

高等学校试用教材

工 程 数 学

算法语言·计算方法

华中工学院 数学教研室 编
软件教研室

人 民 教 育 出 版 社

工 程 数 学
算法语言·计算方法

华中工学院 数学教研室 编
软件教研室

*

人民教育出版社出版
北京发行所发行
重庆新华印刷厂印装

*

1978年8月第1版 1979年3月第1次印刷
印数000,001—170,000

书号 13012·0206 定价 0.60 元

引 言

电子计算机的出现有力地推动了科学技术的发展。今天，掌握电子计算机进行科学计算，已成为广大工程技术人员的迫切要求。这本书正是为了适应这一需要编写的。

本书分两篇，这两篇是有机地联系在一起。第一篇“算法语言”着重体现算法语言是算法的描述工具，为学习第二篇“计算方法”作准备；第二篇介绍电子计算机上的常用算法时，则尽量落实到框图或程序。

本教材教学时数，“算法语言”可为16—18学时，“计算方法”可为22—24学时。由于学时较少，本书只是介绍一些必要的基本概念和基本方法，作为进一步学习这方面内容的入门。使用本书时，可按实际需要内容加以充实或取舍。

算法语言和计算方法都是实践性较强的学科，建议在教学过程中加强上机实习，结合专业需要解决实际课题。

本书第一篇“算法语言”是由华中工学院林化夷编写的，由北京工业大学曹德和同志主审，参加这一部分审稿工作的单位尚有北京邮电学院、上海科技大学、上海交通大学等。

第二篇“计算方法”是由华中工学院王能超编写的，由上海交通大学孙增光教授主审，参加这一部分审稿工作的单位尚有清华大学、西安交通大学、上海交通大学以及南京航空学院等。

参加审稿的同志认真负责，对原稿提出了不少宝贵意见，并协助进行了修改。另外，在编写过程中还得到其他许多同志的指导

和帮助。对此，编者一并表示深切的谢意。

由于编写时间较短，加之编者水平有限，因此，本书难免存在不少缺点和错误，恳切希望读者给予批评指正。

编 者

一九七八年六月

目 录

第一篇 算法语言 ALGOL

引言	1
第一章 算法语言的基本符号与源程序的结构	5
§1 基本符号与标识符	5
§2 源程序的结构	7
第二章 算式	11
§1 数、变量和标准函数	11
§2 赋值语句、简单算术表达式	13
§3 输入与输出过程语句	16
第三章 分支	21
§1 关系式与条件语句	22
§2 转向语句与复合语句	36
§3 开关说明与开关命名符	42
第四章 循环	51
§1 循环语句	51
§2 下标变量与数组说明	65
§3 分程序	76
第五章 过程	89
§1 一般过程	89
§2 函数过程	99
第一篇参考书和文献	108

第二篇 计算方法

第一章 算法与误差	109
§1 算法	109
§2 误差	117

第二章	方程求根	122
§ 1	引言.....	122
§ 2	二分法.....	124
§ 3	迭代法.....	128
§ 4	迭代过程的加速.....	133
§ 5	牛顿法.....	136
§ 6	弦截法.....	140
第三章	函数插值	144
§ 1	问题的提法.....	144
§ 2	线性插值.....	146
§ 3	拉格朗日插值公式.....	148
§ 4	插值余项.....	153
§ 5	逐步插值法.....	157
§ 6	分段插值法.....	160
§ 7	数值微分.....	165
第四章	数值积分	169
§ 1	插值求积公式.....	169
§ 2	变步长的梯形法则.....	175
§ 3	求积公式的误差.....	178
§ 4	龙贝方法.....	184
第五章	常微分方程的数值解法	189
§ 1	引言.....	189
§ 2	改进的欧拉方法.....	190
§ 3	龙格-库塔方法.....	195
§ 4	步长的自动选择.....	200
§ 5	一阶方程组.....	202
第六章	线性方程组的解法	208
§ 1	迭代法.....	208
§ 2	消去法.....	216
§ 3	矩阵分解方法.....	231
附录一	样条插值	238
附录二	曲线拟合	246

