% 加\*为了解内容

% 生成矩阵

    % 直接法

        a = [1,2,3;4,5,6;7,8,9];

    % 冒号一维矩阵 a = 开始：步长：结束，步长为1可省略

        b = 1:1:10;  % 1,2,...10

        b = 1:10;  %与上一个等价

    % 函数生成

        % linspace(开始，结束，元素个数)，等差生成指定元素数的一维矩阵，省略个数则生成100个

            c = linspace(0,10,5);

        % 特殊矩阵

            e = eye(4);  % eye(维数)单位阵

            z = zeros(1,4);  % zeros(维数)全零阵

            o = ones(4,1);  % ones(维数)全1阵

            r = rand(4);  % rand(维数)0~1分布随机阵

            rn = randn(4);  % randn(维数)0均值Gaussian分布随机阵

%%

% 矩阵运算

    diag\_a = diag(a,1);  % diag(行向量，主对角线上方第k条斜线)用行向量生成对角阵

    tril\_a = tril(a,1);  % tril(矩阵，主对角线上方第k条斜线)生成矩阵的下三角阵，triu上三角阵

    % 加、减、乘、乘方

        a\*a

    % 点运算

        % a.\*b , a./b , a.\b , a.^b  对应元素的\*,/,\,^运算

        a.\*a

    % 逆矩阵

        pinv(a)  % 伪逆矩阵，当a不是方阵，求广义逆矩阵；当a是可逆方阵，结果与逆矩阵相同

    % 特征值，特征向量

        [v,D] = eig(a);  % 输出v为特征向量，D为特征值对角阵

    % \*行列式

        det(a)

    % \*秩

        rank(a)

    % \*伴随

        compan(b)

%%

% 矩阵的修改

    %部分替换

        chg\_a = a;

        chg\_a(2,3) = 4;  % (行，列)元素替换

        chg\_a(1,:) = [2,2,2];  % (行,:)替换行，为[]删除该行

        chg\_a(:,1) = [];  % (:,列)替换列，为[]删除该列

    % 转置

        T\_a = a';

    % 指定维数拼接

        c1\_a = cat(1,a,a);  % 垂直拼接

        c2\_a = cat(2,a,a);  % 水平拼接

    % \*变维

        rs\_a = reshape(a,1,9);  % 元素个数不变，矩阵变为m\*n

%%

% 信息获取

    % 矩阵的行列数

        [row\_a, col\_a] = size(a);  % [行数，列数]

    % 行列中最大的

        len\_a = length(a);

%%

% 多维数组

    % 创建

        % 直接法

            mul\_1(:,:,1) = [1,2,3;2,3,4];

            mul\_1(:,:,2) = [3,4,5;4,5,6];

        % \*扩展法

            mul\_2 = [1,2,3;2,3,4];

            mul\_2(:,:,2) = [3,4,5;4,5,6];  % 若不赋值第一页，第一页全为0

        % cat法

            mul\_31 = [1,2,3;2,3,4];

            mul\_32 = [3,4,5;4,5,6];

            mul\_3 = cat(3,mul\_31,mul\_32);  % 把a1a2按照“3”维连接

%%

% \*字符串

    % 创建

        str0 = 'hello world';  % 单引号引起

        str1 = 'I''m a student';  % 字符串中单引号写两遍

        str3 = ['I''m' 'a' 'student'];  % 方括号链接多字符串

        str4 = strcat(str0, str1);  % strcat连接字符串函数

        str5 = strvcat(str0, str1);  % strvcat连接产生多行字符串

        str6 = double(str0);  % 取str0的ASCII值，也可用abs函数

        str7 = char(str6);  % 把ASCII转为字符串

    % 操作

        % 比较

            strcmp(str0, str1);  % 相等为1，不等为0

            strncmp(str0, str1, 3);  % 比较前3个是否相等(n)

            strcmpi(str0, str1);  % 忽略大小写比较(i)

            strncmpi(str0, str1, 3);  % 忽略大小写比较前3个是否相等

        % 查找替换

            strfind(str0, str1);  % 在str0找到str1的位置

            strmatch(str1, str0);  % 在str0字符串数组中找到str1开头的行数

            strtok(str0);  % 截取str0第一个分隔符（空格，tab，回车）前的部分

            strrep(str0, str1, str2);  % 在str0中用str2替换str1

        % 其他

            upper(str0);  % 转大写，lower转小写

            strjust(str0, 'right');  % 将str0右对齐，left左对齐，center中间对齐

            strtrim(str0);  % 删除str0开头结尾空格

            eval(str0);  % 将str0作为代码执行

%%

%转换

    % \_\_\_2\_\_\_  -->  如num2str，将数字转字符串； dec2hex，将十进制转十六进制

    str\_b = num2str(b);

