



普通高等教育“十五”国家级规划教材

农业数学实验

孟 军 尹海东 任永泰
傅丽芳 王淑艳 张战国

编著



科学出版社
www.sciencep.com

普通高等教育“十五”国家级规划教材

农业数学实验

孟 军 尹海东 任永泰 编著
傅丽芳 王淑艳 张战国

顾凤岐 葛家麒 审稿

科学出版社

北京

内 容 简 介

本书为普通高等教育“十五”国家级规划教材,是面向高等农业院校的教学实验课教材。主要内容有:Matlab、SPSS 和 LINGO 3 个教学软件使用方法的介绍,数值计算方法、优化分析、统计分析方向的实验和数据挖掘方法的简介及其实验。本书另辟一章,介绍结合农业生产的实际情况的 10 个综合实验。

本书可以作为农业院校学生数学实验课的教材,也可以作为数学建模课程的辅助教材,同时可供农业科技人员参考。

图书在版编目(CIP)数据

农业数学实验/孟军等编著. —北京:科学出版社, 2005

(普通高等教育“十五”国家级规划教材)

ISBN 7-03-015542-4

I. 农… II. 孟… III. 农业科学:数学-实验-高等学校-教材
IV. S11

中国版本图书馆 CIP 数据核字(2005)第 049577 号

责任编辑:周 辉 甄文全 潘继敏/责任校对:宋玲玲

责任印制:安春生/封面设计:陈 敬

科学出版社 出版

北京东黄城根北街16号

邮政编码:100717

<http://www.sciencep.com>

丽源印刷厂 印刷

科学出版社发行 各地新华书店经销

*

2005 年 7 月第 一 版 开本:787×1092 1/16

2005 年 7 月第一次印刷 印张:12 3/4

印数:1—2 500 字数:292 000

定价:20.00 元

(如有印装质量问题,我社负责调换(新欣))

前 言

本教材为普通高等教育“十五”国家级规划教材。

随着计算机技术的迅猛发展和广泛应用,数学在各个学科中的应用也越来越广泛,并成功地解决了科研、生产中的许多问题。在农业生产和科研中数学也起到越来越重要的作用,比如遗传密码的研究、病虫害的控制、遗传育种规律的研究、粮食产量的预测和土壤配方施肥等。一些实验费用较高或在实践中不能实验的研究内容,可以通过建立数学模型,并在计算机上模拟解决其具体的实验问题,数学已经渗透到各个学科不同专业,成为各学科研究工作中不可缺少的工具。因此,面向 21 世纪农业院校所培养的人才对数学素质的要求,不仅要掌握基本数学知识和具有一定逻辑推理能力,还应具有综合运用数学方法,借助计算机解决实际问题的意识和能力。

在传统观念中,学数学只需用脑,不必动手,实际上与其他自然科学一样,数学也需要使用“观察”和“实验”来形成、发展和检验。数学实验与其他自然科学实验不同的是它所面对的不是物质材料,而是图形、数据等非物质材料,或称为思想材料。前一阶段,由于数学学科发展的日益形式化,数学教学也越来越偏重形式,强调逻辑思维能力的培养,而离“观察”和“实验”越来越远,其结果是学生对数学灵魂的领会、数学方法的掌握、数学能力的提高越来越受到限制。但是随着计算机的出现,为数学实验教学提供了强有力的工具。在数学实验的过程中,学生经过:直觉—一探试—出错—思考—猜想—验证的过程,极大地激发了学习兴趣,使学生的创造潜能得到了充分的开发,这是数学教学中培养学生的创造性思维和综合能力的有效方法。