第一篇 算法语言 ALGOL

引 言

算法语言 ALGOL 是国内广泛流行的一种程序设计语言。为了照顾内容的通用性,故本篇不涉及特定类型的电子计算机;为了体现算法语言是算法的一种描述工具,本篇在内容安排上采取了从算法到语言的叙述,即将算法划分为算式、分支、循环和过程几种结构形式,然后按描述算法的需要有选择地引进语法成分;为使从算法到语言的过渡更为自然,在这部分内容中充分利用了框图这种描述方式作为桥梁。ALGOL 语言是人们与计算机对话的工具,为了能用 ALGOL 语言把计算步骤交给计算机,首先对电子计算机作一概略的介绍。

一、从电子计算机谈起

电子计算机是一种能执行逻辑运算和算术运算的机器。按电子计算机的用途来看,可将电子计算机分为两类:一类是“通用电子计算机”;另一类是“专用电子计算机”。“通用电子计算机”一般用作科学计算、方案设计,它的特点是解题能力强,运算速度快,数值范围大,并有记忆和逻辑判断能力。“专用电子计算机”是为解决某一方面的计算任务或自动控制而特制的,它的特点是结构简单、体积小、稳定可靠、环境适应性强。

电子计算机的结构大体上可分为以下五个主要部分:

- (1) 输入设备——计算机靠它来接受人们给它的运算任务。
- (2) 存贮器——计算机靠它来记忆。

(3) 运算器——计算机靠它来实现各种运算。

(4) 控制器——计算机靠它来指挥机器各部分自动协调工作。

(5) 输出设备——计算机靠它来把运算结果告诉人们。

然而,这五个部分究竟是怎样相互联系来完成计算任务的呢?

下面是各部分之间的关系图:

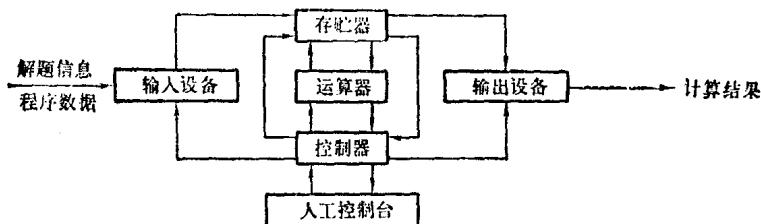


图 0-1

二、利用电子计算机解题的一般步骤

利用电子计算机解算生产实际中的问题,大体上有以下几个步骤:

(1) 构造数学模型——就是将实际问题归纳为明确的数学问题。一个生产过程、一条基本运动规律、一种技术设计,总可以经过分析、试验之后,用一系列数学算式来描述,构成数学模型。

(2) 选择计算方法——对数学问题,选择运算简单、工作量节省、并能保证精确度要求的计算方法,确定计算步骤。

(3) 计算过程的程序设计——计算方法中的公式,原始数据和计算步骤,构成计算问题的完整计算过程。为了让计算机来实现这个过程,还得把它们编成程序(即以计算机能识别的语言,将计算过程表示出来)。

(4) 将计算程序和原始数据输入,上机计算,最后计算机输出计算结果。

应当指出的是:计算机的工作特点之一是每当新的信息送入

内存单元时，该单元的原有信息即被消除，否则原有信息仍然保留。

三、什么是算法语言

算法语言是算法的一种描述工具。在电子计算机产生初期，人们用电子计算机解题，需将解题步骤用机器语言编成程序。这样编出的程序叫做手编程序。

随着电子计算机的高速化，以及解题规模的扩大，一方面手编程序需要耗费大量人力与时间。另一方面手编程序极易出错，不易检查，检查错误所费的时间往往比机器解题所需时间多数百倍，乃至数千倍，为了解决这个问题，设计了种种便于编写程序的程序语言，算法语言就是其中一种，算法语言是介于机器语言和数学语言之间的一种通用语言。例如：

编写计算 $x=4$ 时， $y=x^4-1$ 的值的程序。

用算法语言来写就是下列形式：

```
begin
  integer x;
  real y;
  x:=4;
  y:=x4-1;
  writel(y)①
end
```

