

高等教育“十三五”精品规划教材

线性代数实验导论



高等教育“十三五”（精品）规划教材

# 线性代数实验导论

XIANXINGDAISHU SHIYAN DAOLUN

谢小良 胡桔州 主编



南京大学出版社

南京大学出版社

书名：线性代数实验导论

ISBN：978-7-305-11802-9

作者：谢小良 胡桔州

出版社：南京大学出版社

定价：39.80元

# 前 言

线性代数是大学教育中一门主要的数学基础课,更是一门应用性很强的课程,这不仅是因为它广泛地应用在科学、工程、经济和管理等各个领域,而且是因为它必需借助于计算机的运用,才能真正解决工程和管理中实际计算问题。在大学数学教学中,由于学时有限,学生对线性代数一些重要的基本概念与理论难以深入理解,对有关的基本方法难以熟练掌握,更谈不上在后续的各专业学习当中使用。

国外早在 20 世纪 80 年代以前,一般在线性代数课程末尾都要求学生做一个大型作业,并在大型机上用高级语言来实现。为了减轻学生学习线性代数的负担,美国新墨西哥大学计算机科学系主任 Cleve Moler 教授用 Fortran 编写了最早的 MATLAB 程序包,后面由美国 Mathworks 公司于 1984 年真正推向市场,使之成为面向科学计算、可视化以及交互式程序设计的高科技计算环境,它将数值分析、矩阵计算、科学数据可视化以及非线性动态系统的建模和仿真等诸多强大功能集成在一个易于使用的视窗环境中,为科学研究、工程设计、信息处理、金融财务建模等必须进行有效数值计算的众多领域提供了一种全面的解决方案,并在很大程度上摆脱了传统非交互式程序设计语言(如 C、Fortran)的编辑模式,代表了当今国际科学计算软件的先进水平。随着计算机的普及,各种方便高效的计算工具不断涌现,美国国家基金会(NSF)也于 1992 年开始正式推出了 ATLAST 计划(即用软件工具增强线性代数教学),引领了国际上线性代数教学改革的新趋势。

国内西安电子科技大学的陈怀琛教授及其团队也是自 20 世纪 90 年代开始在工科大学大面积推广科学计算平台软件 MATLAB 在数学及各科课程中的使用,卓有成效。他认为利用 MATLAB 进行线性代数问题的求解有三大好处:一是对于低阶(三阶及以下)的线性代数问题, MATLAB 能很简单方便地提供图形帮助,这对于理解线性代数的理论和概念是很有利的;二是对于高阶的线性代数问题,通过调用 MATLAB 函数或自己编程,帮助学生快速而准确地进行大量数据的数值计算;三是通过补充各学科领域的线性代数应用实例,提高了学生学习兴趣,加强了学生利用计算机软件解决数学问题能力的锻炼。

国家级教学名师、北京航空航天大学李尚志教授认为:通过数学实验,感受到数学的美妙,享受到数学的快乐!

国家级教学名师、国防科技大学吴孟达教授认为:数学建模不只是一种竞赛,更应成为常态化教学模式,依托数学建模与实验,让数学教学变得生动活泼!

东南大学的张小向博士认为:在学时紧张的条件下,教师应该根据学生的接受能力,有针对性地选取一些实际应用的案例,把线性代数中抽象的数学理论和数学建模的思想结合起来,增强教学效果,实现科学性、实用性、趣味性的有机统一。

我们面临的是在计算机与网络迅猛发展中成长起来的新一代大学生,如何激发学生数学学习兴趣,通过实验做数学,并用数学,提高分析问题、解决问题的能力至关重要。我们根据教育部“充分运用信息技术提高大学教学”战略,结合我国大学数学教育教改发展实际来编写这本实践教程。

本书内容共分为三大模块:

## 2 线性代数实验导论

---

第一部分, MATLAB 简介。包括基本操作、图形处理、符号运算、程序编辑与调试等。

第二部分, 基于 MATLAB 的线性代数实验。依据现行线性代数教材体系分专题章节编排, 以利于与正常教学同步。内容包括行列式计算、矩阵计算、矩阵的初等变换与应用、向量与线性方程组、特征值与矩阵对角化、实对称矩阵与二次型等。每个专题包括基本概念、MATLAB 实验、典型例题、习题及提示等。

