

高等教育电教教材

《程序设计语言 FORTRAN77》 辅导材料

黄庙由 编

高等教育出版社

312

14/1

TP312
F11Y.1

高等教育电教教材

《程序设计语言FORTRAN77》

辅 导 材 料

黄 庙 由 编

高等教育出版社

内 容 提 要

本书是为《程序设计语言 FORTRAN77》(邓自立编著)一书编写的辅导材料。全书共分十二章,每章都和《程序设计语言 FORTRAN77》一书各章相对应。在每章中扼要阐述了该章的学习要点、难点和疑点,并给出了教材中的全部练习题,每一道题都有参考答案,所有程序都经计算机运行验证,附有运行结果。

本书可作为学习 FORTRAN77 语言的高等学校师生以及其他有关技术人员自学的辅导材料。

JS366/25

高等教育电教教材

《程序设计语言 FORTRAN77》

辅 导 材 料

黄庙由编

*

高等教育出版社出版

新华书店北京发行所发行

北京印刷一厂印装

*

开本787×10921/16 印张9 字数206,000

1986年7月第1版 1986年7月第1次印刷

印数00,001—30,200

书号13010·01284 定价1.30元

前 言

本书是为《程序设计语言 FORTRAN77》(邓自立编著)一书编写的辅导材料。全书共分十二章,每章都和《程序设计语言 FORTRAN77》一书中各章相对应,在每章中扼要阐述了该章的学习要点、难点和疑点,并对教材中的一些例题作了补充分析说明,教材中的练习题都在相应各章中全部列出,每道题都有参考答案,所有程序都经计算机运行验证,附有运行结果。

需要说明的是,本书中的习题解答,大多数只提供了一种解法。事实上,许多题目可以有多种不同的解法。而且本书的习题解答是按照循序渐进的规律,适应读者运用学习至该章为止的内容,以及利于复习和加深理解刚学习过的内容而编写的,因此不一定每题的解答都是最佳解法。

《程序设计语言 FORTRAN77》的编著者邓自立同志在百忙中主审了全书,对该书稿提出了宝贵意见;本书的所有程序系在华南工学院计算中心运行验证,特此表示衷心感谢。

由于编者水平有限,加上时间仓促,错误难免,敬请读者批评指正。

编者

1985年11月

目 录

第一章	FORTRAN77 程序的基本结构	1
第二章	数据类型、常数、变量	5
第三章	算术赋值语句、算术表达式	10
第四章	输入/输出初步	22
第五章	判定结构与控制转移语句	30
第六章	循环结构与循环控制语句	35
第七章	数组与数组元素	74
第八章	FORTRAN77 过程	88
第九章	程序单位间的数据通讯	101
第十章	字符数据处理	106
第十一章	输入/输出的格式描述	116
第十二章	输入/输出语句与文件的使用	130

第一章 FORTRAN77 程序的基本结构

一、学习提要

在这一章的文字教材中,以两个简单的 FORTRAN77 程序引例,让大家看到了 FORTRAN77 程序的概貌。就象初到一个城市,让你从最高处俯瞰整个城市一样,虽然不可能了解每部分的细节,但却可以知道它的轮廓和涉及的方面。使我们在详细学习后面各章内容时,能够产生一定的联想,有助于理解所学部分的内容、它在程序中的作用和与其他内容的关系。

这两个程序引例虽然很简单,但是出现了程序设计语言 FORTRAN77 若干个重要特点中的两个特点:表控格式输入/输出和 IF—THEN—ELSE 结构。

读者明了我们的意图后,就不必为一时不能透彻理解本章的全部内容而感到紧张了。

学习了这一章应该掌握的内容和术语是:关键字、符号名、程序的书写格式、语句与语句标号和程序结构的基本术语。

二、教材中例题的说明

由于电视教学受荧光屏幅度的限制,在编制这章内容的电视教学片时,对文字教材中引例 1 和引例 2 的 FORTRAN77 程序稍作了改动。改动过的程序及运行结果如下:

引例 1 程序:

```
C  A FORTRAN77 PROGRAM
PROGRAM EXAM1
READ*,C,W
PRINT*, 'COSTb:',C
PRINT*, 'WEIGH:',W
IF (W.GT. 20.0) THEN
    BILL=0.0032*C*(W-20.0)
ELSE
    BILL=0
END IF
WRITE(*,4) BILL
```

