



武汉理工大学
WUHAN UNIVERSITY OF TECHNOLOGY

MATLAB 数值计算 方法与实践

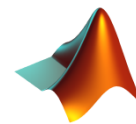
主讲老师：余文壘

Email: yuwenzhao1989@qq.com

QQ: 705417271

2018/3/8

- 2015年毕业于华中科技大学电气学院，博士；
- 武汉理工大学交通学院，讲师。



⑩ MATLAB数值计算方法与实践，1学分，1-8周

⑩ 考试方式：出勤（20%）、作业完成（80%）

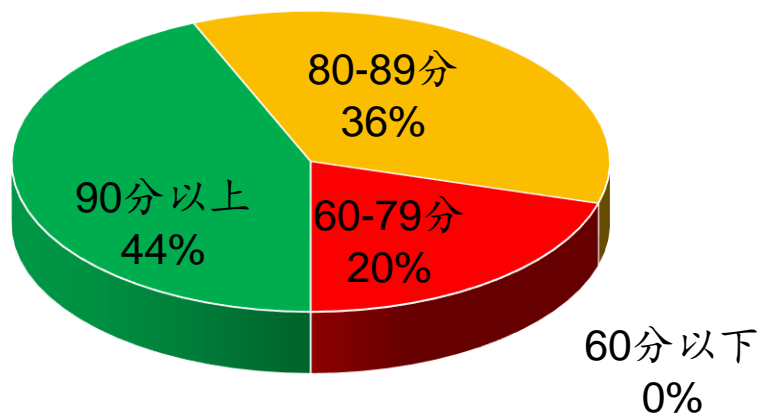
课后作业
(电子版本)

课后作业汇总
(纸质版本)

30%

70%

2016-2017第二学期 成绩分布



⑩ 不及格：

- 缺勤超过4次
- 多次督促均未交作业

⑩ 态度



⑩ 课程内容



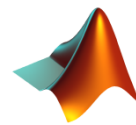
10 参考书目

- [美] John H.Mathews, Kurtis D. Fink著, 《数值方法 (Matlab版) 》, 电子工业出版社, 2010。
- [美] Shoichiro Nakamura著, 《科学计算引论-基于MATLAB的数值分析》, 电子工业出版社, 2002。
- [美] Cleve B.Moler著 张志勇等编译, 《MATLAB数值计算 (2013修订版中译本) 》

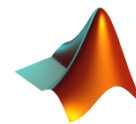
第一章 MATLAB基础知识介绍



- ⑩ MATLAB 简介
- ⑩ MATLAB 编码规则和规范
- ⑩ MATLAB 数学运算
- ⑩ MATLAB 程序流程控制
- ⑩ MATLAB 函数定义及程序调试
- ⑩ MATLAB 二维绘图



- ⑩ MATLAB 简介
- ⑩ MATLAB 编码规则和规范
- ⑩ MATLAB 数学运算
- ⑩ MATLAB 程序流程控制
- ⑩ MATLAB 函数定义及程序调试
- ⑩ MATLAB 二维绘图