第三部分, 比较系统地介绍了线性代数软件方法在运筹学中的应用, 以此作为线性代数实验方法的深化与提高。前三章分别介绍了软件包 WinQSB、MATLAB 以及 LINGO 在运筹学中的应用, 第四章对经济管理中的典型案例进行了分析和建模, 并用上述软件轻捷简便地求得了问题的优化结果, 由此不但让学生深刻体会了线性代数实验方法的奇思妙用, 而且大大提高了学生应用数学解决实际问题的能力, 进一步提升了学生学习的兴趣。

本书是本科院校数学类、计算机类、工程类和经济管理类等专业学生学习线性代数相关课程的必备教材, 是公共数学开展实践教学的重要参考资料和辅导丛书。

由于编写时间仓促, 加之作者水平有限, 书中难免存在不妥之处, 希望各位专家、同行、读者不吝赐教, 以求本书能在教学实践中得到不断的完善和充实。

编者

# 目 录

## 第一部分 MATLAB 简介

### 第 1 章 MATLAB 的基本操作

1.1	MATLAB 的界面	3
1.2	变量与表达式	3
1.3	MATLAB 赋值语句和输出格式	5
1.4	Help 求助命令和联机帮助	8
1.5	MATLAB 矩阵运算	8
	习 题	10

### 第 2 章 MATLAB 中的图形处理

2.1	二维作图函数 plot	13
2.2	图形修饰	14
2.3	多窗口图形	14
2.4	快捷画图	15
	习 题	17

### 第 3 章 MATLAB 中的符号运算

3.1	符号变量及其建立	19
3.2	符号表达式的计算	19
3.3	计算精度与数据类型转换	19
3.4	符号微积分运算	20
	习 题	21

### 第 4 章 MATLAB 程序设计与调试

4.1	M 文件	23
4.2	程序结构与流程控制	24
4.3	程序调试	26
	习 题	28

## 第二部分 线性代数 MATLAB 实验

### 第 5 章 行列式计算

5.1	基本概念	33
5.2	行列式计算的 MATLAB 实验	33
5.3	典型案例	42
	习 题	43

### 第 6 章 矩阵计算

6.1	基本概念	46
6.2	矩阵计算的 MATLAB 实验	46
6.3	典型案例	55

习    题 .....	58
<b>第 7 章 矩阵的初等变换与应用</b>	
7.1 基本概念 .....	62
7.2 初等变换求解线性方程组的 MATLAB 实验 .....	62
7.3 典型案例 .....	71
习    题 .....	80
<b>第 8 章 向量与线性方程组</b>	
8.1 基本概念 .....	83
8.2 向量与线性方程组的 MATLAB 实验 .....	85
8.3 典型案例 .....	93
习    题 .....	105
<b>第 9 章 特征值与矩阵对角化</b>	
9.1 基本概念 .....	107
9.2 特征值与矩阵对角化的 MATLAB 实验 .....	107
9.3 典型案例 .....	113
习    题 .....	125
<b>第 10 章 实对称矩阵与二次型</b>	
10.1 基本概念 .....	135
10.2 实对称矩阵与二次型的 MATLAB 实验 .....	136
10.3 典型案例 .....	140
习    题 .....	146
<b>第三部分 线性代数软件方法在运筹学中的应用</b>	
<b>第 11 章 WinQSB 的应用</b>	
11.1 线性规划 .....	157
11.2 整数规划与指派问题 .....	160
11.3 目标规划 .....	161
11.4 运输问题 .....	164
11.5 最短路问题 .....	165
11.6 旅行商问题(TSP) .....	167
<b>第 12 章 MATLAB 的应用</b>	
12.1 MATLAB 求解线性规划模型 .....	170
12.2 MATLAB 求解整数规划模型 .....	172
12.3 MATLAB 求解非线性规划模型 .....	174
<b>第 13 章 LINGO 的应用</b>	
13.1 LINGO 软件基本用法介绍 .....	177
13.2 用 LINGO 软件求解整数规划问题 .....	181
13.3 用 LINGO 软件求解非线性规划问题 .....	182
13.4 用 LINGO 软件求解指派问题 .....	184
<b>第 14 章 典型案例分析</b>	
14.1 面试问题 .....	187
14.2 人员安排问题 .....	189
14.3 投资计划问题 .....	191

