矩阵运算

实验目的

1. 学会用MATLAB软件生成矩阵.
2. 熟悉MATLAB软件中关于矩阵操作的各种命令.
3. 熟悉MATLAB软件中关于矩阵运算的各种命令.

矩阵运算

- 矩阵的生成　(eye,size,ones,zeros,rand,diag,compan,vander,sym)
- 矩阵的修改　A(:,end:-1,1), diag, tril, triu, fliplr, flipud,
- 矩阵的运算　(+, -, inv, /, \, ./. .\ .*, *)

矩阵的生成

一、矩阵的生成

1、 一般矩阵的生成：

（1） 输入矩阵时要以“[ ]”为其标识，即矩阵的元素应在“[ ]”内部；

（2） 同行元素之间可由空格或“，”分隔，行与行间用“；”或回车符分隔；

（3） 矩阵元素可为运算表达式；

（4） 如不想获得中间结果，可以“ ；”结束。

矩阵的生成

例1： 输入矩阵 $A=\begin{bmatrix} 1 & 2 & 3 \\ 1 & 1 & 1 \\ 4 & 5 & 6 \end{bmatrix}$

解：

1. 直接输入法

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

2. 表达式输入法

A=[1 sqrt(4) 3;sin(pi/2) 1 1;4 5 abs(-6)]

矩阵的生成

例2： 输入矩阵

$$B=\begin{pmatrix}1+9i & 2+8i & 3+7i\\ 4+6i & 5+5i & 6+4i\\ 7+3i & 8+2i & 0+i\end{pmatrix}$$

解： 在Matlab环境中定义了两个记号 i 和 j，可以用来直接输入复数矩阵

输入命令

```
B=[1+9i 2+8i 3+7i;4+6i 5+5i 6+4i;7+3i 8+2i 0+i]
```

矩阵的生成

2、特殊矩阵的生成：

（1）生成单位矩阵

eye(n)	生成 n 阶单位阵
eye(m,n)	生成 $m\times n$ 阶单位阵
eye(size(A))	生成与矩阵A大小相同的单位阵

```
>> eye(4,7)
ans =
    1     0     0     0     0     0     0
    0     1     0     0     0     0     0
    0     0     1     0     0     0     0
    0     0     0     1     0     0     0
```

矩阵的生成

（2）生成全 1 矩阵

ones(n)	生成 n 阶全1矩阵
ones(m,n)	生成 $m \times n$ 阶全1矩阵
ones(size(A))	生成与矩阵A大小相同的全1矩阵

（3）生成全 0 矩阵

zeros(n)	生成 n 阶全0矩阵
zeros(m,n)	生成 $m \times n$ 阶全0矩阵
zeros(size(A))	生成与矩阵A大小相同的全0矩阵

矩阵的生成

例3：试生成 4 阶和 3×4 阶的单位阵，全0矩阵及全1矩阵；

解　　输入命令：eye(4)

输入命令：eye(3,4)

输入命令：zeros(4)

输入命令：zeros(3,4)

输入命令：ones(4)

输入命令：ones(3,4)

矩阵的生成

例4：已知矩阵

$$A=\begin{pmatrix}1 & 2 & 3\\4 & 5 & 6\\7 & 8 & 9\end{pmatrix}$$

试生成与矩阵A大小相同的单位阵，全0阵及全1阵；

解：输入命令

```
A=[1 2 3;4 5 6;7 8 9];    B1=eye(size(A))
```

```
B2=zeros(size(A))
```

```
B3=ones(size(A))
```

矩阵的生成

（4）生成随机矩阵

rand(n)	生成(0,1)之间均匀分布的 n 阶随机阵
rand(m,n)	生成 $m\times n$ 阶均匀分布随机数矩阵
randn(m,n)	生成 $m\times n$ 阶正态分布随机数矩阵
rand(size(A))	生成与矩阵A大小相同的随机阵

```
>> rand(2,3)
ans =
    0.9501    0.6068    0.8913
    0.2311    0.4860    0.7621
```

矩阵的生成

（5）生成对角矩阵

生成的对角矩阵的MatLab调用格式为:

$$\boxed{A= \mathrm{diag}(v,k)}$$

生成第 k 个对角线由向量 v 组成的对角阵， k 可以是正数，零或负数。当 k 是零时指主对角线(可简记为A= diag(v))， k 是负数时 v 从主对角线向左平移相应列数， k 是正数时 v 从主对角线向右平移相应列数。

矩阵的生成

例5：　（1）输入命令

```
>>A=diag([1 2 3 4])
>>A=diag([1 2 3 4],1)
>>A=diag([1 2 3 4],-1)
```

