* Matlab 实验课报告1

姓名：\_\_\_\_\_\_王杰永\_\_\_\_\_\_\_\_\_\_\_学号\_\_\_\_\_\_\_\_03190886\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1.写一个函数，输入为一个向量，将此向量元素按由大到小重排，并输出(不要使用内部函数sort)。

代码：

% 降序排序 冒泡排序算法

function [] = Sorting(vector)

len=length(vector);

for i=len:-1:1

for j=1:i-1

if vector(j)<vector(j+1);

temp=vector(j);

vector(j)=vector(j+1);

vector(j+1)=temp;

end

end

end

disp(vector)

end

%升序排序 选择排序算法

function [] = Sorting2(vec)

len=numel(vec);

for i=1:len

indexmin=i;

for j=i+1:len

if vec(j)<vec(indexmin)

indexmin=j;

end

end

temp=vec(i);

vec(i)=vec(indexmin);

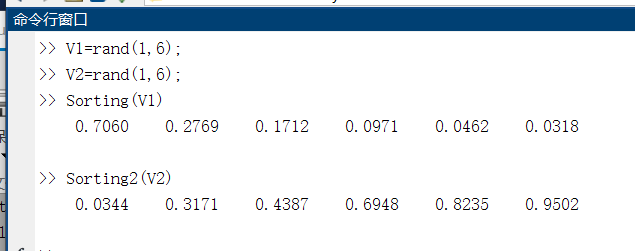
vec(indexmin)=temp;

end

disp(vec);

end

运行结果(请随机生成一个向量，然后排序)



2.写一个函数，输入为一个矩阵，输出为矩阵的最大值及其下标，使用循环。

代码：

% 矩阵的最大值以及其下标

function [] = maxInMatrix(matrix)

[row,colume]=size(matrix);

row\_index=1;

colume\_index=1;

for i=1:row

for j=1:colume

if matrix(i,j)>matrix(row\_index,colume\_index)

row\_index=i;

colume\_index=j;

end

end

end

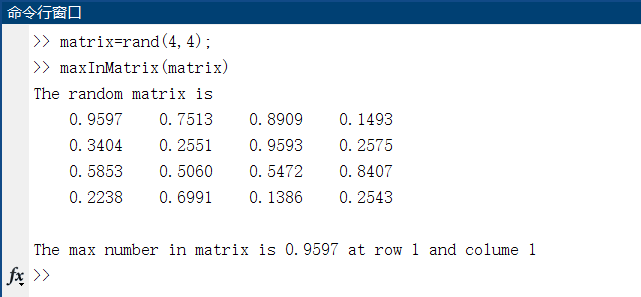
fprintf('The random matrix is\n');

disp(matrix);

fprintf('The max number in matrix is %.4f at row %d and colume %d\n',matrix(row\_index,colume\_index),row\_index,colume\_index)

end

运行结果：(随机生成一个矩阵，测试你的函数结果)



思考： 如何使用max函数实现此操作？

如果矩阵有多个最大值，如何返回多组下标？

**1.两次max实现找到最大值**

**2.在找到最大值后，遍历一次矩阵，将与最大值相等的元素的下标存在两个向量中**

**代码如下**

function[]=maxInMatrix2(matrix)

[row,colume]=size(matrix);

target=max(max(matrix));

max\_row=[];

max\_col=[];

for i=1:row

for j=1:colume

if matrix(i,j)==target

max\_row=[max\_row,i];

max\_col=[max\_col,j];

end

end

end

fprintf('The matrix is:\n');

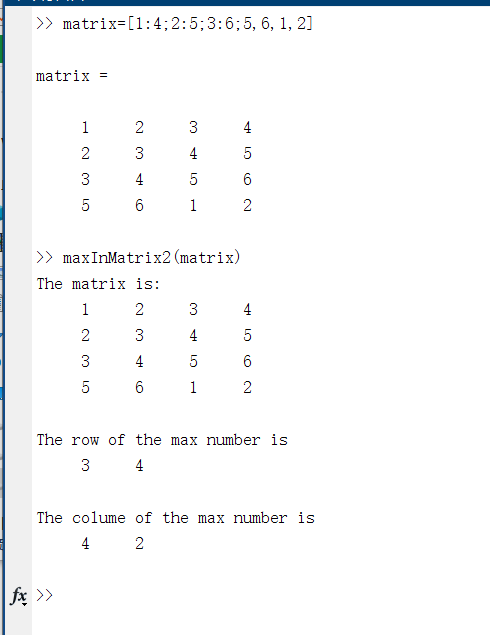
disp(matrix);

fprintf('The row of the max number is\n');

disp(max\_row);

fprintf('The colume of the max number is\n');

disp(max\_col);