---

14.4	生产配料问题 .....	192
14.5	合理下料问题 .....	193
14.6	存储问题 .....	194
附 录	.....	197
参 考 文 献	.....	198

# 第一部分

## **MATLAB** 简介

# 第 1 章 MATLAB 的基本操作

## 1.1 MATLAB 的界面

### 1.1.1 MATLAB 的打开、设置

通常的 MATLAB 窗口是标准的 Windows 风格,如图 1-1 所示。

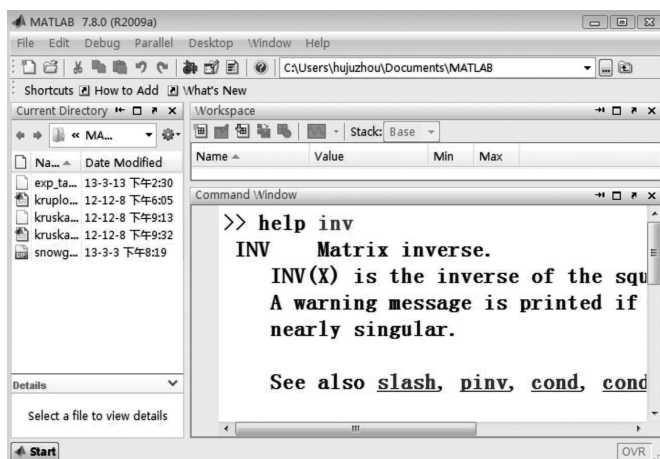


图 1-1 MATLAB 窗口界面

可以按以下方式打开或设置 MATLAB 窗口:

找到桌面 MATLAB 图标,双击即可,或者通过单击“开始”菜单 → 所有程序 → MATLAB.还可以按照自己的喜好对 MATLAB 桌面调整,也可以通过 Desktop → Desktop Layout → Default 恢复至系统默认设置。

### 1.1.2 MATLAB 的关闭

一般有三种方式:单击 File → Exit MATLAB,如图 1-2 所示;或单击右上角;或输入 quit.

## 1.2 变量与表达式

在命令提示符“>>”后输入命令,然后按下回车键,就可以进行一些简单计算了,比如下例.此外,还可以使用 Backspace 键删除修改指令及上 ↑ 下 ↓ 编辑键调出前后运行过的指令。



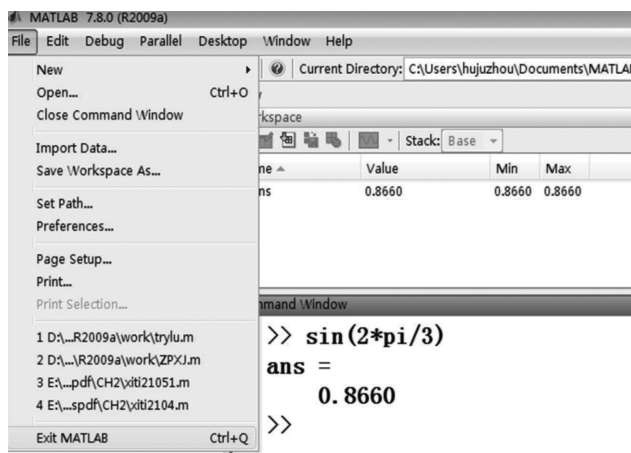


图 1-2 关闭 MATLAB 示意图

【例 1-1】 计算  $\sin \frac{2\pi}{3}$ .

`>> sin(2 * pi/3) ← % ←`表示然后按下回车键的操作,以后同在屏幕上即可显示以下结果(图 1-2):

```
ans =
    0.8660
```

这里, MATLAB 语句会将此结果存到一个预设的临时变量 `ans` 中去,且显示在屏幕上,如果你不想让结果显示出来,只需在该语句的最后加一个分号“;”即可,注意分号一定是英文状态下的.系统提供几个特有的常数变量,最好不要随意改变,列表 1-1 如下.

表 1-1 特殊变量

变量名	意义	变量名	意义
<code>ans</code>	最近答案	<code>inf</code>	无穷大
<code>eps</code>	浮点数相对精度	<code>NAN</code>	非数
<code>pi</code>	3.1415926535897……	<code>realmax</code>	机器最大浮点实数
<code>i</code>	虚数单位	<code>realmin</code>	机器最小浮点实数
<code>j</code>	虚数单位		

MATLAB 表达式的书写必须按系统提供的基本数学函数规范使用,列表 1-2 如下.