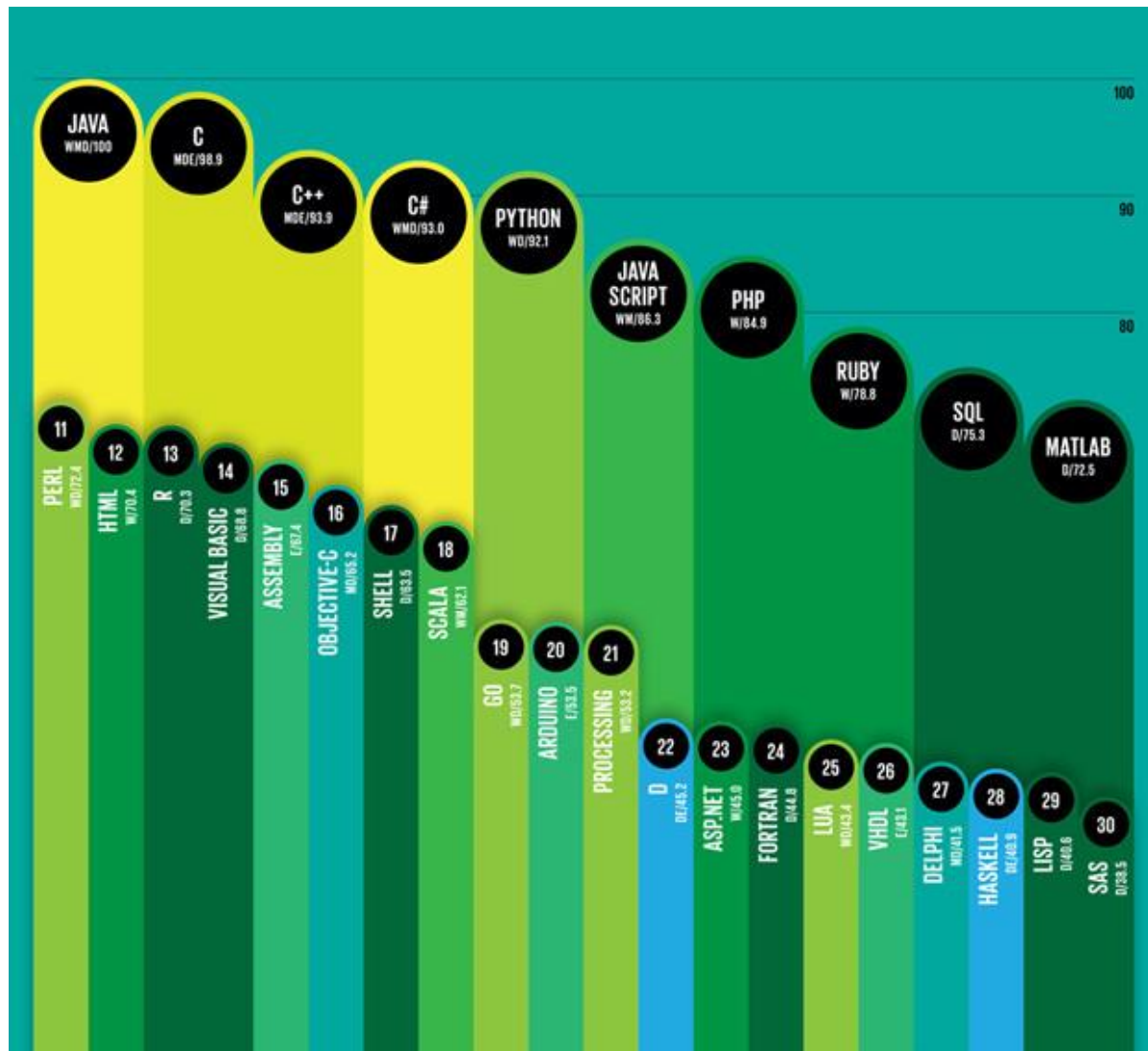
MATLAB 简介



- ⑩ MATLAB是MATrix LABoratory两个单词合成而来，也即“矩阵实验室”。
- ⑩ MATLAB 最初用于科学计算，其核心计算技术原先是源自于70年代早期Cleve Moler博士和其同事开发的LINPACK及EISPACK（Fortran编写）。
- ⑩ Jack Little 、 Cleve Moler 将MATLAB以C语言重写，并于1984年成立MathWorks公司，首次推出MATLAB 1.0商用版。