实验,从其定义上理解有实施、验证的意思,它是在对实际问题的某种假设和猜想的基础上,借助于外围设备和具体操作方案而实施验证原有思考的过程。猜想和假设是实验内容的来源,验证是实验的目的,外围设备是实验的工具,操作方案是实验的手段。数学实验就是对数学猜想、假设进行验证。在数学的发展历程中,数学家们一直在对数学的猜想和假设进行验证,只不过他们所采用的方法是在纸上进行推演、证明,这种工作只被少数的数学专业人员所熟悉,而对于广大的非数学专业人员却无从所知,并且这样的工作只是解决数学理论问题。我们这里所提到的数学实验是针对非数学专业的学生,主要包括两方面的内容:①培养学生对实践中数学问题的解决能力,使他们能借助于计算机和相关软件,解决这些问题,这与数学建模的思想相似,但数学实验注重对数学问题的假设和求解,而数学建模注重对实际问题的解决;②通过计算机的计算实现对数学猜想和假设的验证。

根据我们对数学实验的理解,将数学实验分为两个层次来开设,第一个层次是对学生的基本技能的训练。第二个层次是培养学生运用所学的数学方法,借助计算机去解决实际问题的能力,这也在数学课上培养学生创造性思维和综合能力的一个最有效的方法。本书的主要内容是在第一层次的基础上,选择一些综合的题目,让学生应用所学的数学软件,在计算机上求解,既扩大学生的知识面,又激发了学生探索的欲望。我们所编写的教

材就是针对第二个层次而进行的。

本书在编写上具有如下特色:

(1) 具有农业院校的特点。本书无论是内容的设计,还是实验例题的选择都考虑到数学在农业上的应用。比如聚类分析、试验设计、线性规划这些方法在农业的科研实践中都经常用到,通过对这些方法的学习,不仅可以使学生掌握数学实验的基本技能和基本方法,还可以加深学生对数学应用的感性认识,为将来走上工作岗位后,应用数学方法解决实际问题打下良好的基础。

(2) 本书介绍了 Matlab、SPSS 和 LINGO 三个数学软件。这三个软件各具特点,Matlab 主要用于进行数值计算和优化计算,SPSS 主要用于统计分析问题的解决,LINGO 主要用于规划问题的计算。这样的编排开阔了学生的视野,使学生对三个软件都有所学习,掌握三个软件各自的优点,保证学生在实验的过程中能够选取最适合的计算软件,尽快完成实验的计算。

(3) 在结构上将单一实验与综合实验相结合。本书的第二、三、四章分别讲解了数值计算、优化分析和统计分析三个方面的基础知识和实验方法,使学生掌握了数学实验的基本知识和相应计算软件的使用。在此基础上,第六章编写了 10 个综合实验,加强学生运用数学方法,借助于数学软件解析综合问题能力的训练。这样使学生既掌握数学实验方面的基础知识,又培养了解决综合问题的能力。

(4) 本书在第五章介绍了数据挖掘方法。数据挖掘方法是近代才兴起的新技术,是计算机技术和数学方法相结合的产物,在当今的信息时代具有广阔的应用前景。通过第五章的学习,可以使学生了解数据挖掘的思想和基本方法,增加学生的学习兴趣,扩大学生的知识面。

本书编写的分工为:东北农业大学的孟军、尹海东负责提出全书编写的总体思想,并负责全书的统稿工作。任永泰、傅丽芳编写第一章,王淑艳编写第二章,任永泰编写第三章,傅丽芳编写第四章,孟军、张战国编写第五章,孟军编写第六章的前 5 个实验,尹海东编写第六章的后 5 个实验。东北林业大学的顾凤岐教授和东北农业大学的葛家麒教授认真审阅了全书,并提出了宝贵的意见,对此表示衷心的感谢。

本书在编写和试用的过程中得到了东北农业大学数学系全体教师的热心帮助,在此表示诚挚的谢意。由于作者水平所限,书中缺点和错误难免,恳切希望得到广大读者的批评和指正!