表 1-2 最常见的基本函数

函数名	功能及举例	函数名	功能及举例
<code>sin</code>	正弦 $\sin(x)$	<code>exp</code>	e 为底的指数 $\exp(x)$
<code>cos</code>	余弦 $\cos(x)$	<code>log</code>	自然对数 $\log(x)$
<code>tan</code>	正切 $\tan(x)$	<code>log2</code>	以 2 为底的对数 $\log_2(x)$
<code>cot</code>	余切 $\cot(x)$	<code>log10</code>	常用对数 $\log_{10}(x)$
<code>sec</code>	正割 $\sec(x)$	<code>rem</code>	求余数 $\text{rem}(x, 2)$
<code>csc</code>	余割 $\csc(x)$	<code>fix</code>	向零方向取整 $\text{fix}(x)$
<code>asin</code>	反正弦 $\text{asin}(x)$	<code>round</code>	四舍五入到最邻近整数 $\text{round}(x)$
<code>acos</code>	反余弦 $\text{acos}(x)$	<code>sign</code>	符号函数 $\text{sign}(x)$
<code>atan</code>	反正切 $\text{atan}(x)$	<code>abs</code>	取绝对值(模) $\text{abs}(x)$

sqrt	平方根 $\text{sqrt}(x)$	rand	$[0,1]$ 之间的均匀随机数
------	----------------------	------	------------------

**【例 1-2】** 求函数  $y = \sin x$  在点  $x = \frac{2}{3}\pi$  处的值,并将结果保存到变量  $y$  中.

```
>> x = 2 * pi/3; y = sin(x) ←
```

% 这里分号的作用是分隔语句,此外使变量  $x$  的值不显示即可得到以下结果:

```
y =  
0.866
```

变量的基本作用是接受数据输入,保存运行结果.MATLAB 规定,变量名字可以是多个字符,但第一个字符必须是英文字母,可以由字母、数字或下划线  $_$  组成.此外使用过的所有当前变量值都会存放在 Workspace(工作空间)中,可以通过输入“whos”指令查看.

**【例 1-3】** 计算表达式  $\frac{\cos(x+y)\pi - \sin 2z}{|x|+1} + e^{-\frac{1}{2}\ln 2} - \sqrt{2}$ , 其中  $x = 1 + i, y = 1 - i, z = 15^\circ$ .

```
>> x = 1 + i; y = 1 - i; z = pi/12;
```

```
>> w = (cos((x + y) * pi) - sin(2 * z))/(abs(x) + 1) + exp(-log(2)/2) - sqrt(2)
```

```
w =  
-0.5000
```

MATLAB 系统可以通过菜单对当前工作窗口进行有关操作,也可通过键盘在命令窗口输入指令方式操作,下面将若干通用指令列表 1-3 如下.

表 1-3 通用基本操作指令、专用符号

指令名	作用
whos	查看当前内存中的变量
clear	清除内存中的变量
clc	清除 MATLAB 命令窗口中所有显示内容
dir	列出指定目录下的文件和子目录
cd	改变当前工作目录
disp	输出运行后的变量和字符内容
demo	进入 MATLAB 演示系统
help	在线帮助
,	语句或数据之间的分隔
%	在命令行中用于注释
;	置于语句末尾时使结果不显示;而置于矩阵中时使数据换行

## 1.3 MATLAB 赋值语句和输出格式

### 1.3.1 赋值语句与表达式

MATLAB 中最频繁的操作是变量的赋值,其格式如下:

变量 = 表达式

这里“=”号不能理解为相等,它是赋值符号,作用是将右边表达式中的运行结果传递给

左边变量对应的内存里.

### 1.3.2 MATLAB 运算与输出格式

MATLAB 的运算符可以分为三大类:

- (1) 算术运算符;
- (2) 关系运算符;
- (3) 逻辑运算符.

#### 1. 算术运算符

各类算术运算符及其功能列表 1-4 如下.

表 1-4 算术运算符

运算符	功能	运算符	功能
+	相加	^	乘方
-	相减	.*	数组相乘
*	相乘	.^	数组乘方
\	左除	.\	数组左除
/	右除	./	数组右除

注意:

(1) 除法分为左除及右除, 主要是因为矩阵乘法不满足交换律.