1	0	0	0
0	2	0	0
0	0	3	0
0	0	0	4

0	1	0	0	0
0	0	2	0	0
0	0	0	3	0
0	0	0	0	4
0	0	0	0	0

0	0	0	0	0
1	0	0	0	0
0	2	0	0	0
0	0	3	0	0
0	0	0	4	0

矩阵的生成

增加（合适的）行数和列数：

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

```
>>A=diag([2 3 4],-1)
```

矩阵的生成

（6）伴随矩阵

对于多项式

$$P(x) = kx^n + a_1 x^{n-1} + a_2 x^{n-2} + \cdots + a_{n-1} x + a_n$$

定义其伴随矩阵

$$A = \begin{pmatrix} -a_1/k & -a_2/k & \cdots & -a_{n-1}/k & -a_n/k \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{pmatrix}$$

矩阵的生成

生成的伴随矩阵的Matlab调用格式为:

A=compan(P)

P：多项式系数构成的行向量

例6： 考虑多项式 $P(x)=2x^4+4x^2+5x+6$

试写出该多项式的伴随矩阵

解： 输入命令

P=[2 0 4 5 6]; A=compan(P)

eig(A)； roots(P); （相等）

矩阵的生成

A =

0	-2.0000	-2.5000	-3.0000
1.0000	0	0	0
0	1.0000	0	0
0	0	1.0000	0

ans =

0.6907 + 1.5434i
0.6907 - 1.5434i
-0.6907 + 0.7565i
-0.6907 - 0.7565i

ans =

0.6907 + 1.5434i
0.6907 - 1.5434i
-0.6907 + 0.7565i
-0.6907 - 0.7565i

（7）Vandermonde 矩阵

假设有一个序列 $C=\{c_1,c_2,\cdots,c_n\}$，则可写出一个矩阵，其第(i,j)元素满足$v_{i,j}=c_i^{n-j}$， $i,j=1,2,\cdots,n$。这样可以构造一个 Vandermonde 矩阵

$$V=\begin{pmatrix} c_1^{n-1} & c_1^{n-2} & \cdots & c_1 & 1 \\ c_2^{n-1} & c_2^{n-2} & \cdots & c_2 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ c_n^{n-1} & c_n^{n-2} & \cdots & c_n & 1 \end{pmatrix}$$

矩阵的生成

生成的Vandermonde矩阵的Matlab调用格式为:

V=vander(C)

例7: 若向量 C=[1,2,3,4,5]，试写出该向量对应的 Vandermonde 矩阵

解： 输入命令

```
C=[1 2 3 4 5];    V=vander(C)
```

V =

1	1	1	1	1
16	8	4	2	1
81	27	9	3	1
256	64	16	4	1
625	125	25	5	1

矩阵的生成

3、 符号矩阵的生成：

如果已经建立起数值矩阵 A，则可以由B=sym(A) 转成符号矩阵。

比如输入向量：B=[sin(pi), cos(2), exp(6)]，则系统会认定为近似数值：B =[0.0000,-0.4161,403.4288]，无法用于精确计算。

语句将其转化成符号矩阵,即未数值化的函数形式。这样所有数值矩阵均可以通过这样的形式转换成符号矩阵，可以利用符号运算工具箱获得更高精度的解。

值得注意的是,不管原来的数值矩阵是以分数形式给出，还是以小数形式给出，转换成符号矩阵后都将以最接近于原数的有理形式给出。

矩阵的生成

```
>> b=[1,2,3;sqrt(2),sin(pi/3),0]

b =

    1.0000    2.0000    3.0000
    1.4142    0.8660         0

>> sym(b)

ans =

[         1,         2,         3]
[   sqrt(2), sqrt(3/4),         0]
```

矩阵的生成

生成特殊矩阵的函数	
eye(n)	n阶单位矩阵
zeros(n), zeros(m, n)	n阶零矩阵，零矩阵
ones(n), ones(m, n)	n阶全1矩阵，全1矩阵
[ ]	空阵
rand(n)	n阶均匀分布的随机矩阵
randn(n)	n阶正态分布的随机矩阵
compan(P)	伴随阵
diag(v, k)	生成对角阵
vander(C)	生成Vandermonde阵

二、矩阵的修改

1. 冒号表达式

冒号表达式是Matlab中很有用的表达式，在向量生成、子矩阵提取等方面都特别重要。冒号表达式的原型为

$$v = s_1 : s_2 : s_3$$

该函数生成一个行向量 v，其中 s_1 为向量的起始值，s_2 为步距，该向量从 s_1 出发，每隔步距 s_2 取一个点，直至不超过 s_3 的最大值就可以构成一个向量。若省略 s_2，则步距的默认值为 1。