3. 通过图像比较sinx与 的区别，n分别取1:8

要求： 构造8幅子图，每张图片上比较sin和截断级数的图像。

代码

% sinx与泰勒级数

x=-2\*pi:0.1:2\*pi;

y=sin(x);

y1=x;

y2=y1-x.^3./factorial(3);

y3=y2+x.^5./factorial(5);

y4=y3-x.^7./factorial(7);

y5=y4+x.^9./factorial(9);

y6=y5-x.^11./factorial(11);

y7=y6+x.^13./factorial(13);

y8=y7-x.^15./factorial(15);

subplot(2,4,1);

plot(x,y1,x,y);

subplot(2,4,2);

plot(x,y2,x,y);

subplot(2,4,3);

plot(x,y3,x,y);

subplot(2,4,4);

plot(x,y4,x,y);

subplot(2,4,5);

plot(x,y5,x,y);

subplot(2,4,6);

plot(x,y6,x,y);

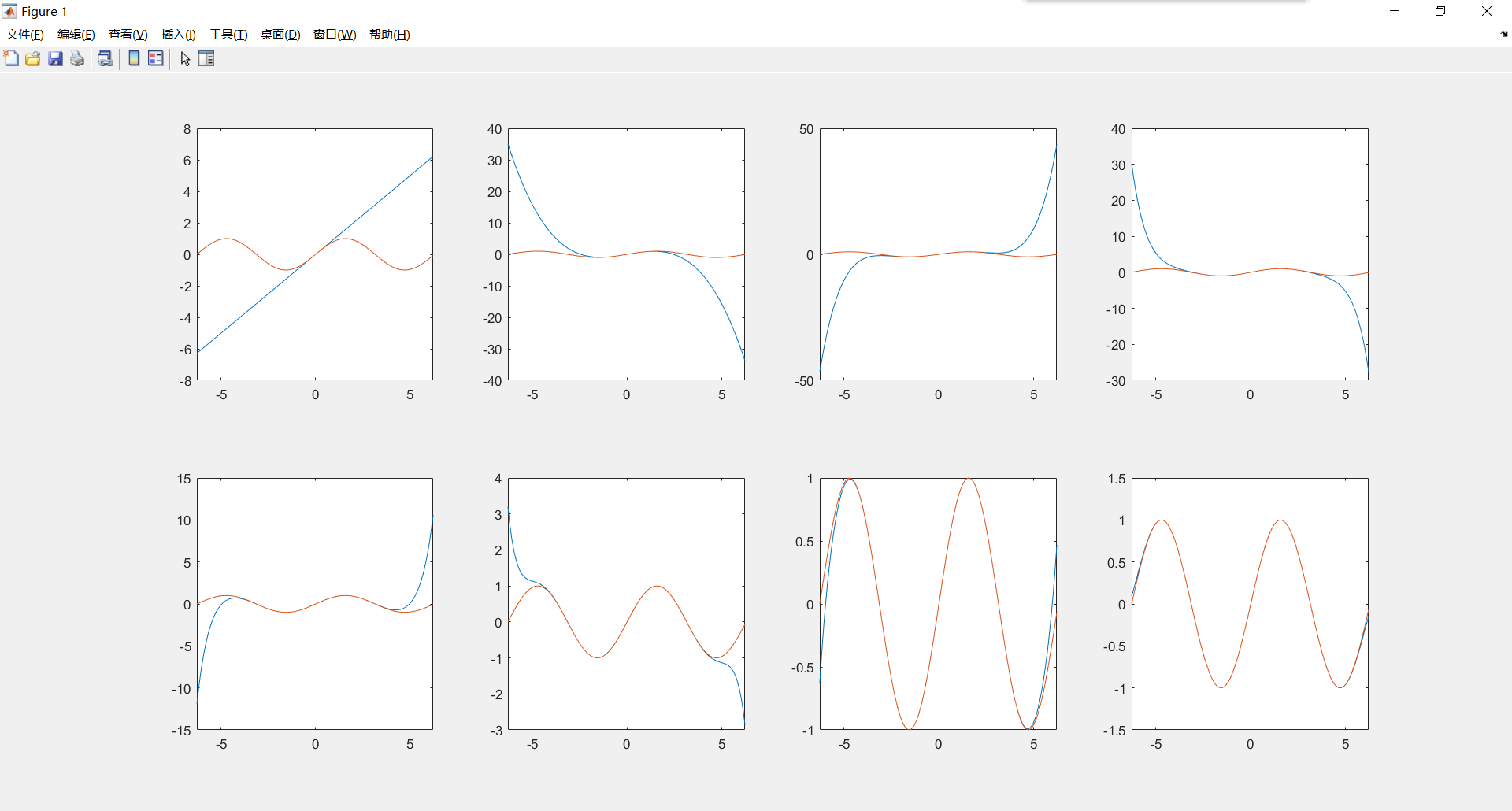
subplot(2,4,7);