(2) 带点运算符是 MATLAB 最有创新和特色之处, 其运算在矩阵(向量) 对应元素之间进行; 可以大大提高计算效率, 适应于并行计算.

**【例 1-4】** 设  $a=10, b=4, A=[4\ 5\ 6], B=[2\ 3\ 4]$ , 计算以下表达式, 观察并解释你的结果:  $a+b, a-b, a*b, a/b, a^b, A+B, A-B, A.*B, A./b, A.^2, b*A$ .

指令如下:

```
>> clear % 清除内存原有变量
>> format compact % 设置行紧缩输出格式
>> a=10; b=4;
>> A=[4 5 6]; B=[2 3 4];
>> X1=a+b; X2=a-b; X3=a*b; X4=a/b; X5=a^b;
>> Y1=A+B; Y2=A-B; Y3=A.*B; Y4=A./b; Y5=A.^2; Y6=b*A;
>> X=[X1; X2; X3; X4; X5]; Y=[Y1; Y2; Y3; Y4; Y5; Y6];
>> X, Y
```

X =

```
1.0e+004 *
0.0014
0.0006
0.004
0.0003
1.0000
```

```

Y =
    6.0000    8.0000   10.0000
    2.0000    2.0000    2.0000
    8.0000   15.0000   24.0000
    1.0000    1.2500    1.5000
   16.0000   25.0000   36.0000
   16.0000   20.0000   24.0000

```

如果增加一条语句,将改变输出数据位数:

```
>> format long           % 设置按 15 位有效数字的科学计数形式输出
```

```
>> X,Y
```

试运行以下指令,观察结果?

```
>> format rat           % 设置按有理分数形式输出
```

## 2. 关系运算符

各类关系运算符及其功能列表 1-5 如下.

表 1-5 关系运算符

运算符	功能	运算符	功能
>	判断大于关系	>=	判断大于等于关系
<	判断小于关系	<=	判断小于等于关系
==	判断等于关系	~=	判断不等于关系

【例 1-5】 设  $A = [1\ 3; 2\ 4]$ , 判断  $A$  中元素是否大于 2, 观察并解释结果.

```
>> clear
```

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

```
>> A > 2           % 对矩阵 A 中每个元素进行关系比较, 得到结果为逻辑值
```

```
ans =
```

```

    0     1
    0     1

```

## 3. 逻辑运算符

各类逻辑运算符及其功能列表 1-6 如下.

表 1-6 逻辑运算符

运算符	名称	功能说明
&	逻辑与	$1 \& 1 = 1, 1 \& 0 = 0, 0 \& 1 = 0, 0 \& 0 = 0$
	逻辑或	$1   1 = 1, 1   0 = 1, 0   1 = 1, 0   0 = 0$
~	逻辑非	$\sim 1 = 0, \sim 0 = 1$

【例 1-6】 设  $A = [1\ -2\ 3; 4\ 5\ -4; 5\ -6\ 7]$ , 请设计一段程序: (1) 直接输出小于 0 元素; (2) 查找出小于 0 的元素位置, 并解释结果.

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

```
>> A([4 6 8])           % 显示矩阵 A 的序号为 4、6、8 的元素
```

```
ans =
```

```

   -2     -6    -4

```

```
>> B = find(A < 0)           %find 是用来查找非零元素序号的逻辑函数
B =
     4
     6
     8
```

注意:矩阵元素存储是按列优先进行的.

## 1.4 Help 求助命令和联机帮助

MATLAB 的在线帮助功能非常丰富,最简单的方法就是在命令窗口中直接输入 help 指令,就会显示与该命令(函数)相关的帮助信息.例如:

```
help sin ←
SIN   Sine of argument in radians.
SIN(X) is the sine of the elements of X.
See also asin, sind.
Overloaded methods:
codistributed/sin
sym/sin
Reference page in Help browser
doc sin
```

点击 doc sin 后,还可得到详细的范例.

此外,还可通过命令窗口帮助菜单获得相关文档.建议读者经常上网访问 Mathworks 公司的中文网址获得帮助,并跟踪了解 MATLAB 的最新动态:

```
http://www.mathworks.cn
还有一些资源丰富的 MATLAB 论坛等,如:
http://www.iloveMATLAB.cn/forum.php
```

## 1.5 MATLAB 矩阵运算

MATLAB 是基于矩阵数据结构的科学计算系统,其运算包括矩阵(向量)运算及数组(Array)运算,是其最有特色之处.