MATLAB流程度度（1）



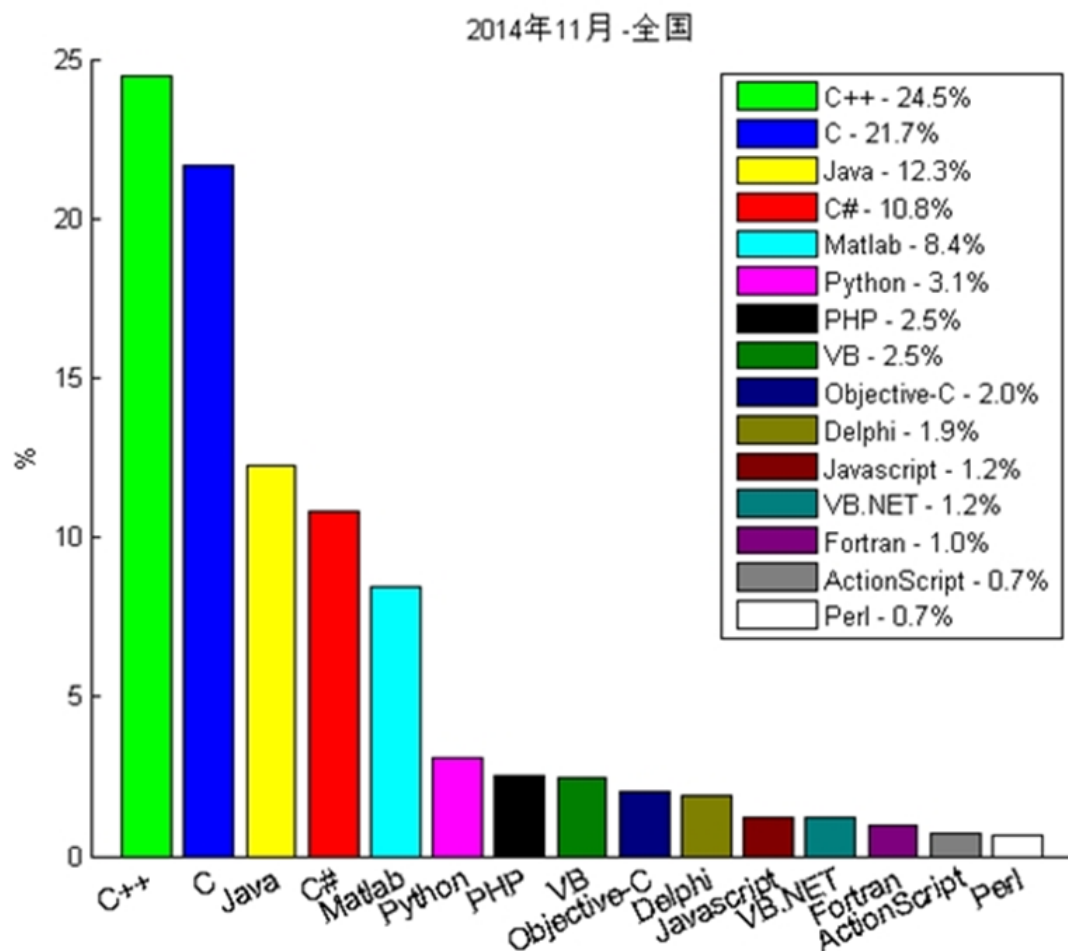
2014 IEEE Spectrum
最流行编程语言排行榜

排行榜筛选了12项指标，综合了10个来源（含IEEE Xplore、Google、GitHub）的数据，最终评选出这个排行榜（满分100，得分越高排名越靠前）。



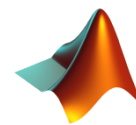
MATLAB流程度（2）

12



CodeForge网站最新公布2014年11月“我最喜爱的编程语言”排行榜

来自全国范围内11月份对五万六千多名软件工程师的问卷调查。调查选项为单选，每个参与调查者只能选择一个自己最喜欢的编程语言。



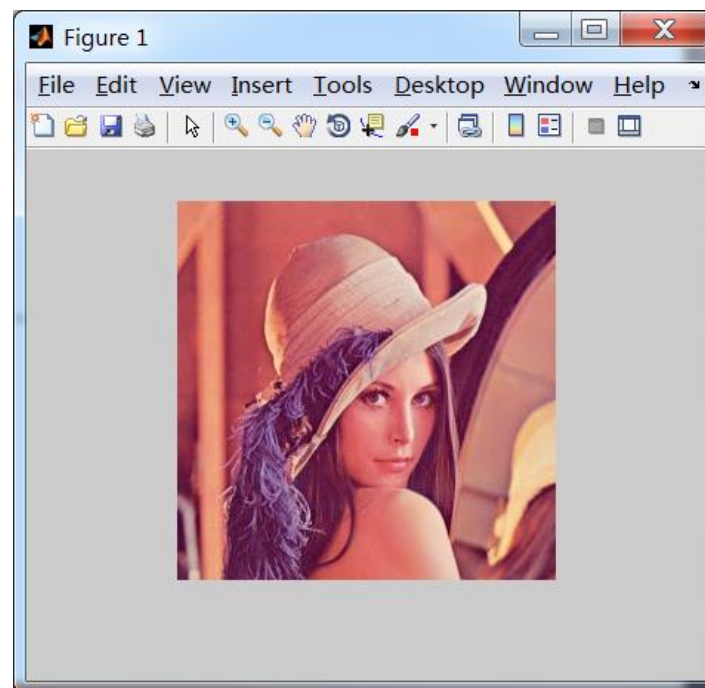