    % abs，double取ASCII码；char把ASCII转字符串

    abs\_str = abs('aAaA');

第2部分：程序结构

%%

a = 5;

x = [1, 2]; y =[3, 4];

%%

%选择结构

    %if-elseif-else-end

        if a>0

            disp(x);

        elseif a==0

            disp(a);

        else

            disp(a-1);

        end

    %switch-case-otherwise-end

        switch a

            case 0

                disp(a);

            case 1

                disp(a+1);

            otherwise

                disp('aaa');

        end

    %try-catch

        try

            z = x\*y;

        catch

            z = x.\*y;  % 若try出错，则执行

        end

        disp(z);

%%

% 循环结构

    % for 循环变量=初值:步长:终值 - end

    for i=0:1:10  % 步长为负，则初值大于终值

        disp(i);  % 循环体内不可对循环变量做修改

    end

    % while-end

    while a>2

        disp(a);

        a = a-1;

    end

%%

%程序控制

    %continue 跳过当次循环剩下语句，进入下一循环

    %break 跳出当前循环

    %return 跳出程序并返回

%%

%m文件

    %脚本文件：没有输入输出参数，执行后变量结果返回工作空间，可直接运行

        %以下是脚本文件，文件名假设为exp.m

        %\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

            clear

            r = 5;

            s = pi\*r\*r;

            p = 2\*pi\*r;

            disp(s)

            disp(p)

        %\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        %以下是调用

        %\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

            exp

        %\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

    %函数文件：以function开头，有输入输出，变量为局部变量不返回工作空间，需要调用

        %以下是函数文件

        %\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

            function [s, p] = circ(r)  % 文件命名应与函数名一致，系统找文件名circ.m

            %CIRC 计算圆面积和周长  % 简单说明

            %参数：输入参数r:圆半径；输出参数s:圆面积，p:周长  % 详细说明

                s = pi\*r\*r;

                p = 2\*pi\*r;

            end

        %\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        %以下是调用

        %\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

            [a, b] = circ(5);  % 返回为多个参数时，若写a = circ(5)则保留第一个返回值

        %\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        %\*以下是带子函数的函数文件

        %\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        function y = key(w)  % 主函数放第一个，函数名为key

            if w==0

                y = type0(w);  % 调用子函数type0

            else

                y = type1(w);

            end

        end

        function y0 = type0(a)  % 子函数，各子函数之间顺序无所谓

            y0 = a+1;

        end

        function y1 = type1(a)

            y1 = a+4;

        end

        %\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        %\*函数输入输出参数可以不定

        %nargin：输入参数个数，nargout：输出参数个数

        %varargin：输入参数内容的元胞数组，varargout：输出参数

        %以下是函数文件

        %\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        function varargout = idk(varargin)

            x = length(varargin);

            varargout{1} = x;

            varargout{2} = x+1;

        end

        %\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

第3部分：图像绘制

%%

x = 0:0.1:2\*pi;

y1 = sin(x);

y2 = cos(x);

%%

% 二维曲线绘制

    % 基本函数

        % plot(y)

            % y为向量

                plot(y1);  % 纵坐标为y的值；横坐标自动为元素序号(角标+1)，此处为1~9

            % y为矩阵

                figure;  % 开启新绘图窗口，下一次绘图在新窗口

                y = [y1', y2'];

                plot(y);  % 当y为矩阵，按每一列画出曲线，颜色自动区分

        % plot(x, y)

            % xy为向量

                plot(x, y1);  % 先绘制曲线

        % plot(x1, y1, x2, y2...)