编者

2005 年 1 月于哈尔滨

目 录

前言

第一章 常用软件的使用	1
第一节 Matlab 软件的使用	1
第二节 SPSS 简介	12
第三节 LINGO 软件的使用	15
第二章 数值化方法	19
第一节 插值与拟合	19
第二节 线性方程组求解	29
第三节 非线性方程求根	39
第四节 常微分方程数值解	44
第三章 优化方法	58
第一节 线性规划问题的提出及其数学模型	58
第二节 非线性规划	68
第三节 整数规划	75
第四章 统计分析	85
第一节 统计图形	85
第二节 方差分析	94
第三节 回归分析	100
第四节 聚类分析	105
第五节 判别分析	113
第六节 试验设计	117
第五章 数据挖掘技术及其算法简介	132
第一节 数据挖掘的基本内容	132
第二节 关联规则	135
第三节 决策树方法	140
第四节 贝叶斯分类	144
第六章 综合实验	151
实验一 应用蒙特卡洛方法对粮食产量的模拟	151
实验二 分形与混沌简介	156
实验三 用目标规划方法解决农业生产计划安排问题	159
实验四 Lotka-Volterra 生态数学模型的求解	162
实验五 小孩与玩具	165
实验六 施肥效果分析	167
实验七 作物育种方案的预测	171

第一章 常用软件的使用

随着计算机技术的迅速发展,在近几十年许多优秀的数学软件相继诞生。到目前为止,比较优秀的数学软件已有 30 余种,应用于数学教学方面的自由软件也不下 10 余种。这些数学软件在开发之初,大都本着减轻数学工作者编程负担的动机,经过数年的开发形成了各自的风格,也奠定了各自的应用领域。纵观这些数学软件,它们有的可以解决数值和符号计算;有的可以广泛应用于科学计算、建模、仿真和数据分析处理及工程作图;有的专攻统计分析、时间序列;有的专攻线性规划。由于应用领域及篇幅的限制,在这里我们只对 Matlab、SPSS、LINGO 这三种比较常用的数学软件的使用作简单的介绍。

第一节 Matlab 软件的使用

Matlab 这个名字是由 MATrix 和 LABoratory 两词的前三个字母组合而成。20 世纪 70 年代后期,美国新墨西哥大学 Cleve Moler 教授用 Fortran 编写了萌芽状态的 Matlab。经过几年的校际流传以及后来 Matlab 以商品形式的出现,到 20 世纪 90 年代初期,Matlab 在数值计算方面就已经独占鳌头。在国际学术界,Matlab 已经被确认为准确、可靠的科学计算标准软件。在设计研究单位和工业部门,Matlab 也被认作进行高效研究、开发的首选软件工具。另外,国内各大高校也将 Matlab 作为数学实验教学的首选软件。

一、Matlab 语言简介

一种语言之所以能如此迅速地普及,显示出如此旺盛的生命力,是由于它有着不同于其他语言的特点,正如同 Fortran 和 C 等高级语言使人们摆脱了需要直接对计算机硬件资源进行操作一样,被称为第四代计算机语言的 Matlab,利用其丰富的函数资源,使编程人员从烦琐的程序代码中解放出来。Matlab 最突出的特点就是简洁。Matlab 用更直观的、符合人们思维习惯的代码,代替了 Fortran 和 C 语言的冗长的代码。给用户带来的是最直观、最简洁的程序开发环境。

(一) Matlab 语言的特色

Matlab 语言的主要特点是:语言简洁,库函数丰富;运算符丰富(事实上 Matlab 是用 C 语言编写的,所以它提供了和 C 语言一样丰富的运算符);程序语法限制不严格,自由度大;图形功能强大;强大的工具箱;源程序具有开放性。

下面仅就 Matlab 的几个特色举几个例子,让广大读者对 Matlab 有一个感性上的认识。

1. 矩阵求逆问题

例 1.1.1 $A = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 4 & 5 \\ 1 & 1 & 2 \end{bmatrix}$, 求 A^{-1} 。

Matlab 的解决过程如下:

```
>>A=[1 2 3;3 4 5;1 1 2]
```

```
A =
```

```
1    2    3
3    4    5
1    1    2
```

```
>>A^(-1)
```

```
ans =
```

```
-1.5000    0.5000    1.0000
 0.5000    0.5000   -2.0000
 0.5000   -0.5000    1.0000
```

