此范围之外的符号不能使用。

算法语言是用来描述运算过程(算术运算与逻辑运算)的,因此,它的基本符号不但应该包括字母和标点符号(算法语言中称为分隔符),而且还必须包括数字、算术运算符、逻辑运算符、逻辑值等各种与运算过程有关的成份。本语言的 86 个基本符号由四大类组成:

(1) 字母(共 26 个)

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

(2) 数字(共 10 个)

0 1 2 3 4 5 6 7 8 9

(3) 逻辑值(共 2 个)

'TRUE' (真) 'FALSE' (假)

(4) 定义符(共 48 个)

1) 在ALGOL60 中,基本符号共 116 个(其中字母是由大,小写共 52 个英文字母组成),书写形式略有不同。

1. 运算符(共 21 个)

a. 算术运算符(共 6 个)

+ - × / // ↑

b. 关系运算符(共 6 个)

'GR' (相当于 >) 'LS' (相当于 <)

'GQ' (相当于 ≥) 'LQ' (相当于 ≤)

'NQ' (相当于 ≠) =

c. 逻辑运算符(共 3 个)

'OR' (或) 'AND' (与) 'NOT' (非)

d. 顺序运算符(共 6 个)

'IF' (如果) 'THEN' (则) 'ELSE' (否则)

'FOR' (对于) 'DO' (做) 'GOTO' (转向)

2. 分隔符(共 9 个)

:(冒号) ; (分号) ,(逗号)

.(小数点) ₁₀ (小拾) :=(赋值号)

'UNTIL' (直到) 'STEP' (步长)

'WHILE' (当...的时候)

3. 括号(共 9 个)

() 两个构成一对圆括号

[] 两个构成一对方括号

{ } 两个构成一对行括号

'BEGIN' (开始)与 'END' (结束)构成一对语句括号,

'CODE' (代码)

4. 说明符(共 6 个)

'INTEGER' (整) 'REAL' (实)

'BOOLEAN' (布尔) 'ARRAY' (数组)

'PROCEDURE' (过程) 'SWITCH' (开关)

5. 分类符(共 3 个)

'VALUE' (值) 'LABEL' (标号) 'STRING' (行)

在以上基本符号中,有许多符号的意义是明显的,有些则不然. 关于它们的意义与作用,将在后面各章分别加以解释,这里先作一些说明:

(1) 为什么有些习惯符号不用,而要重新造一些符号,例如“大于”要写成 'GR' 而不用 “>”? 这是因为本语言的基本符号共有 86 个,而本计算机的一项重要设备(55 型电传打字机)仅有 52 个符号可供使用,因此,有些符号只好用 52 个符号的某些组合来表示;

(2) 用字母拼成的基本符号其两端必须带“'”,否则就不是基本符号. 例如, 'STEP' 与 STEP 在算法语言中表示不同的含义,前者是基本符号,它有确定的意义,表示步长;后者则是由基本符号 S, T, E, P 组合而成,只能用来给一些量或句子起名字,而不能用来代表“步长”这个意思;

(3) 只用英文字母,大小写不分,例如 A 与 a, 'INTEGER' 与 'integer' 被认为是等同的. 由于 55 型电传打字机上键盘的字母是大写的,因此,本书在写源程序时都用大写字母;

(4) 要注意区分形状相似而意义不同的符号,例如乘号 “×” 与字母 “X”, 数字 “0” 与字母 “O”, ...

§ 2 数

数的表示形式与通常的十进制数的书写形式差别不大,例如

算法语言中的数

十进制的数

51

51

-1665

-1665

+3.1416

+3.1416

1105836

12.0	12
2_{10}^4	2×10^4
$12.1_{10}-3$	12.1×10^{-3}
$-145_{10}-6$	-145×10^{-6}

数按其书写形式分成两类:

(1) 'INTEGER' 型(整型) 凡是只出现“+”, “-”号 (“+”号可省略)和数字 0—9 的数是整型数。在上面列举的数中只有前二个是整型数;