### 1.5.1 矩阵(向量)的建立与运算

矩阵建立离不开标点和符号.具体来说有以下的使用方式:

- (1) 矩阵一般由放在方括号里的元素组成,元素之间使用空格隔开,如  $A = [1 \ 2]$ .
- (2) 逗号是列分隔号,如  $A = [1, 2]$  与  $A = [1 \ 2]$  的结果一样,都代表一行两列的矩阵.
- (3) 分号在矩阵中是行分隔符,如  $A = [1; 2]$  代表两行一列的矩阵分号在语句末尾时,表明该行指令的结果不显示.
- (4) 单引号' 可以使矩阵(向量)进行转置操作.
- (5) 要取出矩阵元素时,可以使用圆括号,例如取出  $A = [1 \ 2]$  中的各元素的指令是

$A(1), A(2)$ .

(6) 在使用函数时必须跟圆括号,如前面例 1-1、例 1-2 和例 1-3.

(7) 对于一次性要求计算各元素的函数值时,可以使用 MATLAB 提供的数组运算,简单快捷.如  $A = [0, \pi/4, \pi/2, 3 * \pi/4, \pi]$ , 则  $\sin(A)$  即可输出由各点的正弦值组成的向量.常见的初等函数都支持数组运算.

(8) 上面  $A$  中的元素是一个以初值为 0、公差  $0.25 * \pi$ 、终值  $\pi$  的等差数列,可以用一个带冒号的简单语句来完成:  $A = 0:0.25 * \pi:\pi$ .

**【例 1-7】** 运行下列指令,观察并解释结果:

```
>> A = [0:0.25 * pi:pi];
>> B = 0:0.25 * pi:pi;
>> C = linspace(0, pi, 5);           % 输出区间[0, pi] 中 4 等分各点
>> A, B, C
```

```
A =
    0    0.7854    1.5708    2.3562    3.1416
B =
    0    0.7854    1.5708    2.3562    3.1416
C =
    0    0.7854    1.5708    2.3562    3.1416
```

此外冒号的用法非常灵活、实用.

**【例 1-8】** 运行下列指令,观察并解释结果:

```
>> A = [1 2 3; 4 5 6];
A = 1     2     3
     4     5     6
>> A(2,:)           % 显示 A 的第 2 行各元素,冒号代表所有列
ans =
     4     5     6
>> A(2,:) - 4 * A(1,:)   % 求 A 的第 2 行元素减去第 1 行元素的 4 倍之差
ans = 0     -3     -6
>> B = [A(1), A(2), A(3), A(4)]   % 显示 A 的序号为 1 ~ 4 的元素
B = 1     4     2     5
>> C = [A(1,1), A(1,2), A(2,1), A(2,2)]
                                   % 输出 A 的第 1, 2 列的部分元素组成行向量
C = 1     4     2     5
>> D = A(:, 1:2)           % 输出 A 的第 1, 2 列的元素组成的矩阵
D =                                   % 按原来相对位置不变,第一个冒号代表取全部行
     1     2                                   % 第二个下标代表取全部 1 至 2 列
     4     5
```

从该例中可以看到矩阵是有行、列的二维表格,但在计算机中存储是一维的,且按列优先原则进行.

## 1.5.2 数组运算

数组的运算是指数组对应元素之间的运算,也称点运算.点运算符参见表 1-4.

一维数组相当于向量,二维数组相当于矩阵,因此矩阵是数组的子集.矩阵的加、减、数乘等运算与数组运算是一致的,但有两点要注意:

(1) 对于乘法、乘方和除法等三种运算,矩阵运算与数组运算的运算符及含义都不同:矩阵运算按线性代数定义,使用通常符号;数组运算按对应元素进行运算定义,使用点运算符;

(2) 数与矩阵加减及矩阵除法在数学上是没有意义的,而在 MATLAB 中为简便起见,定义了这两类运算.例如对于上例运行:

```
>> D = A + 2
```

```
D =
```

```
3     4     5
6     7     8
```

**【例 1-9】** 运行下列指令,观察并解释结果:

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

```
>> B = sin(pi * A/6)
```

```
B =
```

```
0.5     0.866     1
0.866     0.5     0
```

```
>> A .* B, A ./ B, B.^2
```

```
ans =
```

```
0.5     1.7321     3
3.4641     2.5     0
```

```
ans =
```

```
0.5     0.433     0.3333
0.2165     0.1     0
```

```
ans =
```

```
0.25     0.75     1
0.75     0.25     0
```

思考:如果运行  $A * B$  及  $A/B$  会报什么信息,运行  $B^2$  呢? 为什么?