4 FORMAT(1X,'FREIGHT:',

 + F7.2)

 END

输入数据: 100.0, 35.5

输出结果:

	1										2										3									
	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0
COST:											1	0	0	.	0	0	0	0	0	0	0									
WEIGH:												3	5	.	5	0	0	0	0	0	0									
FREIGHT:												4	.		9	6														

引例 2 程序:

PROGRAM EXAM2

15 READ*,C,W,N1,N2

IF (C.LT.O.) STOP

PRINT*, 'COST:',C,'bbbWEIGH:',W

PRINT*, 'NO1:',N1,'bbbbbb','NO2:',N2

BILL=3.5*N1+7.*N2+B(C,W)

WRITE (*,4) BILL

4 FORMAT (1X,'FREIGHT:',F7.2)

GO TO 15

END

C THIS IS A FUNCTION SUBPROGRAM

FUNCTION B (DC,DW)

IF (DW.GT.20.0) THEN

 B=-0.0032*DC*(DW-20.0)

ELSE

 B=0

END IF

END

输入数据:100.0, 35.5, 2, 1

输出结果:

1										2										3										4										5									
1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0										
COST:										1	0	0	.	0	0	0	0	0	0	WEIGH:										3	5	.	5	0	0	0	0	0											
NO 1:										2										NO 2:										1																			
FREIGHT:										1	8	.	9	6																																			

三、习题与解答

1. 解释下列术语:

- (1) 程序
- (2) FORTRAN77 程序
- (3) 标准 FORTRAN77 字符集
- (4) 符号名
- (5) 语句关键字
- (6) 语句标号
- (7) 可执行语句
- (8) 非执行语句

解: (1) 计算机处理数据,必须把处理的步骤以指令的形式提供给计算机,这样的一串指令序列就叫做程序。

(2) 用 FORTRAN77 语言编写的程序称为 FORTRAN77 程序,又称为 FORTRAN77 源程序。

(3) 标准 FORTRAN77 字符集由 26 个英文大写字母, 10 个数字和 13 个专用字符(或称特殊字符)组成。(注:我国国家标准 GB3057-82《程序设计语言 FORTRAN》规定,货币符号“\$”改用“¥”)

(4) 符号名由 1 至 6 个字母、数字字符组成,它的第一个字符必须是字母。

(5) 语句关键字,简称关键字,是由一串字母组成的特殊字符序列,用来标识 FORTRAN77 的语句。除赋值语句和语句函数语句外,每个语句都是以关键字开始的。FORTRAN77 的关键字是采用有一定意义的英文单词,但它不能由意义相同的英文单词来代替。

(6) 语句标号提供引用语句的方法。语句标号由 1 至 5 位数字的序列组成,它可以是 1 至 99999 之间的任意整数。语句标号可以放在语句始行的第 1 列至第 5 列的任何位置。

(7) 可执行语句是指明动作的语句。对此类语句编译时需产生目标指令。

(8) 非执行语句是给编译程序提供附加信息的语句,如描述数据的特性、排列和编辑信息,说明语句功能及程序单位的类别等。

2. 源程序纸上每行总共多少列?什么叫程序行?它又分多少区?各区用途是什么?标志是什么?在一行里语句内容写在什么位置?一行写不完怎么办?

解: 源程序纸上每行总共 80 列(即有 80 格)。每行从第 1 列至第 72 列组成的字符序列称为程序行。它又分三个区,标号区、续行区和语句区。标号区,指第 1 至第 5 列,用来书写语句标号;续行区,指第 6 列,用来书写续行标志,续行标志可以是除空白字符或数字 0 以外的 FORTRAN77 标准字符集的任意字符;语句区,指第 7 至第 72 列,语句内容就写在这个区中。如果语句

内容较长,一行写不完,可以利用续行,一个语句最多可以有 19 个续行。

3. 源程序纸上一行可以书写两个或两个以上的 FORTRAN77 语句吗?

解: 不行。

4. 什么叫程序单位?有几种程序单位?程序单位中没有 END 语句可以吗?有两个或多个 END 语句可以吗?