            plot(x, y1, x, y2);  % 在同一个窗口同一坐标轴绘制多条曲线

    % 线性图形格式设置

        % 线形颜色数据点

            plot(x, y1, 'b:o');  % 蓝色 点线 圆圈

            % b蓝 g绿 r红 c青 m紫 y黄 k黑 w白

            % -实线 :点线 --虚线 -.点画线

            % .实点 o圆圈 x叉 +十字 \*星号 s方块 d钻石 v下三角 ^上三角 <左三角 >右三角 p五角星 h六角星

        % 坐标轴

            plot(x, y1);

            axis([-1\*pi, 3\*pi, -1.5, 1.5]);  % 规定横纵坐标范围

    % 图形修饰

        % 标题标签

            title('a title');  % 图像标题

            xlabel('this is x');  % x轴标记，同理还有ylabel，zlabel

        %图例设置

            legend('hahaha', 'location', 'best');  % str的顺序与绘图顺序一致; 'best'指图例位置最佳化，还有其他位置

        %图形保持

            plot(x, y1);

            hold on;  % 在原有窗口y1曲线上增加绘制下一个图形

            plot(x, y2);  % y2在同一窗口内被绘制

            hold off;

        %分割绘制

            subplot(2, 2, 1);  % 分割成2x2区域，在第一块区域绘制下一个图形

            plot(x, y1);  % y1被绘制在4块区域的第一块

            subplot(2, 2, 2);  % 分割方法相同，区域改变

            plot(x, y2);  % y2在第二块区域

%%

%\*二维特殊图形绘制

    %柱状图

        bar(x, y, width, '参数');  % x横坐标向量，m个元素; y为向量时，每个x画一竖条共m条，矩阵mxn时，每个x画n条;

                                   % width宽度默认0.8，超过1各条会重叠;

                                   % 参数有grouped分组式，stacked堆栈式; 默认grouped

                                   % bar垂直柱状图,barh水平柱状图,bar3三维柱状图,barh3水平三维柱状图(三维多一个参数detached, 且为默认)

    %饼形图

        pie(x, explode, 'lable');  % x为向量显示每个元素占总和百分比, 为矩阵显示每个元素占所有总和百分比

                                   % explode向量与x同长度，为1表示该元素被分离突出显示，默认全0不分离

                                   % pie3绘制三维饼图

    %直方图

        hist(y, n);  % y为向量，把横坐标分为n段绘制

        hist(y, x);  % x为向量，用于指定每段中间值, 若取N = hist(y, x), N为每段元素个数

    %离散数据图

        stairs(x, y, 'b-o');  % 阶梯图，参数同plot

        stem(x, y, 'fill');  % 火柴杆图，参数fill是填充火柴杆，或定义线形

        candle(HI, LO, CL, OP);  % 蜡烛图:HI为最高价格向量,LO为最低价格向量,CL为收盘价格向量,OP为开盘价格向量

    %向量图

        compass(u, v, 'b-o');  % 罗盘图横坐标u纵坐标v

        compass(Z, 'b-o');  % 罗盘图复向量Z

        feather(u, v, 'b-o');  % 羽毛图横坐标u纵坐标v

        feather(Z, 'b-o');  % 羽毛图复向量Z

        quiver(x, y, u, v);  % 以(x, y)为起点(u, v)为终点向量场图

    %极坐标图

        % polar(theta, rho, 'b-o');  % 极角theta, 半径rho

        theta = -pi:0.01:pi;

        rho = sin(theta);

        polar(theta, rho, 'b')

    %对数坐标图

        semilogx(x1, y1, 'b-o');  % 把x轴对数刻度表示, semilogy是y轴对数刻度表示，loglog是两个坐标都用对数表示

    %双纵坐标

        plotyy(x1, y1, x2, y2, 'fun1', 'fun2');  % fun规定了两条条线的绘制方式，如plot,semilogx,semilogy,loglog,stem等

    %函数绘图

        f = 'sin(2\*x)';

        ezplot(f, [0, 2\*pi]);  % 绘制f并规定横坐标范围，也有[xmin, xmax, ymin, ymax]

        x = '2\*cos(t)';

        y = '4\*sin(t)';

        ezplot(x, y);  % 绘制x(t),y(t)在[0, 2\*pi]图像, 也可以在最后用[tmin, tmax]规定t的范围

%%

%三维曲线曲面绘制

    %三维曲线

        x = 0:0.1:2\*pi;

        y = sin(x); z = cos(x);

        plot3(x, y, z, 'b-\*');