这份程序用普通的语言来表示，其意义如下：

开始

整型数 x ;

① 按照“中华人民共和国国家标准程序设计语言 ALGOL”的规定，在输出语句符号 write 与输入语句符号 read 后，加上表示输出(入)量个数的数字。例如这里只输出一个量 y ，故输出语句写成 writel(y)。

实型数 y ;

将 4 赋给变量 x ;

计算 x^4-1 的值并把结果送到 y 中;

打印计算结果 y

结束

由于算法语言接近数学语言,因此,用这种语言编写程序大大减少了编写程序的时间,而且也便于人们阅读、检查和修改。用算法语言编写的程序叫做源程序。

目前,程序设计语言有数十种之多,最常用的是 ALGOL₆₀ (ALGORITHMIC LANGUAGE-60 的缩写)和 FORTRAN (FORMULA TRANSLATION 的缩写)。我国自行设计和制造的各种电子计算机所使用的程序设计语言与 ALGOL₆₀ 基本相同。因此,本书中仅介绍算法语言 ALGOL。

第一章 算法语言的基本符号 与源程序的结构

任何语言都有特定的一套基本符号,例如英语是由 26 个字母和标点符号构成的。算法语言 ALGOL 也是一些基本符号构成的。下面先介绍 ALGOL 的基本符号。

§ 1 基本符号与标识符

一、基本符号

现将本书涉及到的 ALGOL 基本符号,分为四大类列出:

1. 字母^①:

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:

0, 1, 2, 3, 4, 5, 6, 7, 8, 9。

3. 逻辑值:

true(真), false(假)。

4. 定义符:

(1) 算术运算符:

+ (加), - (减), × (乘), / (除), ↑ (乘幂);

(2) 关系运算符:

< (小于), ≤ (小于等于), = (等于), ≥ (大于等于),

> (大于), ≠ (不等于);

① 目前,我国许多计算机上,字母都是大小写不分,本书也遵循这个习惯。

(3) 逻辑运算符:

\vee (或), \wedge (与), \neg (非);

(4) 顺序运算符:

goto(转向), **if**(如果); **then**(则), **else**(否则), **for**
(对于), **do**(执行);

(5) 分隔符:

·(小数点), ,(逗号), ;(分号), :(冒号), 10(小拾),
:= (赋值号), □(空档), **step**(步长); **until**(直到),
while(当);

(6) 括号:

() (圆括号), [] (方括号), **begin**(开始),
end(结束);

(7) 说明符:

integer(整型), **real**(实型), **array**(数组), **switch**
(开关), **procedure**(过程);

(8) 分类符:

value(值)。

二、标识符

什么叫做标识符呢? 以字母开头的由字母和数字组成的符号序列叫做标识符。例如:

a, b2c, eps, root, A176, sum, alpha, beta, ...

等等, 都是标识符, 因为它们都是字母开头, 且仅仅是由字母与数字组成的序列。但是, 下列符号不能作为标识符: -

8, 54B, 12xy (因为不是以字母开头);

β , πr , 4₁₀37, K-point (不只是字母和数字)。

标识符只起“标识”的作用。在写 ALGOL 程序时, 标识符可由程序人员自由选用, 但通常在确定标识符时, 应尽量采用易于识别和

易于记忆的符号,例如:

t (时间), h (高), r (半径), s (边),
 sum (和), $term$ (项), $alpha$ (希腊字母 α)。

在 ALGOL 中,有一些特殊意义的黑体字,如 **true**, **false**, **goto**, **if**, **then**, **else**, **for**, **do**, **step**, **until**, **while**, **begin**, **end**, **integer**, **real**, **array**, **switch**, **procedure**, **value** 等,不能与标识符混同。一般说来,不能用同一标识符表示两个不同的对象(当然也有例外情况)。选用标识符不得与标准函数标识符相同。下面是标准函数的标识符: **abs**, **sign**, **sqrt**, **arctan**, **ln**, **exp**, **sin**, **cos** 及 **entier**。