- ⑩ MATLAB 语言的简洁高效性
- ⑩ MATLAB 语言的科学运算功能
- ⑩ MATLAB 语言的绘图功能
- ⑩ MATLAB 庞大的工具箱与模块集
- ⑩ MATLAB 强大的动态系统仿真功能

⑩ BMP读取

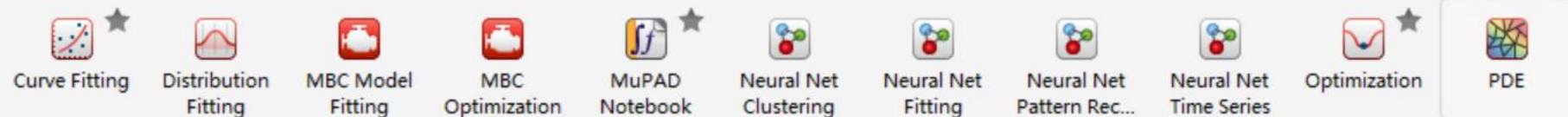
- C语言实现: Bmp.c BmpUtil.c (1500行代码)
- Matlab实现:

```
>> imshow('lena.bmp')
```

```
>>
```



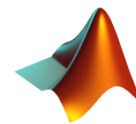
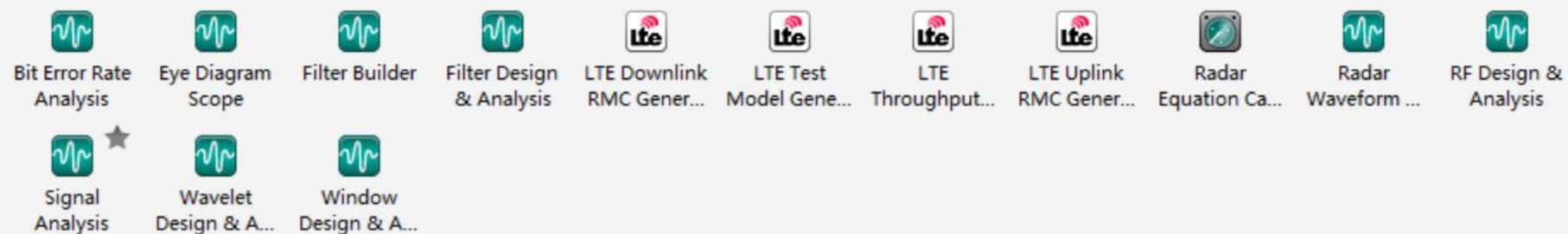
数学、统计和优化