由此可见,对于其他语言,比如 Fortran 和 C 等较难于解决的矩阵求逆问题,Matlab 的解决过程是如此简单。

2. 解方程和求特征值问题

例 1.1.2 设 $A = \begin{bmatrix} 3 & 2 & 4 \\ 2 & 0 & 2 \\ 4 & 2 & 3 \end{bmatrix}$, $B = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$, 求 $AX=B$ 的解及 A 的特征值。

Matlab 的解决过程如下:

```
>>A=[3 2 4;2 0 2;4 2 3]
```

```
A =
```

```
3    2    4
2    0    2
4    2    3
```

```
>>B=[1;2;3]
```

```
B =
```

```
1    2    3
```

```
>>X=A^(-1)*B
```

```
X =
```

```
1.5000   -0.7500   -0.5000
```

```
>>eig(A)
```

```
ans =
```

```
-1.0000   -1.0000    8.0000
```

3. 画出 $z = \frac{1}{4}x^2 + \frac{1}{3}y^2$ 的图形

Matlab 的解决过程是:

```
>>clf, x=-5:5; y=x; [x, y]=meshgrid(x, y); z=1/4*x^2+1/3*y^2;
```

```
surf(x, y, z);
hold on, colormap(hot); stem3(x, y, z, 'bo')
```

返回图形(如图 1.1.1)。

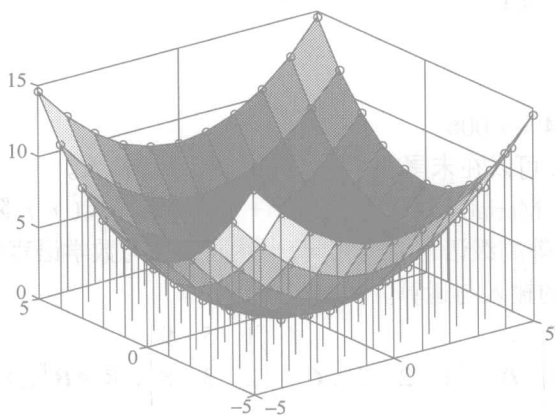


图 1.1.1 $z = \frac{1}{4}x^2 + \frac{1}{3}y^2$ 图形

4. 验证三角等式: $\sin(\theta_1 - \theta_2) = \sin\theta_1\cos\theta_2 - \cos\theta_1\sin\theta_2$

Matlab 解决过程如下:

```
>>syms cita1 cita2;
>>y=simple(sin(cita1)*cos(cita2)-cos(cita1)*sin(cita2))
y =
    -sin(-cita1+cita2)
```

表明公式正确。

(二)Matlab 的基本知识

1. 基本运算与函数

在 Matlab 窗口下进行基本数学运算,只需将数学运算式改写成计算机能够识别的表达式直接键入提示号>>之后,按回车即可。

例 1.1.3 求 $[83 - 6(99 + 7)^2] \div 7^3$ 的算术运算结果。

Matlab 解决过程如下:

```
>>(83-6*(99+7)^2)/7^3
ans =
   -196.3061
```

我们还可以将上述表达式的结果赋给一个变量 a 。

```
>>a=(83-6*(99+7)^2)/7^3
a =
   -196.3061
```

此时 Matlab 会直接显示 a 的值,并且在没有再次对 a 赋值的情况下可以调出 a 的

值,也可以使用它:

```
>>a
a=
-196.3061
>>a^3
ans=
-7.5649e+006
```

如果表达式很长,可以在末尾加...,并在下一行继续输入。

由例 1.1.3 可知,Matlab 识别所有的加(+)、减(-)、乘(*)、除(/)以及幂运算(^),并且优先级与 C 语言等高级语言完全相同,当然同样也与数学语言相一致。

例 1.1.4 矩阵的输入与运算。

设 $A = \begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$, $B = [1 \ 2 \ 3]$, $C = \begin{bmatrix} 1 & 5 & 9 \\ 2 & 3 & 8 \\ 1 & 2 & 1 \end{bmatrix}$, 求 AB^T , $A - C$, $3A$ 。