    %三维曲面

        %三维网格

            x = -5:0.1:5;  % 规定了x轴采样点，也规定了x轴范围

            y = -4:0.1:4;  % 规定了y轴采样点，也规定了y轴范围

            [X, Y] = meshgrid(x, y);  % 得到了xoy面网格点

            Z = X.^2+Y.^2;

            mesh(X, Y, Z)  % XY是meshgrid得到的网格点，Z是网格顶点，c是用色矩阵可省略

        %三维表面图

            x = -5:0.1:5;

            y = -4:0.1:4;

            [X, Y] = meshgrid(x, y);

            Z = X.^2+Y.^2;  % 以上部分同上

            surf(X, Y, Z)  % 与上一个类似

第4部分：多项式

%%

%多项式

    %创建

        p = [1, 2, 3, 4];  % 系数向量，按x降幂排列，最右边是常数

        f1 = poly2str(p, 'x');  % 生成好看的字符串 f1 = x^3 + 2 x^2 + 3 x + 4，不被认可的运算式

        f2 = poly2sym(p);  % 生成可用的符号函数 f2 = x^3 + 2\*x^2 + 3\*x + 4

    %求值

        x = 4;

        y1 = polyval(p, x);  % 代入求值；若x1为矩阵，则对每个值单独求值

    %求根

        r = roots(p); % p同上，由系数求根，结果为根植矩阵

        p0 = poly(r);  % 由根求系数，结果为系数矩阵

%%

%数据插值

    %一维插值

        %yi = interp1(X, Y, xi, 'method')

        X = [-3, -1, 0, 1, 3];

        Y = [9, 1, 0, 1, 9];  % XY为已知点横纵坐标向量

        y2 = interp1(X, Y, 2);  % 差值预估在x=2的y的值，x不能超过已知范围(此处x<3)

        y2m = interp1(X, Y, 2, 'spline');  % 用spline方法(三次样条)差值预估在x=2的y的值

    %二维插值

        %zi = interp1(X, Y, Z, xi, yi, 'method')

%%

X = [2, 3, 9, 15, 6, 7, 4];

A = [1, 7, 2; 9, 5, 3; 8, 4 ,6];

B = [1, 7, 3; 9, 5, 3; 8, 4 ,6];

%数据统计

    %矩阵最大最小值

        y = max(X);  % 求矩阵X的最大值，min最小值

        [y, k] = max(X);  % 求最大值，k为该值的角标

        [y, k] = max(A, [], 2);  % A是矩阵，'2'时返回y每一行最大元素构成的列向量，k元素所在列；'1'时与上述相同

    %均值和中值

        y = mean(X);  % 均值

        y = median(X);  % 中值

        y = mean(A, 2);  % '2'时返回y每一行均值构成的列向量；'1'时与上述相同

        y = median(A, 2);  % '2'时返回y每一行中值构成的列向量；'1'时与上述相同

    %排序

        Y = sort(A, 1, 'ascend');  % sort(矩阵, dim, 'method')dim为1按列排序，2按行排序；ascend升序，descend降序

        [Y, I] = sort(A, 1, 'ascend');  % I保留了元素之前在A的位置

    %求和求积累加累乘

        y = sum(X);  % 求和

        y = prod(X);  % 求积

        y = cumsum(X);  % 累加

        y = cumprod(X);  % 累乘

%%

%\*数值计算

    %最(极)值

        %多元函数在给定初值附近找最小值点

        x = fminsearch(fun, x0);

    %函数零点

        x = fzero(fun, x0);  % 在给定初值x0附近找零点

第5部分：符号函数

%%

%符号对象创建

    %sym函数

        p = sin(pi/3);