矩阵的修改

例8： 试探不同的步距，从 $t \in [0,\pi]$ 区间取出一些点构成向量

解：（1）步距为：0.2

```
v1=0:0.2:pi

v1 =

     0    0.2000    0.4000    0.6000    0.8000    1.0000
  1.2000    1.4000    1.6000    1.8000    2.0000    2.2000
  2.4000    2.6000    2.8000    3.0000
```

矩阵的修改

（2）步距为：-1

v2=0:-1:pi

v2 =   Empty matrix: 1-by-0

（3）步距为：-1

v3=pi:-1:0

v3 =   3.1416   2.1416   1.1416   0.1416

（4）步距为：1

v4=0:pi

v4 =   0   1   2   3

矩阵的修改

2. 子矩阵的提取

（1）提取子矩阵在Matlab是经常需要处理的事。提取子矩阵的具体方法是：

$$\boxed{B = A(v_1, v_2)}$$

其中向量 v_1 表示子矩阵要包含的行号构成的向量， v_2 表示子矩阵要包含的列号构成的向量。若v_1为：，则表示要提取所有的行，v_2 亦有相应的处理结果。关键词end表示最后一行（或列，取决于其位置）。

矩阵的修改

例9： 设

$$A=\begin{pmatrix} 1 & 4 & 3 & 6 \\ 2 & 7 & 9 & 5 \\ 5 & 7 & 6 & 6 \end{pmatrix}$$

（1）提取矩阵A全部奇数行，所有列；

（2）提取矩阵A的3,2,1行，2,3,4列；

（3）将矩阵A左右翻转，即最后一列排在最前面；

解：分别输入命令

```
A=[1 4 3 6;2 7 9 5;5 7 6 6]; B1=A(1:2:end,:)
```

```
B2=A([3,2,1],[2,3,4])
```

```
B3=A(:,end:-1:1)
```

矩阵的修改

（2）一些特殊子矩阵的提取

diag(A,k)	表示将矩阵 A 的第 k 个对角线的元素提取组成列向量，若 k 省略 等价于k=0，表示主对角线元素。
tril(A,k)与tril(A)	表示提取矩阵A 的下三角部分，未提取部分用 0 补齐.
triu(A,k)与triu(A)	表示提取矩阵A 的上三角部分，未提取部分用 0 补齐。
rot90(A)	表示将矩阵 A 逆时针旋转90^0
fliplr(A)	表示将矩阵 A 左右翻转
flipud(A)	表示将矩阵 A 上下翻转

矩阵的修改

例10： 已知矩阵

$$A=\begin{pmatrix}1 & 2 & -3\\ 4 & -6 & 6\\ 3 & -8 & 9\end{pmatrix}$$

输入命令

```
A=[1 2 -3;4 -6 6;3 -8 9];v=diag(A)
```

```
diag(A,2)
```

```
rot90(A)
```

```
triu(A,2)
```

```
triu(A)
```

```
fliplr(A)
```

```
flipud(A)
```

矩阵的修改

```
>> A=[1 2 -3;4 -6 6;3 -8 9];v=diag(A)

v =

    1
   -6
    9

>> diag(A,2)

ans =

   -3

>> rot90(A)

ans =

   -3     6     9
    2    -6    -8
    1     4     3
```

矩阵的修改

```
>> triu(A,2)

ans =

     0     0    -3
     0     0     0
     0     0     0

>> triu(A)

ans =

     1     2    -3
     0    -6     6
     0     0     9

>> fliplr(A)

ans =

    -3     2     1
     6    -6     4
     9    -8     3

>> flipud(A)

ans =

     3    -8     9
     4    -6     6
     1     2    -3
```

矩阵的修改

3. 矩阵的扩充和部分元素的删除

（1）矩阵的扩充可用“[ ]”将小矩阵扩充成大矩阵。

* 可用命令D=[A B]构造矩阵D，其中A和B必须有相同的行。

* 可用命令 D=[A;B C] 构造矩阵D，其中B和C必须有相同的行，B和C 的列数之和必须等于A的列数。

矩阵的修改

例11：设 $A=\begin{pmatrix}1&3&6&8\\2&5&7&9\end{pmatrix}$ $B=\begin{pmatrix}1&0\\0&1\end{pmatrix}$ $C=\begin{pmatrix}1&1\\0&0\end{pmatrix}$

计算 $(B\ C)\begin{pmatrix}A\\B\ C\end{pmatrix}$

解：输入命令

```
>>A=[1 3 6 8;2 5 7 9]; B=[1 0;0 1];C=[1 1;0 0];
>>D1=[B C]
>>D2=[A;B C]
>>D1*D2
```