Matlab 解决过程如下:

```
>>A=[1 4 7;2 5 8;3 6 9];
>>B=[1 2 3];
>>C=[1 5 9;2 3 8;1 2 1];
```

```
>>A*B'
```

```
ans=
30 36 42
```

```
>>A-C
```

```
ans=
0 -1 -2
0 2 0
2 4 8
```

```
>>3*A
```

```
ans=
3 12 21
6 15 24
9 18 27
```

由此例可以看出 Matlab 对于矩阵的输入与运算也是非常简洁的,并且与数学习惯是很接近的。另外 $A = [1 \ 4 \ 7; 2 \ 5 \ 8; 3 \ 6 \ 9]$ 中的 “[]” 表示数组,空格是元素之间的分隔符,可以用“,”代替,“;”是行分隔符表示换行。而语句末尾的“;”用来抑制输出,加“;”则本行命令不输出结果,不加则输出结果。

例 1.1.5 矩阵元素的调用。Matlab 对矩阵元素的调用、修改、增加是非常灵活的,这一点为编程人员提供了极大的方便。

可以给 A 赋一个空矩阵,而后期可以对 A 这个矩阵自由操作。

```
>>A=[]
```

```
A=
[ ]
A(1,1)=5
A=
5
```

此时, Matlab 把矩阵 A 赋成 1×1 阶矩阵。

```
>>A(2,3)=7
A=
5 0 0
0 0 7
```

此时, Matlab 又把 A 扩张成 2×3 阶矩阵, 这一点其他高级语言做起来是很困难的。

```
>>B=[1 2 3 4;5 6 7 8;9 1 2 3]
>>B =
```

```
1 2 3 4
5 6 7 8
9 1 2 3
```

```
>>B(2,3) %调出 B 的第二行第三列的元素
```

```
ans=
7
```

```
>>B(2,:) %调出 B 的第二行的元素
```

```
ans=
5 6 7 8
```

```
C=B(:,3) %调出 B 的第三列的元素并赋给 C
```

```
C=
3
7
2
```

```
>>B(2:3,1:3) %调出 B 的第二行到第三行, 第一列到第三列元素(子矩阵)
```

```
ans=
5 6 7
9 1 2
```

```
>>B([1 3],[1 3 4]) %调出 B 的第一、三行, 第一、三、四列交叉得到的子矩阵
```

```
ans=
1 3 4
9 2 3
```

```
>>B([1 3],:)=[] %删除 B 的第一、三行
```

```
B=
5 6 7 8
```

```
>>B=[B;1 1 1 1] %将 1,1,1,1 加到 B 的第四行
```

```

B=
     5     6     7     8
     1     1     1     1
>>B(:, :)           %相当于>>B
ans=
     5     6     7     8
     1     1     1     1

```

例 1.1.6 关于向量的运算。

```

>>X=[1 2 3 4];
>>Y=[5 6 7 8];
>>min(X)             %返回 X 中元素的最小值
ans=
     1
>>mean(Y)            %返回 Y 中各元素的平均值
ans =
     6.5000
>>dot(X, Y)          %返回 X 与 Y 的内积
ans=
     70

```

关于向量运算的函数还有很多,表 1.1.1 给出了部分此类常用的函数。

表 1.1.1 常用函数(x 为向量)

函数名	函数意义
min(x)	向量 x 的元素的最小值
max(x)	向量 x 的元素的最大值
mean(x)	向量 x 的元素平均值
median(x)	向量 x 的元素的中位数
std(x)	向量 x 的元素的标准差
diff(x)	向量 x 的相邻元素的差
sort(x)	对向量 x 的元素进行排序(sorting)
length(x)	向量 x 的元素个数
norm(x)	向量 x 的欧几里得(Euclidean)长度
sum(x)	向量 x 的元素总和
prod(x)	向量 x 的元素总乘积
cumsum(x)	向量 x 的累计元素总和
cumprod(x)	向量 x 的累计元素总乘积
dot(x, y)	向量 x 和 y 的内积
cross(x, y)	向量 x 和 y 的外积