解: 程序单位由语句和任选注解行的序列组成。程序单位分为主程序和辅程序两种。程序单位必须以 END 语句结束。END 语句指出程序单位的结束,只允许在程序单位的最后一行出现一个 END 语句,而不允许再在程序单位中的其他位置出现 END 语句。

5. 什么叫做可执行程序?可执行程序由哪些程序单位组成?可执行程序中有没有主程序可以吗?有两个或两个以上主程序呢?可执行程序没有辅程序可以吗?有两个或两个以上辅程序可以吗?

解: 可执行程序是若干个程序单位的集合,它必须仅由一个主程序单位和任意个(包括零个)辅程序单位组成。因此,可执行程序中不能没有主程序,也不能有两个或两个以上主程序,而没有辅程序是可以的。一个可执行程序可以有两个或两个以上辅程序。

6. 下列符号串中哪些可以作为符号名?哪些不可以?为什么?

X	1X	3Y	Y3	BGE/73	BGE73
MI*	ACCEL	TOTAL	BILL	C-3	Z001
MAX	DJS-B	N1(3)	D. D. T	NUMBER	VALUE
-P	GM(2,3)	GAMMA	IBM-370	α	DPS8
ALPHA	NO.	P1H2	ITEM	β	BETA

解: 可以作为符号名的有:

X	Y3	BGE73	ACCEL	TOTAL	BILL	Z001
MAX	NUMBER	VALUE	GAMMA	DPS8	ALPHA	P1H2
ITEM	BETA					

不可作为符号名的有:

1X 3Y 不是字母领头

BGE/73 MI* C-3 DJS-B N1(3) D. D. T GM(2,3) NO.

出现非字母非数字字符

-P 出现非字母非数字字符,且不是字母领头

IBM-370 出现非字母非数字字符,且超过六个字符

α β 出现非 FORTRAN77 标准字符集的字符

第二章 数据类型、常数、变量

一、学习提要

在程序设计语言 FORTRAN77 中的数据比平常人们所说的数据有着更广的含义。

在 FORTRAN77 中数据的表现形式有：常数、变量、数组、函数和字符子串。

数据类型有六类：整型、实型、双精度型、复型、字符型和逻辑型，而整型、实型、双精度型和复型又统称为算术型。

在这一章中介绍了数据表现形式中的常数和变量，而数组、函数和字符子串在以后的章节叙述。

这一章对常数数据和变量数据的六种类型都作了介绍，一般只要求掌握其中的整型、实型、字符型和逻辑型，重点是整型和实型，字符型和逻辑型分别在第十章和第五章有更深入的讨论。而双精度型和复型供需要的读者选学。

整型常数在计算机内部能精确表示，运算速度快，因而表示物体的个数、运算的次数、多项式的项数、资料编号等都应该采用整常数的表达形式。

一个实型常数可以有多种不同的写法，我们在设计程序和输入数据时可自由选用。比如，这三个实型常数：

-845000. -0.845E6 -84.5E4

它们所表示的数值是完全相等的。

然而，计算机在输出带 E 指数的实常数时，就统一采用一种“规格化”形式。如上面的数值若用带 E 指数的形式输出，则计算机的输出形式为：-0.845E+06。

字符常数是包括字符串和它左右两边的分界撇号的。分界撇号之间的字符串称为字符常数的值，字符串字符的个数则是字符常数的长度。在字符常数中，空白符是有效字符。在字符常数的字符串中若含有撇号字符，要用两个相邻的且其间不插入空白字符的撇号表示。每一对相邻的撇号只算作一个字符。比如，字符常数'ISN'T'，它的值是 ISN'T，长度是 5 而不是 6。

逻辑常数只有两个：.TRUE. 和 .FALSE.，书写时不能漏了两边的小圆点。

变量是除常数外最常用的数据表现形式。变量是具有名字和类型的实体。变量的命名遵循符号名的命名规则。编译程序按照变量名给变量分配存储单元，以存放数值。变量的数值是可以改变的。

变量的类型说明有三种不同的方式：I—N 规则隐含说明、IMPLICIT 语句隐含说明和类型语句显式说明。I—N 规则只适用于说明整型与实型变量。

在同一程序单位中,经类型语句显式说明了类型的变量,则 IMPLICIT 语句或 I—N 规则隐含说明对其失效;而经 IMPLICIT 语句隐含说明了类型的变量,则 I—N 规则对其失效;而未经类型语句和 IMPLICIT 语句说明的整型与实型变量,均遵循 I—N 规则隐含说明。

二、习题与解答

1. 把下列十进制数写成 FORTRAN77 整常数:

5000 789,673 -8775 4096.0 0.05×10^5
 2^{15} 43.35×10^4 16×10^3 $+100,000 \frac{1}{0.01}$
 -8790000×10^{-3}

解: 5000 789673 -8775 4096 5000
 32768 433500 16000 +100000 或 100000
 100 -8790

2. 把下列十进制数写成 FORTRAN77 实常数:

891 0.050 +981.10 -10^{-2} 6.4×10^{-6}
 100,000,000 $-9.99 \times 10^{-18} \frac{1}{100000}$ 10^{15}

解: 891. 或 891.0 0.050 或 0.05 +981.10 或 981.1
 -0.01 0.0000064 或 $6.4E-6$ 100000000. 或 $1E8$
 $-9.99E-19$ 0.00001 或 $1E-5$ $1E15$

3. 指出下列字符串中,哪些是整常数?哪些是实常数?

哪些不是 FORTRAN77 的常数?为什么?

5,000 5000 5000. $9.2E21$ 0. $-E7$
 10^{-4} 523E8 1.E4 .E3 7.12E02
 -7.12 712. $-.4E-3$.89 e 0.E2
 $-0.5E-31$ π 2.718...

解: 整常数: 5000

实常数: 5000. $9.2E21$ 0. 523E8 1.E4
 7.12E02 -7.12 712. $-.4E-3$.89
 0.E2 $-0.5E-31$

不是 FORTRAN77 的常数:

5,000 常数中不允许出现分隔逗号
 $-E7$ 缺少基本实常数
 .E3 仅“.”字符不能作为基本实常数

10^{-4} 不允许出现乘方形式

e π 不能直接用数学中的符号作为常数,应写出具体的数值

2.718... FORTRAN77 常数的数值应是确定的,不允许含有省略符号“...”

4. 指出下列字符串中哪些是正确的字符常数?它的长度是什么?哪些不是字符常数?为什么?

(1) 'DELLTA

(2) 'ENDFIND"

(3) '+-*/'

(4) 'IT"S'

(5) 'IT'S'

解: 正确的字符常数: '+-*/' 长度为 4

'IT"S' 长度为 4

不是字符常数:

'DELLTA 缺右边分界撇号

'ENDFIND" 右边分界撇号应是单一个撇号

'IT'S' 分界撇号内字符串中的撇号字符要用两个相邻的撇号来表示

5. 指出下表所列各个常数的类型,在适当的栏上打上记号“√”:

	整型	实型	字符型	逻辑型	双精度型	复型	不是 FORTRAN77 常 数
21.5							
-0.0							
123321							
1.573E++16							
-1,000,000							
-.216E-10							
3.612E+N							
.FALSE.							
.TRUE.							
-9.D-20							
(1.6, -2.5E4)							
5D4							
-1.624E-1.8							
'TRUE'							
'EVE*SPUDDING'							
'4+3=6'							
TRUE							
'BROTHEN'							
(2.5D0, -6)							
(0,0)							
'2.5D-1.6'							

解:

	整型	实型	字符型	逻辑型	双精度型	复型	不是 FORTRAN77 常 数
21.5		✓					
-0.0		✓					
123321	✓						
1.573E++16							✓
-1,000,000							✓
-.216E-10		✓					
3.612E+N							✓
.FALSE.				✓			
.TRUE.				✓			
-9.D-20					✓		
(1.6, -2.5E4)						✓	
5D4					✓		
-1.624E-1.3							✓
'TRUE'			✓				
'EVE'SPUDDING'			✓				
'4+3=6'			✓				
TRUE							✓
'BROTHEN'							✓
(2.5D0, -6)							✓
(0,0)						✓	
'2.5D-1.6'			✓				

6. 变量类型按隐式说明,指出下列字符串中哪些是整型变量?哪些是实型变量?哪些不能作为变量?为什么?