今后还可看到:标识符不但代表一些量的名字,而且还可以作为程序中其他对象的名字。这样说来,标识符本身没有独立的意义。

§ 2 源程序的结构

什么是计算问题的源程序呢? 我们从一个简单的例子入手介绍它的基本内容和结构。

例 已知 a, b, c, x 的值,计算 $y = ax^2 + bx + c$ 的值。

在 ALGOL 中,首先应该说明 a, b, c, x, y 都是表示实数范围内取值的变量,写成

```
real a, b, c, x, y;
```

然后,将变量 a, b, c, x 的值输入到机器内存贮器里代表变量 a, b, c, x 的单元中去,写成:

```
read4(a, b, c, x);
```

变量 a, b, c, x 的给定值输入机器后,接着就按计算公式进行运算,写成:

```
y := (a * x + b) * x + c;
```

最后,将计算结果 y 的值打印在输出纸上,写成:

`writel(y);`

以上就是采用算法语言对这个计算问题进行的全部描述,如果把它们写在一起并添加某些符号,就构成了计算问题的源程序(ALGOL 程序)。

```
begin.....开始部分
  real a, b, c, x, y; .....说明部分
  read4(a, b, c, x);
  y := (a * x + b) * x + c; } .....语句部分
  writel(y)
end.....结束部分
```

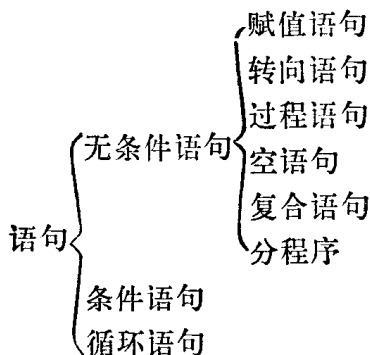
上面的源程序,用普通的语言来表示,其意义如下:

```
开始.....开始部分
  实型数 a, b, c, x, y; .....说明部分
  (从光电输入机上通过数据纸带)读入 a, b, c, x;
  计算  $ax^2 + bx + c$  的值并把结果送到 y 中; } 语句部分
  (在打印机上)输出计算结果 y
结束.....结束部分
```

以上的简单例子表明:源程序是由开始部分,说明部分,语句部分以及结束部分所构成的。开始部分和结束部分是告诉机器从什么地方开始到什么地方结束,可以看成一对括号。说明部分起介绍作用。在语句部分出现的量,一般都要在说明部分加以说明,这样就可以使机器了解所用的量具有什么性质(类型、种类),从而对不同性质的量作出不同的处理,并分配相应的存贮单元。在ALGOL中,说明部分编写在语句部分的前面,不允许二者交叉地编写。

语句部分是程序的主体,正如一篇文章,除了标题及附注之

外，是由表达各种意思的语句组成的一样。在 ALGOL 程序中，完成计算过程的是具有各种功能的语句。语句大体上分为以下几类：



这些语句的形式及其作用，将在后面各节随着描述算法的需要陆续介绍。

习 题 一

1. 下列符号中，哪些是标识符？哪些不是标识符？

x	<i>cat</i>	$x+3$	<i>next</i>	$42y$
x_{12}	$3a$	<i>begin</i>	<i>while</i>	B_1
$t(3)$	$p[2]$	158	$T_{1.4}$	<i>abc</i>
g^{-1}	ωt	A/M	<i>abs</i>	(<i>last</i>)
<i>algol₆₀</i>	<i>Fint</i>	<i>Bessal</i>	<i>function</i>	

2. 将下列程序用普通语言表示，并指出它们是描述什么样的计算公式：

(1) **begin**

```

real  $R, H, V;$ 
read2( $R, H$ );
 $V := 3.141592654 \times R \times R \times H;$ 
writel( $V$ )

```

end

(2) **begin**

```

real  $x, y, u;$ 

```



```
x = 5/13;  
y = 12/13;  
u = 0.6 × x - 0.5 × y;  
y = 0.6 × x + 0.6 × y;  
x = u;  
write2(x, y)  
end
```