plot(x,y7,x,y);

subplot(2,4,8);

plot(x,y8,x,y);

结果：(将图片插入



4.通过三维图形命令绘制曲面的图像

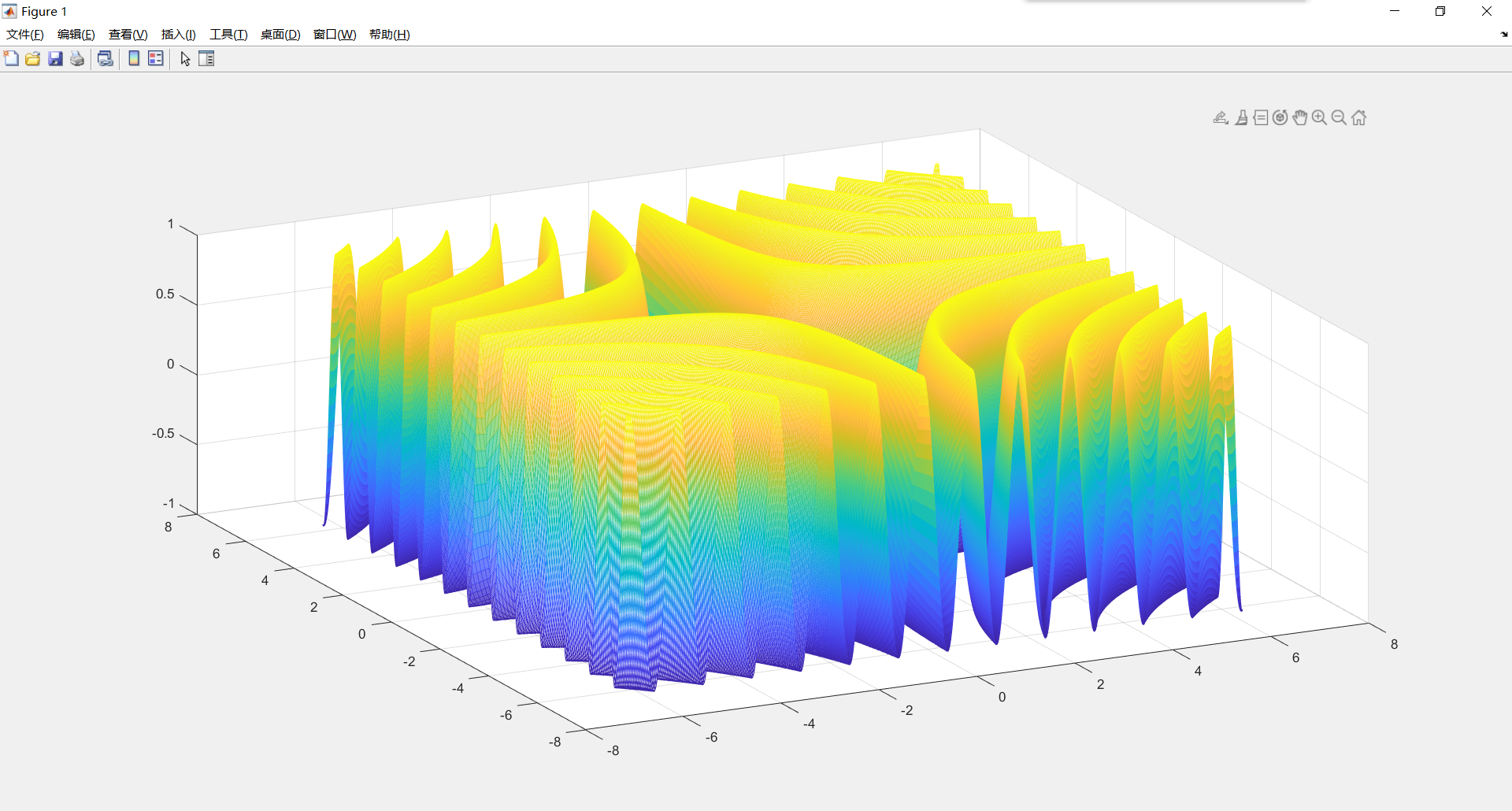
代码：

[x,y]=meshgrid(-2\*pi:0.02:2\*pi);

z=sin(x.\*y);

mesh(x,y,z);

结果：



5.绘制空间曲线,并绘制其动态曲线图，自行确定参数的范围，譬如.