(2) 'REAL' 型(实型) 非整型数(即有小数点或指数部份的数)叫实型数。在上面列举的数中后五个都是实型数。

由于计算机字长有限,数的取值范围也受到相应的限制。在 DJS-6 机中,实型数用浮点表示,占两个单元(48 位),其绝对值在 $8.7 \times 10^{-78} \sim 2.8 \times 10^{78}$ 之间。绝对值小于 8.7×10^{-78} 的数, DJS-6 机认为是 0;绝对值大于 2.8×10^{78} 的数, DJS-6 机容纳不下,产生溢出。整型数也占两个单元,但起作用的只是其中的一个单元(24 位),故整型数的绝对值不能超过 8388607 , 即 $2^{23}-1$ 。

可见,实型数的取值范围比整型数大,这是它的一个优点。但是,整型数在机内的表示是不带误差的,而实型数则往往带有一些误差,只能达到一定的精度。由于本计算机字长比较长,所以这个误差是很小的,在一般工程计算中可以忽略,但是在某些情况下它也会带来一些不便,所以,整型数与实型数各有利弊。在需要用完全精确表示的数时(例如,控制循环次数,记录矩阵中的元素行列位置等),整型数是方便的。

对于数,我们再说明几点:

1. 数的类型是按书写形式区分的,而不管它所表示的内容是整数还是实数。例如,10000 是整型数,而 10^4 也代表 10000,却是实型数。又例如 12 是整型数,12.0 也代表 12,

但它却是实型数；

2. 与十进制表示的数一样，正数之前的“+”号可写可不写，无意义的“0”也是可写可不写。例如 +0.5, 0.5, 0.50, .5 是相等的；

3. 小拾“ $_{10}$ ”是一个基本符号，它本身并不表示一个数，即不等于 10，它的后面必须是且只能是整型数。例如， 1.5_{10} 与 $1.5_{10}3.5$ 的写法都是不允许的。如果要用 $1.5 \times 10^{3.5}$ 这个值，可利用第二章 § 1 将要介绍的指数记号“ $\uparrow$ ”，写成 $1.5 \times 10 \uparrow 3.5$ 。

§ 3 标识符、简单变量与类型说明

在绪言中已经看到，计算圆柱体体积和侧面积的源程序如下：

```
'BEGIN'
  'REAL' R, H, V, S;
  READR (R, H);
  V := 3.141592654 × R × R × H;
  S := 2 × 3.141592654 × R × H;
  OUTPUTR (V, S)
'END'
```

在这份程序中，含有 2, 3.141592654, R, H, V, S 等六个量，其中前两个量在计算过程中永远取定值，后四个量则可以改变其取值。一般地，我们把前者称为常量，后者称为变量。关于常量，我们在上一节已介绍过，至于变量，其分类如下：

变 量	{	简单变量	{	整 型简单变量	} 算术简单变量
				实 型简单变量	
				布尔型简单变量	
				下标变量	
{	下标变量	{	整 型下标变量	} 算术下标变量	
			实 型下标变量		
			布尔型下标变量		

本节只讨论算术简单变量,其它内容将在第三章§1与第四章§1中进行讨论。

3.1 标识符

简单变量用标识符来命名。标识符是以字母开头的字母数字串,例如

X, Y1, EPS, ROOT2, AIJ

等都是标识符。要特别注意,标识符必须以字母开头,且只能由字母与数字组成,因此

5AB 3Q5 (不是字母开头)

A2.6 B[2] A+10 R π α (不只是字母和数字)

都不是标识符。

标识符的长度没有限制,但在本语言中仅前八个有区分意义,例如

ABC72EFGA 与 *ABC72EFGBC*

的前八个符号相同,虽然从第九个符号开始不同,长度也不一样,但它们被认为是同一个标识符。其实,标识符太长也没有必要。

通常,我们选用符合习惯且有助于理解的标识符来代表所要计算的变量,例如

T (时间) *SUM* (和) *ROOT2* (第二个根)

H (高) *ARG* (幅角) *BETA* (希腊字母 β)

在本语言中,有一些标识符专门用于表示标准函数、输入输出过程、控制台变量、已知地址量,例如

SIN COS LN MAX MIN

INPUT READR OUTPUTR PRINTS III AAA

等(见本章§4,第七章和第十章§6),这些标识符不能再用来标识变量。