控制系统设计和分析



信号处理和通信



图像处理和计算机视觉



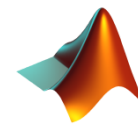
测试和测量



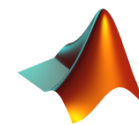
计算金融学

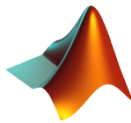
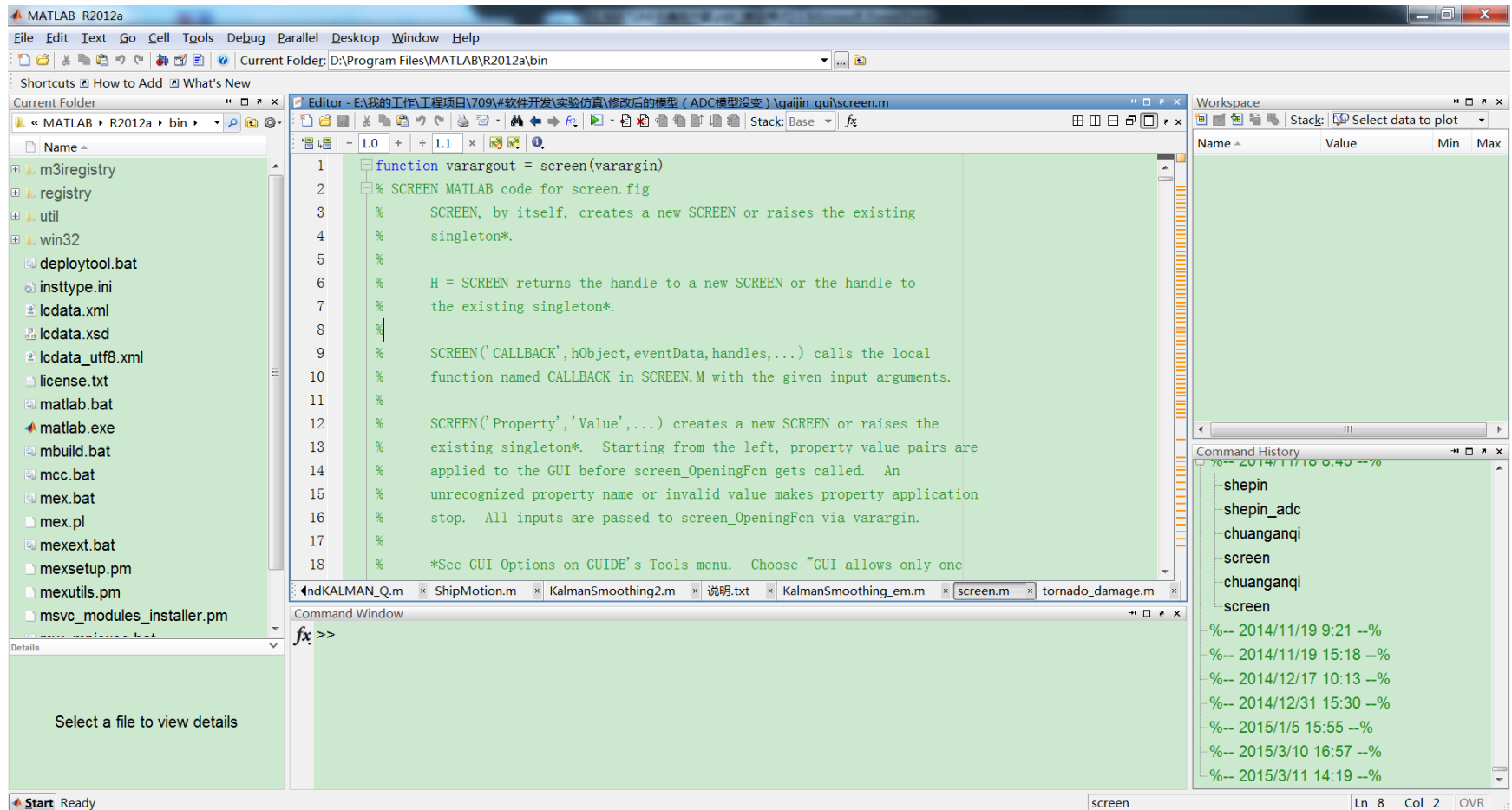