矩阵的修改

（2）用直接赋值的办法对超出矩阵维数的元素赋值来扩充矩阵。

* 如*A*为2 x 4 矩阵，用命令 `A(1:3,1:3)=eye(3)`

  可将矩阵*A*变成3x4矩阵，且前三行为单位阵（不管矩阵原来前三列是什么），第4列的前两行不变，第三行元素用0补齐。

  命令 `A(1:3,1:3)=ones(3)` `A(1:3,1:3)=zeros(3)` 类似

* 命令 `A(:,[i j])= A(:,[j i])` 表示交换*A*矩阵的 i,j 两列

* 命令 `A([i j],:)= A([j i],:)` 表示交换*A*矩阵的 i,j 两行

矩阵的修改

A(i,j)= k　　将矩阵第i行j列的元素变为k，其它元素的值不变。

A(:,j)=[]　　可删除矩阵*A*的第 *j* 列或第 *i* 行（减少维数）。

A(i,:)=[]　　亦可用类似的办法删除矩阵上相邻的若干行与列。

练习：生成7*7随机矩阵A，测试

A(3, 4)=2, A(9, 10)=11, A(:, 6)[], A(3, :)=[]

矩阵的修改

例12： 设 $A=\begin{pmatrix}1 & 4 & 3 & 5\\ 8 & 9 & 7 & 6\end{pmatrix}$ 输入命令

```
A=[1 4 3 5 ;8 9 7 6];A(1:3,1:3)=eye(3)
```

```
A(1:3,1:3)=ones(3)
```

```
A(1:3,1:3)=zeros(3)
```

```
A(3,5)=5
```

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

```
A(:,[1 3])=[ ]
```

```
A(:,[1 2])= A(:,[2 1])
```

矩阵的修改

例13：设 $D=\begin{pmatrix} 1 & 4 & 0 & 1 & 5 \\ 7 & 8 & 0 & 2 & 4 \\ 0 & 0 & 5 & 2 & 7 \end{pmatrix}$ 输入命令

```
D1=D([1 3],:)
```

```
D2=D([1 2],[3 5])
```

```
D3=D(2,4)
```

```
D4=diag(D,-1)
```

```
D5=diag(D,3)
```

```
D6=diag(D,-2)
```

矩阵的修改

矩阵操作函数	
diag	提取对角元素，生成对角矩阵
fliplr	左，右翻转
flipud	上，下翻转
rot90	按逆时针旋转
tril	提取矩阵的主下三角部分
triu	提取矩阵的主上三角部分

三、矩阵的运算

基本矩阵的运算符

运算符	含　义	运算符	含　义
A + B	加法	A.^n	A的各元素n次方
A - B	减法	A.^B	A, B两矩阵对应元素乘方
A * B	乘法	exp (A)	A的所有元素取以e为底的指数
A .* B	对应元素相乘	log (A)	对A的各元素取e为底的对数
A \ B	左除	sqrt (A)	对A的各元素求平方根
A / B	右除	det (A)	求A的行列式
A ./ B	A的元素被B的对应元素除	inv (A)	求A的逆矩阵
k*A	数乘	A’	求A转置
A ^ n	A为方阵时，自乘n次		

矩阵的运算

例14 设 $A=\begin{pmatrix}1&2&3\\2&3&4\\3&4&5\end{pmatrix}$ $B=\begin{pmatrix}1&1&1\\2&2&2\\3&3&3\end{pmatrix}$

求：$C=A+B$ $D=A'$ $E=A^3$ $F=A.*B$ $G=|A|$ $F=A*B$ $G=2A$

解

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

矩阵的运算

```
>> C=A+B
C =
     2   3   4
     4   5   6
     6   7   8
```

```
>> D=A'
D =
     1   2   3
     2   3   4
     3   4   5
```

```
>> E=A^3
E =
   132   192   252
   192   279   366
   252   366   480
```

```
>> F= A .*B
F =
     1    2    3
     4    6    8
     9   12   15
```

```
>> G=det(A)
G =
     0
```

```
>> G=2*A
G =
     2   4   6
     4   6   8
     6   8  10
```

```
>> F= A*B
F =
    14    14    14
    20    20    20
    26    26    26
```

矩阵的运算

例15 设 $A=\begin{pmatrix} 2 & 5 \\ 1 & 3 \end{pmatrix}$ $B=\begin{pmatrix} 4 & -6 \\ 2 & 1 \end{pmatrix}$ 求满足方程$X+AX=B$的矩阵X.