## 习题

1. 计算表达式  $\frac{5 + \cos 47^\circ}{1 + \sqrt{7} - 2i}$  的值,并将结果赋给变量  $x$ ,然后显示出结果.

提示:  $\gg x = (5 + \cos(47 * \text{pi}/180))/(1 + \text{sqrt}(7) - 2 * i)$

2. 建立 5 阶方阵  $A$ ,判断  $A$  的元素是否能被 3 整除.

提示:  $\gg A = [24, 35, 13, 22, 63; 23, 39, 47, 80, 80;$   
 $90, 41, 80, 29, 10; 45, 57, 85, 62, 21; 37, 19, 31, 88, 76]$

```
>> P = rem(A,3) == 0 % 判断 A 的元素是否可以被 3 整除
```

3. 建立矩阵 A, 然后找出 A 中在 [10, 20] 区间的元素的位置.

(1) 自己建立矩阵 A.

```
提示: >> A = [4, 15, -45, 10, 6; 56, 0, 17, -45, 0]
```

```
A =
```

```
    4    15   -45    10     6
   56     0    17   -45     0
```

(2) 找出在区间 [10, 20] 中的元素的位置.

```
提示: >> find(A >= 10 & A <= 20)
```

```
ans =
```

```
    3
    6
    7
```

4. 在 MATLAB 中字符串是视字符为元素的向量(矩阵). 建立一个字符串向量, 然后对该向量做如下处理:

(1) 取第 1 ~ 5 个字符组成的子字符串;

(2) 将字符串倒过来重新排列;

(3) 将字符串中的小写字母变成相应的大写字母, 其余字符不变;

(4) 统计字符串中小写字母的个数.

```
提示: >> ch = 'ABc123d4e56Fg9';
```

```
>> subch = ch(1:5) % 取子字符串
```

```
subch =
```

```
    ABc12
```

```
>> revch = ch(end:-1:1) % 将字符串倒排
```

```
revch =
```

```
    9gF65e4d321cBA
```

```
>> k = find(ch >= 'a' & ch <= 'z'); % 找小写字母的位置
```

```
>> ch(k) = ch(k) - ('a' - 'A'); % 将小写字母变成相应的大写字母
```

```
>> char(ch)
```

```
ans =
```

```
    ABC123D4E56FG9
```

```
>> length(k) % 统计小写字母的个数
```

```
ans =
```

```
    4
```

5. 分别建立  $3 \times 3$ 、 $3 \times 2$  和与矩阵 A (A 为  $2 \times 3$  矩阵) 同样大小的零矩阵.

(1) 建立一个  $3 \times 3$  零矩阵.

```
提示: >> zeros(3)
```

(2) 建立一个  $3 \times 2$  零矩阵.

```
提示: >> zeros(3,2)
```

(3) 设 A 为  $2 \times 3$  矩阵, 则可以用 zeros(size(A)) 建立一个与矩阵 A 同样大小的零矩阵.



提示: `>> A = [1 2 3; 4 5 6];`

`%` 产生一个  $2 \times 3$  阶矩阵 A

`>> zeros(size(A))`

`%` 产生一个与矩阵 A 同样大小的零矩阵

6. 建立随机矩阵:

(1) 在区间  $[20, 50]$  内均匀分布的 5 阶随机矩阵.

提示: `>> x = 20 + (50 - 20) * rand(5)`

(2) 均值为 0.6、方差为 0.1 的 5 阶正态分布随机矩阵.

提示: `>> y = 0.6 + sqrt(0.1) * randn(5)`

7. 将 101 ~ 125 等 25 个数填入一个 5 行 5 列的表格中, 使其每行每列及对角线的和均为 565.

提示: `>> M = 100 + magic(5)`