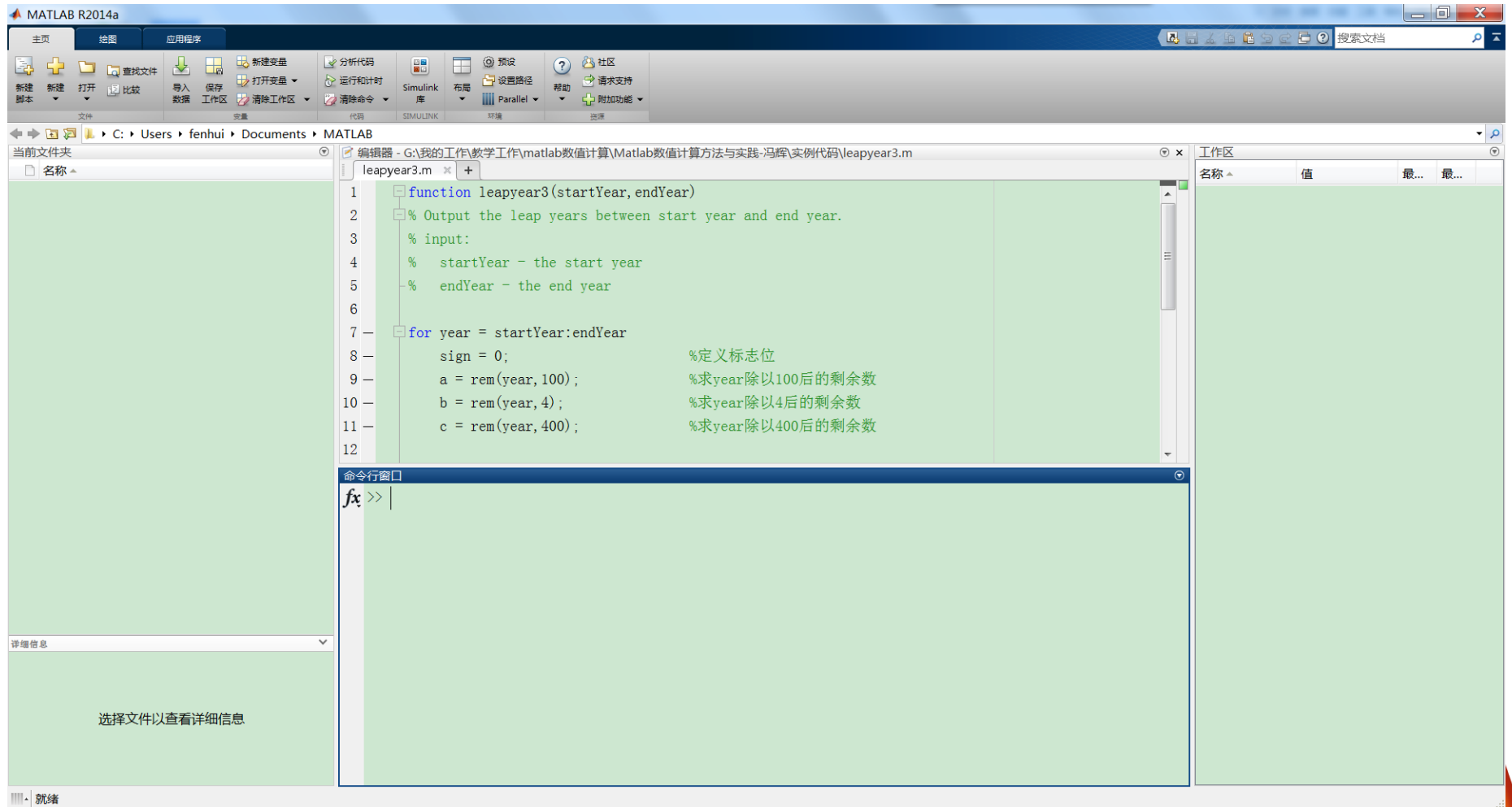
计算生物学



年份	版本号	
1984	1.0	
.....	
	三月	九月
2006	R2006a (7.2)	R2006b (7.3)
.....
2014	R2014a (8.3)	R2014b (8.4)
2015	R2015a (8.5)	R2015b (8.6)







⑩ 定义和初始化

- C/C++中要先定义变量，如果未初始化，其值是不确定的
- Matlab中变量在写入时不需预先定义（默认均为double类型，8个字节），但在读取时应该已被定义

⑩ 数组长度

- C/C++中数组长度固定，可用malloc/new函数扩展内存空间，并复制已有值到新开辟的连续内存空间
- Matlab中变量长度、维度在写入时都可以自动扩展
 - 建议：为了提高性能，最好避免变量维数自动扩展，而是预先定义空的多维变量
 - 注意：Matlab变量长度的自动扩展只在写入变量时发生，读取变量时则不可访问越界
 - 例如：对于定义的变量 $a=[1\ 2;3\ 4]$ ，赋值语句 $a(5,5)=1$ 会自动扩展，而 $b=a(10,10)$ 则会访问越界出错

⑩ 自增操作

- C/C++可用 $x++$ 或 $++x$
- Matlab只能 $x = x+1$;

⑩ 数组下标

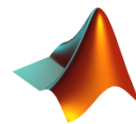
- C/C++: 下标从0开始, 用中括号表示。如 $a[0]$, 二维数组 $a[0][1]$
- Matlab: 下标从1开始, 用小括号表示, 如 $a(1)$, 二维数组 $a(1, 2)$

⑩ 函数定义

- C语言：函数只能有一个返回值，多个返回值可通过定义结构体、使用指针作为参数传入函数实现
- Matlab：函数可有多个返回值，通过[a b] = function()获取

本章主要内容

- ⑩ MATLAB简介
- ⑩ MATLAB 编码规则和规范
- ⑩ MATLAB 数学运算
- ⑩ MATLAB 程序流程控制