例 1.1.7 关于特殊矩阵的生成。


```

>>eye(3) %生成 3 阶单位阵
ans =
    1     0     0
    0     1     0
    0     0     1
>>eye(3,4) %生成 3×4 阶“单位阵”(此处是 Matlab 不同
ans = %于数学理论之处,姑且也将它称为单位阵)
    1     0     0     0
    0     1     0     0
    0     0     1     0
>>A=[1 2 3 4;5 6 7 8;1 0 0 0]
A =
    1     2     3     4
    5     6     7     8
    1     0     0     0
>>length(A) %返回矩阵 A 的行数与列数中的最大值
ans =
    4

```

关于此类的函数还有很多,表 1.1.2 给出用函数建立矩阵的几种常用方法。

表 1.1.2 建立矩阵的常用方法

函数名	函数意义
eye(n)	生成 n 阶单位方阵, n 为正整数
eye(m, n)	生成 m×n 阶单位阵
zeros(n)	生成 n 阶 0 方阵
zeros(m, n)	生成 m×n 阶 0 阵
rand(m, n)	生成 m×n 阶随机数矩阵
diag(A)	取 A 的对角
tril(A)	取 A 的下三角
triu(A)	取 A 的上三角

例 1.1.8 计算 $\cos(20)e^{-3}$ 的值。

```
>>y = cos(20) * exp(-3)
```

```
y =
```

```
0.0203
```

cos 是余弦函数, exp 是指数函数, 表 1.1.3 中列出了常用的数学函数。

Matlab 还提供了大量的数学函数, 在这里由于篇幅的限制不再一一列出, 读者可以在使用时查阅相关文献。

表 1.1.3 常用数学函数

函数名	函数意义
abs(x)	纯量的绝对值或向量的长度
angle(x)	复数 x 的相角
sqrt(x)	开平方
real(x)	复数 x 的实部
imag(x)	复数 x 的虚部
conj(x)	复数 x 的共轭复数
round(x)	四舍五入至最近整数
fix(x)	无论正负, 舍去小数至最近整数
floor(x)	地板函数, 即舍去正小数至最近整数
ceil(x)	天花板函数, 即加入正小数至最近整数
rat(x)	将实数 x 化为分数表示
rats(x)	将实数 x 化为多项分数展开
sign(x)	符号函数。当 $x < 0$ 时, $\text{sign}(x) = -1$; 当 $x = 0$ 时, $\text{sign}(x) = 0$; 当 $x > 0$ 时, $\text{sign}(x) = 1$
sin(x)	正弦函数
cos(x)	余弦函数
tan(x)	正切函数
asin(x)	反正弦函数
acos(x)	反余弦函数
atan(x)	反正切函数
atan2(x, y)	四象限的反正切函数
sinh(x)	超越正弦函数
cosh(x)	超越余弦函数
tanh(x)	超越正切函数
asinh(x)	反超越正弦函数
acosh(x)	反超越余弦函数
atanh(x)	反超越正切函数

例 1.1.9 Matlab 中的逻辑运算与关系表达式。

```
>>A=[0 2 1 5;1 2 7 0];
```

```
>>B=[4 0 7 9;1 1 0 2];
```

```
>>A&B
```

```
ans =
```

```
0    0    1    1
1    1    0    0
```

Matlab 并没有单独定义逻辑变量。在 Matlab 中, 数值只有“0”和非“0”的区分。非 0 往往被认为是逻辑真, 或逻辑 1。除了单独两个数值的逻辑运算外, 像刚才的例子, Matlab 还支持矩阵的逻辑运算。另外 Matlab 的大于、小于和等于关系分别由 >、< 和 = 表示。判定方法不完全等同于 C 这类只能处理单独标量的语言。Matlab 关系表达式返回的是整个矩阵, 如例 1.1.10。