代码：

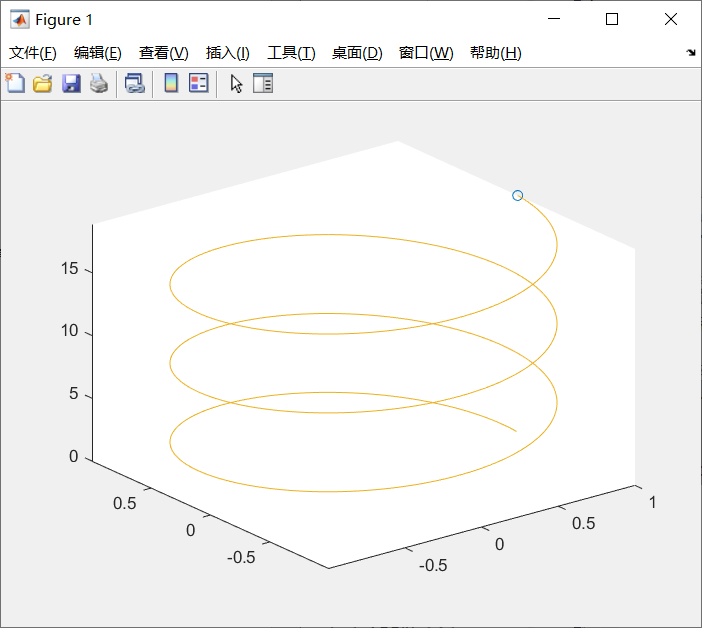
t=0:0.01:6\*pi;

x=cos(t);

y=sin(t);

comet3(x,y,t);

结果



6. 结合线性代数的内容请设计一个算法求解线性方程组 Ax=b,给出一个函数，输入为

系数矩阵A和右端向量b，输出为解x。(结果与A\b比较)

代码：

function[x]=linearEquation(A,b)

mat=[A,b]; %增广矩阵

[row,column]=size(mat); %求出合并向量的行，列值

x=zeros(row,1); %解向量的初始化

temp=zeros(row,column); %阶梯形矩阵的初始化

%阶梯形矩阵的第一行与原矩阵相同

for i=1:column

temp(1,i)=mat(1,i);

end

%每一行都与第k-1行消去，使得当前行的第一个非零元素为0，同时更新mat矩阵

for k=2:row

for j=k:row

temp(j,:)=mat(j,:)-mat(j,k-1)/mat(k-1,k-1)\*mat(k-1,:);

mat(j,:)=temp(j,:);

end

end

%开始回代

x(row)=temp(row,column)/temp(row,column-1); %得到解向量的最后一个分量

%从下向上回代求出其余分量

for i=row-1:-1:1

%先将该分量的系数化为1

x(i)=temp(i,column)/temp(i,i);