- ⑩ MATLAB 函数定义及程序调试
- ⑩ MATLAB二维绘图



MATLAB 编码规则和规范



- ⑩ Matlab中所有的变量都是用**矩阵**形式来表示的，即所有的变量都表示一个矩阵或者一个向量。
- ⑩ 命名规则如下：
 - 变量名对大小写敏感；
 - 变量名的第一个字符必须为英文字母，其长度不能超过31个字符；
 - 变量名可以包含下划线（_）、数字，但不能包含空格符、标点。

尽量不用内置变量作为变量名

ans	预设的计算结果的变量名
eps	MATLAB定义的正的极小值=2.2204e-16
pi	内建的 π 值
inf	∞ 值
NaN	无法定义一个数目
i	虚数单位
nargin	函数输入参数个数
nargout	函数输出参数个数
realmax	最大的正实数
realmin	最小的正实数
flops	浮点运算次数



⑩ 常量、变量、函数命名

- 名字应该能够反应他们的意义或者用途
- 变量小写开头，第二个字母首字母大写（如maxValue）
- 函数首字母均大写（如KalmanFilter）
- 常量全大写（如TOTAL_COUNT）
 - 在程序除申明部分以外，最好不要出现数值，除非这个值在程序整个生命周期内都不会改变。

⑩ 变量申明

- 二维及多维变量，如果已知变量维度，最好对其申明并初始化

注释 (Comments)

29

```
function y = mean(x,dim)
```

```
%MEAN Average or mean value.
```

```
% For vectors, MEAN(X) is the mean value of the elements in X. For  
% matrices, MEAN(X) is a row vector containing the mean value of  
% each column. For N-D arrays, MEAN(X) is the mean value of the  
% elements along the first non-sing
```