例 1.1.10

```
>>B=[1 2 3 4;5 6 7 8];
```

```
>>A=[1 3 5 7;2 4 6 8];
```

```
>>A==B
```

```
ans =
```

```
1    0    0    0
```

```
0    0    0    1
```

确实使得 A 与 B 对应元素相等的位将返回 1, 否则返回 0。Matlab 还可以用 $>=$ 和 $<=$ 这样的符号来比较矩阵对应元素的大小。

另外, Matlab 还提供了 `all()` 和 `any()` 两个函数来对矩阵参数作逻辑判定。`all()` 函数在其中变元全部非 0 时返回 1, 而 `any()` 函数在变元有非零元素时返回 1。`find()` 函数将返回逻辑关系全部满足时的矩阵下标值, 这个函数在编程中相当常用。还可以使用 `isnan()` 类函数来判定矩阵中是否含有 NaN (无穷大型) 数据, 若有则返回这样参数的下标。此类函数还有 `isfinite()`、`isclass()` 和 `ishandle()` 等。

由于 Matlab 的函数非常丰富, 我们以上仅仅是介绍了一些比较常用的, 读者可以查阅一些相关文献和书籍来获取更多的函数。

2. Matlab 的语句流程与控制

作为编程语言的 Matlab, 它支持多种流程控制结构, 比如循环结构、条件转移结构、开关结构, 另外还支持一种全新的试探结构。

循环语句有两种结构: `for... end` 结构和 `while... end` 结构, 这两种循环结构各有特色, 并不完全相同。`for... end` 语句的常用格式为:

```
for 循环变量=S1:S2:S3
```

```
循环体语句组
```

```
end
```

例 1.1.11 用 `for` 循环结构给数组 X 赋予 1 到 10 这 10 个整数。

```
>>for i=1:10; X(i)=i;
```

```
end;
```

```
>>X(回车)
```

```
X=
```

```
1    2    3    4    5    6    7    8    9    10
```

在这个例子中 $S_1=1$ 、 $S_3=10$ 、 S_2 默认为 1 且可以省略。另外, 在 Matlab 中, S_2 可以为负值或非整数, 这一点与其他高级语言有所不同。另外在实际编程中采用循环语句会降低执行速度, 所以前面的程序可由下面命令代替: `i=1:10`。当然由于 10 很小, 看不出速度的差异, 若改成 1000 就看出后者执行速度更快些。

`while` 循环结构的调用格式为:

```
while(循环控制表达式), 循环体语句组 end
```

例 1.1.12 求 $\sum_{i=1}^{100} i$ 的值。

```
>>S=0; i=1; while(i<=100), S=S+i; i=i+1; end
```

```
>>S
```

```
S =
```

条件转移语句的调用格式为:

```
if      条件式 1
      条件块语句组 1
elseif 条件式 2
      条件块语句组 2
...
else   条件块语句组 n
end
```

例 1.1.13 一个简单的转移结构。

```
>> cost = 10;
>> number = 12;
>> if number > 8
sums = number * 0.95 * cost;
end;
>> sums
sums =
```

114.0000

Matlab 从 5.0 版开始提供了开关语句结构,其基本语句结构为:

```
switch 开关表达式
case 表达式 1
      语句段 1
case {表达式 2, 表达式 3, ..., 表达式 m}
      语句段 2
...
otherwise
      语句段 n
end
```

当表达式的值等于表达式 1 时,执行语句段 1,执行完语句段 1 后将转出开关体。当需要在开关表达式满足若干个表达式之一时执行某一程序段,则应该把这样的一些表达式用大括号括起来,中间以逗号分开。当前面列举的所有表达式都不满足时,则执行语句段 n。在 case 语句引导的各表达式中,不要重复使用表达式,否则列在后面的开关通路将永远也不能执行。另外执行结果和 case 语句次序是无关的。

Matlab 从 5.2 版开始提供了一种全新的试探结构,调用格式为:

```
try
      语句段 1
catch
      语句段 2
end
```