YEAR NYEAR SS Z001 IX XI
 AB(8) A/B 8EXY EXY8 C-1
 G+Q MASS BETA α -P
 SIZG ALL N(P) AVERAGE AVERAG

解: 整型变量: NYEAR IX MASS

实型变量: YEAR SS Z001 XI EXY8
 BETA SIZG ALL AVERAG

不能作为变量:

AB(8) A/B C-1 G+Q N(P) 出现非字母非数字字符,不是正确的符号名

8EXY 不是字母领头,不是正确的符号名

-P 出现非字母非数字字符,且不是字母领头,不是正确的符号名

α 出现非 FORTRAN77 标准字符集的字符,不是正确的符号名

AVERAGE 超过 6 个字符,不是正确的符号名

7. 在 FORTRAN77 程序单位中, 变量名 YEAR, COUNT, SUM, AGE, KAPPA 表示整型变量, 变量名 K, MAX, LIMIT, AVER 表示实型变量, 你在程序单位中怎样说明这些变量类型?

解: 可在程序单位中用类型语句显式说明变量类型:

INTEGER YEAR, COUNT, SUM, AGE, KAPPA

REAL K, MAX, LIMIT, AVER

变量名 KAPPA 和 AVER 也可以不在类型语句中出现, 由 I—N 规则隐含说明:

INTEGER YEAR, COUNT, SUM, AGE

REAL K, MAX, LIMIT

8. 利用隐式类型说明, 下列类型语句中可以取消哪些变量名?

REAL RVE, R1, GOM, MASS, LIMIT

INTEGER SUM, P, BATE, LEVEL, KAPPA

解: 变量名 RVE, R1, GOM, LEVEL, KAPPA 可以不在类型语句中出现, 由 I—N 规则隐含说明变量类型:

REAL MASS, LIMIT

INTEGER SUM, P, BATE

9. 在一个程序单位中, 变量名由下列类型语句说明. 试指出每一个变量是什么类型变量, 如果是字符型变量, 指出它的长度.

CHARACTER M*6, NAME*15, Q*4

LOGICAL FLAG

COMPLEX Z1, COM, CGO

DOUBLE PRECISION D1, D2, CHAR

INTEGER X, Y, MIN

REAL MAX, PI

解: 字符型变量: M, 长度为 6;

NAME, 长度为 15;

Q, 长度为 4

逻辑型变量: FLAG

复型变量: Z1, COM, CGO

双精度型变量: D1, D2, CHAR

整型变量: X, Y, MIN

实型变量: MAX, PI

第三章 算术赋值语句、算术表达式

一、学习提要

1. 关于赋值语句

赋值语句是计算机程序用得最多和最基本的语句。FORTRAN77 的赋值语句有：算术赋值语句、字符赋值语句和逻辑赋值语句三种。

这一章只介绍了算术赋值语句。逻辑赋值语句和字符赋值语句分别在第五章和第十章中介绍。

要实现数值计算主要使用算术赋值语句。算术赋值语句中的“=”号，叫赋值号而不叫做等号，表明算术赋值语句的作用是把右边表达式的运算结果赋给左边的变量（或数组元素），而不是表示“=”号的左右相等。

学习赋值语句，首先应加深对计算机实体（变量或数组元素）存储单元两个重要特点的认识：

（1）实体存储单元中存放的数值可反复使用，取之不尽，因此，实体名可在表达式中多次出现。

（2）实体存储单元中存放的数值“一冲”即丢。只要实体名在赋值号的左部出现，执行该赋值语句后，原实体存储单元中的数值就会被消除，而由新的数值所取代。

学习算术赋值语句主要应掌握算术表达式的书写和赋值规则。

算术表达式的书写与平常的数学运算式十分相似。这正是高级算法语言的特点，因此容易掌握。文字教材中已指出了要注意的地方，除此之外还有几个问题希望读者在学习时注意。

（1）在算术表达式中，+、-、*、/运算符的左右操作数虽然允许取不同类型，但是计算机在执行这种运算前，需首先把低类型的操作数转换为高类型的操作数，得耗费一些机时，因此，应尽可能使这些运算的左右操作数为同一类型。例如：

对于数学运算式： $b^2 - 4ac$

写成 FORTRAN77 表达式： $B*B - 4.*A*C$

在表达式中把 4 写成 4.，就使得表达式在进行运算时不必作类型转换了，又如：

对于物理势能计算公式： $m \cdot g \cdot h$

写成 FORTRAN77 表达式： $AM*9.81*H$

原来的公式 $m \cdot g \cdot h$ ， m 表示质量，并不含取整型数的意思，但按 I—N 隐含规则，变量 M 就是整型的了，为避免表达式在计算时作不必要的类型转换，因而可用变量 AM 来表示质量 m ，除非 M 在类型语句中说明为实型。