        P = sym(p, 'r');  % 用数值p创建符号常量P；'d'浮点数'f'有理分式的浮点数'e'有理数和误差'r'有理数

    %syms函数

        syms x;  % 声明符号变量

        f = 7\*x^2 + 2\*x+9;  % 创建符号函数

    %符号运算

        % 加减乘除外

        % '转置 ； ==相等 ； ~=不等

        % sin, cos, tan; asin, acos, atan 三角反三角

        % sinh, cosh, tanh; asinh, acosh, atanh 双曲反双曲

        % conj复数共轭；real复数实部；imag复数虚部；abs复数模；angle复数幅角

        % diag矩阵对角；triu矩阵上三角；tril矩阵下三角；inv逆矩阵；det行列式；rank秩；poly特征多项式；

        % |----expm矩阵指数函数；eig矩阵特征值和特征向量；svd奇异值分解；

    %符号对象精度转换

        digits;  % 显示当前用于计算的精度

        digits(16);  % 将计算精度改为16位，降低精度有时可以加快程序运算速度或减少空间占用

        a16 = vpa(sqrt(2));  % vpa括起的运算使sqrt(2)运算按照规定的精度执行

        a8 = vpa(sqrt(2), 8);  % 在vpa内控制精度，离开这一步精度恢复

%%

%符号多项式函数运算

    %\*符号表达形式与相互转化

        %多项式展开整理

            g = expand(f);  % 展开

            h = collect(g);  % 整理(默认按x整理)

            h1 = collect(f, x);  % 按x整理（降幂排列）

        %因式分解展开质因数

            fac = factor(h);  % 因式分解

            factor(12);  % 对12分解质因数

    %符号多项式向量形式与计算

        syms a b c;

        n = [a, b, c];

        roots(n);  % 求符号多项式ax^2+bx+c的根

        n = [1, 2, 3];

        roots(n);  % 求符号多项式带入a=1, b=2, c=3的根

    %\*反函数

        fi = finverse(f, x);  % 对f中的变量x求反函数

%%

%符号微积分

    %函数的极限和级数运算

        % 常量a，b

        %极限

            limit(f, x, 4);  % 求f(x), x->4

            limit(f, 4);  % 默认变量->4

            limit(f);  % 默认变量->0

            limit(f, x, 4, 'right');  % 求f(x), x->4+, 'left' x->4-

        %\*基本级数运算

            %求和

                symsum(s, x, 3, 5);  % 计算表达式s变量x从3到5的级数和，或symsum(s, x, [a b])或symsum(s, x, [a;b])

                symsum(s, 3, 5);  % 计算s默认变量从3到5的级数和

                symsum(s);  % 计算s默认变量从0到n-1的级数和

            %一维泰勒展开

                taylor(f, x, 4);  % f在x=4处展开为五阶泰勒级数

                taylor(f, x);  % f在x=0处展开为五阶泰勒级数

                taylor(f);  % f在默认变量=0处展开为五阶泰勒级数

    %符号微分

        %单变量求导（单变量偏导）

            n = 1;  % 常量n

            fn = diff(f, x, n);  % f对x的n阶导

            f1 = diff(f, x);  % f对x的1阶导

            diff(f, n);  % f对默认变量的n阶导

            diff(f);  % 默认变量1阶导

        %多元偏导

            fxy = diff(f, x, y);  % 先求x偏导，再求y偏导

            fxyz = diff(f, x, y, z);  % 先求x偏导，再求y偏导,再求z偏导

    %符号积分

        %积分命令

            int(f, x, 1, 2);  % 函数f变量x在1~2区间定积分

            int(f, 1, 2);  % 函数f默认变量在ab区间定积分

            int(f, x);  % 函数f变量x不定积分

            int(f);  % 函数f默认变量不定积分

            % 傅里叶，拉普拉斯，Z变换

%%

%\*符号方程求解

    %符号代数方程

        %一元方程

            eqn1 = a\*x==b;

            S = solve(eqn1);  % 返回eqn符号解

        %多元方程组

            eqn21 = x-y==a;

            eqn22 = 2\*x+y==b;

            [Sx, Sy] = solve(eqn21, eqn22, x, y);  % [Svar1,...SvarN]=solve(eqn1,...eqnM, var1,...varN),MN不一定相等

            [Sxn, Syn] = solve(eqn21, eqn22, x, y, 'ReturnCondition', true);  % 加上参数ReturnCondition可返回通解及解的条件

            % 其他参数(参数加上true生效)

                % IgnoreProperty，忽略变量定义时一些假设

                % IgnoreAnalyticConstraints，忽略分析限制；

                % MaxDegree，大于3解显性解；

                % PrincipleValue，仅主值

                % Real，仅实数解

        %非线性fsolve

            X = fsolve(fun, X0, optimset(option));  % fun函数.m文件名；X0求根初值；option选项如('Display','off')不显示中间结果等