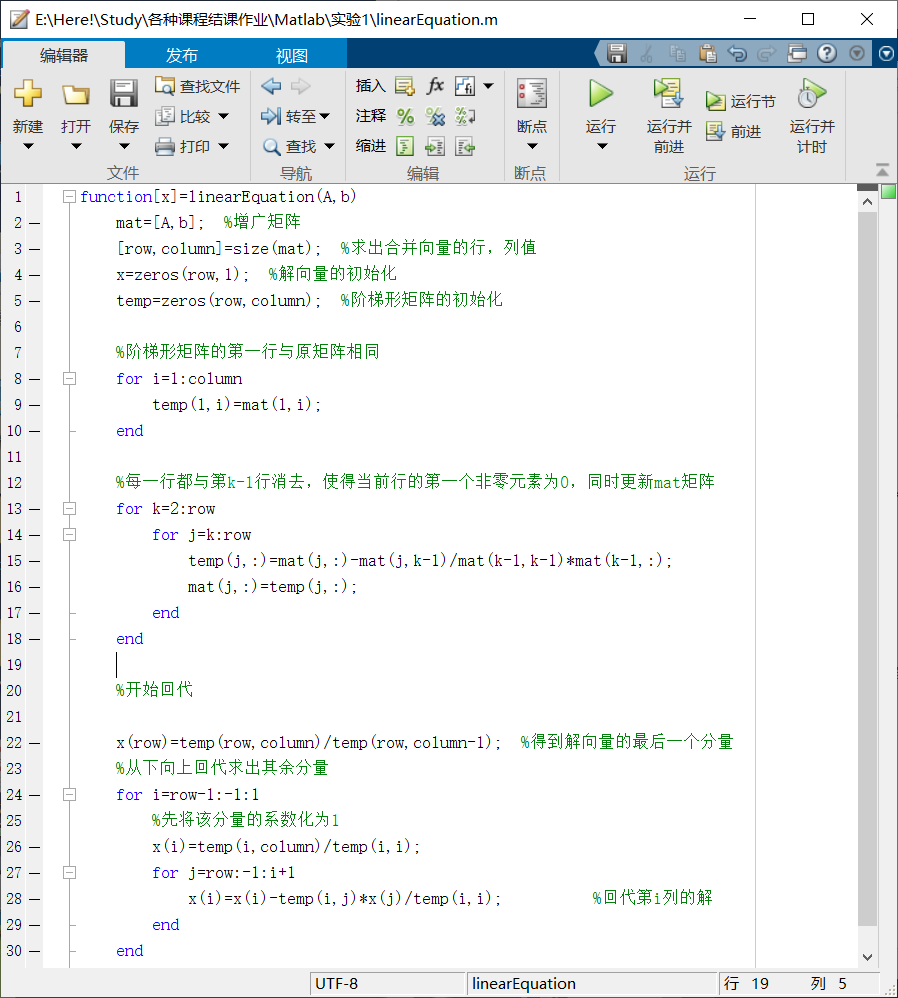
for j=row:-1:i+1

x(i)=x(i)-temp(i,j)\*x(j)/temp(i,i); %回代第i列的解

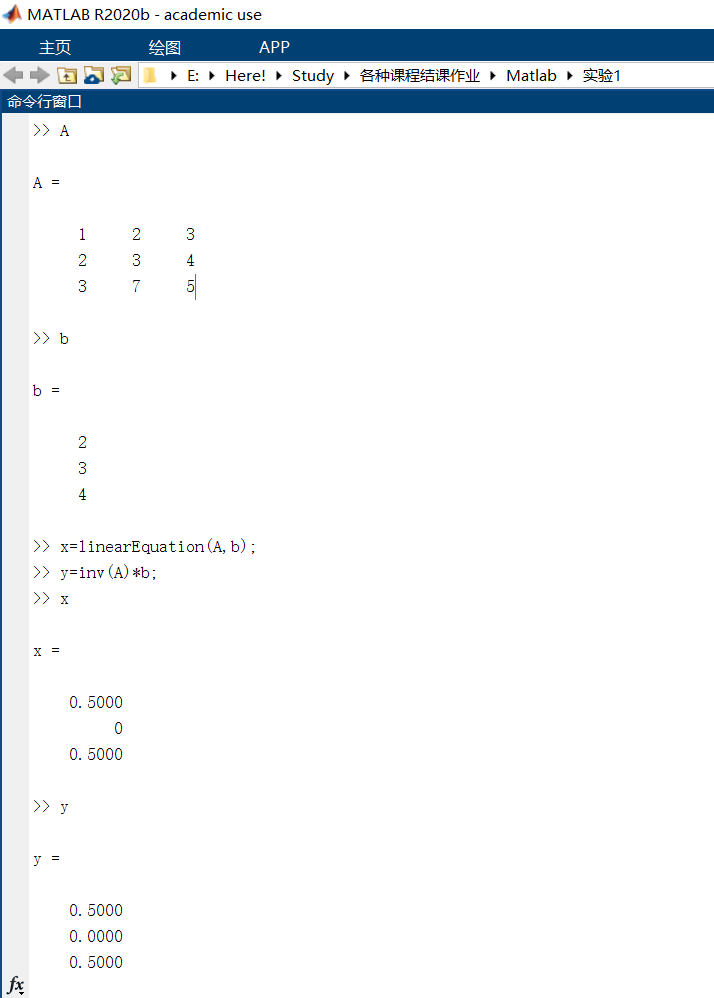
end

end

end



结果



实验感想：

本次实验让我更好的掌握了matlab的基本语法，同时加深了对一些算法理解。特别是最后通过高斯顺序消去法解线性方程组的解的算法让人印象深刻。受益匪浅。