(2) 最好把高次多项式写成分层次提取公因子的方式。例如:

数学多项式: $9 - 5x + 7x^2 - 2x^3$

写成 FORTRAN77 表达式: $9 - X * (5 - X * (7 - 2 * X))$

这样运算速度较快。

(3) 乘方运算有其特殊性。如果指数为整型值, 则不管底数是整型还是实型均不属混合型表达式, 底数为整型(如 $K**3$)则为整型表达式; 底数为实型(如 $X**4$)则为实型表达式。指数为整型的乘方运算, 在执行中计算机系统处理为底数的自乘运算, 速度较快。

如果指数为实型值, 计算机系统处理为对数运算, 如 $X**4$, 计算机系统处理为这样的运算: $e^{4 \ln x}$, 相当于表达式: $EXP(4 * LOG(X))$ 。因而运算速度较慢。指数为实型的乘方表达式, “**”号左右不同类型操作数的转换规律与 +、-、*、/ 运算相同。

2. 关于常数符号名

常数本来是没有名字的, 它的类型和数值由它的书写形式直接表明。FORTRAN77 规定可以通过 PARAMETER 语句给常数命名。初学者对此常常不太理解, 既然可以把常数值赋给变量, 为何又要增加一个常数符号名的概念呢?

变量名和常数符号名都要符合符号名的命名规则, 而且都有类型之分。但它们有一个重要的区别, 变量的值在程序运行中是可变的, 变量在程序中可以多次赋值; 而常数符号名的值是恒定的, 常数符号名在同一程序单位中只能定义一次。具有常数基本性质的常数符号名, 在程序单位中可以出现在许多只允许常数而不允许变量名出现的地方。这里举一个例子, 不过这个例子要在学习了后面的内容后才能完全理解:

```
PARAMETER (K=14)
REAL A(K), B(K, K)
:
DO 10 I=1, K
:
```

如果这些语句是在一个主程序单位, 在有横线的符号位置, FORTRAN77 规定都不能是变量, 而必须是常数值, 因此在这些位置写上常数 14 时, 如果在不同的执行中要改变这个数值为 20, 则要对程序作多处修改。但是, 如果使用了 PARAMETER 语句, 命名了一个常数名 K, 就只要改变这个语句定义常数名 K 的常数值就行了, 这就给修改程序带来很大方便。

我们在 PARAMETER 语句中命名常数名时要注意的是一“=”号右边的表达式必须是只含常数和定义过的常数符号名的常数表达式, 而不允许出现变量、函数引用或数组元素。

3. 关于内部函数

在介绍函数的概念时, 文字教材中用到一个术语“哑元”, 在有些书中称为“虚拟变元”或“形式参数”。

FORTRAN77 有三种函数: 内部函数、语句函数和外部函数。内部函数是由计算机处理系统提供的, 语句函数和外部函数则是由程序员自己定义的函数。

标识内部函数可用属名和专用名。例如绝对值函数：

属名	专用名	变元类型	引用函数结果
	ABS	实	实
ABS	IABS	整	整
	DABS	双精度	双精度
	CABS	复	实

有一个属名、4个专用名。每个专用名对应于一种类型变元的引用，而属名简化了内部函数的引用，每种类型的实元都可使用属名引用内部函数，结果类型（执行类型转换和复变元的绝对值除外）和变元的类型相同。尽管如此，我们还是要了解各个内部函数允许的变元类型。比如，SIN函数只允许实型和复型变元。如果使用整型变元引用SIN函数则是错误的。

值得一提的是，当内部函数作实元时，只有内部函数专用名方可用作实元。例如：

MAX(100, ABS (K))

如果变元K按I—N隐含规则说明为整型，由于ABS(K)为引用MAX函数的实元，对于整型变元K，ABS是绝对值函数的属名而不是专用名，因此，以上引用MAX函数的表达式是错误的。正确的应写为：

MAX(100, IABS (K))

二、习题与解答

1. 设当前存储单元的内容如图所示：

A	0.3	B	3.1	C	5.4	D	20.2
---	-----	---	-----	---	-----	---	------

下列赋值语句：

A = 37./C

B = A + 1.6

D = B**2

执行后，变量A, B, C, D的存储单元的内容是什么？

A		B		C		D	
---	--	---	--	---	--	---	--

解：赋值语句依次执行后，变量A, B, C, D的存储单元的内容为：