解 $X=(I+A)^{-1}B$

```
>>A=[2 5;1 3];
>>B=[4 -6;2 1];
>>I=eye(size(A));
>>X=(A+I)\B
```

```
X=
    0.8571   -4.1429
    0.2857    1.2857
```

注：X=A/B的意义为满足XB=A的矩阵X。

注：X=A\B的意义为满足AX=B的矩阵X。

矩阵的运算

还可以利用伴随矩阵求逆矩阵。

$$A=\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ & & \dots & \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \qquad A^{-1}=\frac{1}{|A|}A^{*}$$

A_{ij}为A中元素a_{ij}的代数余子式，则

$$A^{*}=\begin{pmatrix} A_{11} & A_{21} & \dots & A_{n1} \\ A_{12} & A_{22} & \dots & A_{n2} \\ & & \dots & \\ A_{1n} & A_{2n} & \dots & A_{nn} \end{pmatrix}$$

称为A的伴随矩阵。

矩阵的运算

```
%函数功能：用伴随矩阵求矩阵的逆 %输入矩阵A，必须是方阵
function f = inv_by_adj(A)
n = length(A);
adjA = zeros(n);
%确定伴随矩阵各个元素的值
for i = 1:n
for j = 1:n
B = A; %临时矩阵B
B(i, :) = [];%删去B的第i行第j列并计算A的相应代数余子式a
B(:, j) = [];
a = (-1)^(i + j) * det(B);
adjA(j, i) = a;   %将代数余子式的值赋值给伴随矩阵相应位置的元素
end
end
f = adjA / det(A);   %求矩阵A的逆矩阵
end
```

矩阵的运算

利用伴随矩阵法，写.m文件，求矩阵的逆。

例16 利用伴随矩阵求矩阵 $A=\begin{pmatrix} 1 & 2 & 3 \\ 1 & 3 & 4 \\ 2 & 4 & 5 \end{pmatrix}$ 的逆。

```
>> A=[1 2 3;1 3 4;2 4 5]
invA = inv_by_adj(A)
A*invA
```

作　业

一、完成以下矩阵运算

（1）输入矩阵

$$A=\begin{pmatrix} 2 & 1 & -3 & -1 \\ 3 & 1 & 0 & 7 \\ -1 & 2 & 4 & -2 \\ 1 & 0 & -1 & 5 \end{pmatrix} \qquad B=\begin{pmatrix} 2 & 3 & 5 & 4 \\ 1 & -1 & 8 & 6 \\ -3 & 4 & 6 & -7 \\ 2 & 3 & 5 & 4 \end{pmatrix}$$

（2）求A-B, A+B, A*B, B', $|A|$, A^{-1}

作 业

（3）将A, B扩展为4×8阶的矩阵C=[A B].

（4）提取C中的1，2，4 行；3，5，7列构成的新矩阵D.

（5）提取C中的3，5列构成新矩阵.

（6）建立与A同阶的单位阵，1矩阵，零矩阵.

（7）提取A矩阵中的2行3列的元素.

作　业

二、已知 $A=\begin{pmatrix} 4 & -3 & 2 \\ 3 & 0 & 5 \\ 1 & 5 & 3 \end{pmatrix}$　$B=\begin{pmatrix} 1 & 3 & 4 \\ -2 & 0 & -1 \\ 2 & -1 & 1 \end{pmatrix}$

在matlab命令窗口建立A、B矩阵，并对其进行以下操作：

（1）提取矩阵A的第一行和第三行；

（2）提取矩阵B的第一列和第二列；

（3）交换矩阵A的第一和第二行；

作 业

（4）交换矩阵*B*的第二和第三列；

（5）从横向和纵向合并矩阵*A*和*B*；

（6）将矩阵*B*的第二行第二列元素换成-9；

（7）提取矩阵*B*的主对角线元素构成对角阵；

（8）删除矩阵*A*的第二行；

（9）构造与矩阵*A*同阶的全零阵；

（10）求矩阵*B*的行列式及转置矩阵；

作　业

（11）构建矩阵C，C的第一行和第二行由矩阵A的第一和第二行的第一和第二列的元素构成，C的第三行和第四行由矩阵B的第二和第三行的第二和第三列的元素构成；

（12）进行以下计算；

〈1〉$2A-B$；　　〈2〉$A*B$和$A.*B$；

〈3〉A/B和$A\backslash B$；　　〈4〉$A.\hat{}B$；

作 业

三、利用伴随矩阵法，写.m文件，求矩阵的逆。

利用伴随矩阵求矩阵 $A=\begin{pmatrix} 1 & 2 & 3 \\ 2 & 2 & 1 \\ 3 & 2 & 2 \end{pmatrix}$ 的逆。