函数功能介绍

函数使用注意事项

```
% MEAN(X,DIM) takes the mean along the dimension DIM of X.
```

```
%
```

```
% Example: If X = [1 2 3; 3 3 6; 4 6 8; 4 7 7];
```

函数使用范例

```
%
```

```
% then mean(X,1) is [3.0000 4.5000 6.0000] and
```

```
% mean(X,2) is [2.0000 4.0000 6.0000 6.0000].'
```

```
%
```

```
% Class support for input X:
```

```
% float: double, single
```

输入类型要求说明

```
% See also MEDIAN, STD, MIN, MAX, VAR, COV, MODE.
```

```
% Copyright 1984-2009 The MathWorks, Inc.
```

```
% $Revision: 5.17.4.5 $ $Date: 2010/09/02 13:35:22 $
```

版权及更新日志

```
if nargin==1,
```

```
% Determine which dimension SUM will use
```

```
dim = find(size(x)~=1, 1 );
```

```
if isempty(dim), dim = 1; end
```

函数体注释

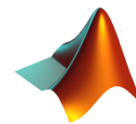
```
y = sum(x)/size(x,dim);
```

```
else
```

```
y = sum(x,dim)/size(x,dim);
```

```
end
```

- ⑩ MATLAB 简介
- ⑩ MATLAB 编码规则和规范
- ⑩ MATLAB 数学运算
- ⑩ MATLAB 程序流程控制
- ⑩ MATLAB 函数定义及程序调试
- ⑩ MATLAB 二维绘图



MATLAB 数学运算



向量和矩阵的表示

⑩ 向量（一维矩阵）的表示

```
>> s = [1 2 3 4];           % 注意 [] 的使用, 及各数字间的空白间隔
```

```
>> t = 2*s
```

```
t =
```

```
    2    4    6    8
```

```
>> s(3) = 5; s(4) = []; s(6) = 10;           % 修改向量中的元素
```

```
>> s
```

```
s =
```

```
    1    2    5    0    0   10
```


向量和矩阵的表示 (2)

⑩ 利用冒号表达式生成特定向量

```
s = s1:step:s2;
```

例：

```
>> s = 1:0.5:4
```

```
s =
```

```
1.0000 1.5000 2.0000 2.5000 3.0000 3.5000 4.0000
```

```
>> s = pi:-0.5:1
```

```
s =
```

```
3.1416 2.6416 2.1416 1.6416 1.1416
```

- 该函数生成一个行向量s，其中s1是起始值，step是步长（若省略步长为1），s2是结束值。

向量和矩阵的表示 (3)

⑩ 矩阵的表示

```
>> A = [1 2 3 4; 5 6 7 8; 9 10 11 12];    % 建立 3×4 的矩阵 A
>> A                                          % 显示矩阵 A 的内容
A =
     1     2     3     4
     5     6     7     8
     9    10    11    12
```



向量和矩阵的表示 (3)

⑩ 生成子矩阵

A =

1	2	3	4
5	6	7	8
9	10	11	12

```
>> B1 = A(1:2:end,:) %提取全部奇数行、所有列
```

B1 =

1	2	3	4
9	10	11	12

```
>> B2 = A([3,2,1],[2,3,4]) %提取3, 2, 1行、2, 3, 4列构成子矩阵
```

B2 =

10	11	12
6	7	8
2	3	4



向量和矩阵的表示 (4)

```
>> A = [A B']    % 将矩阵 B 转置后、再以行向
                % 量并入矩阵 A
```

A =

```
1  2  3  4 10  6  2
5  6  7  8 11  7  3
9 10 11 12 12  8  4
```

```
>> A(:,[1 4]) = [] % 删除第一、四列 (: 代表
                    % 所有行, []是空矩阵)
```

A =

```
2  3 10  6  2
6  7 11  7  3
10 11 12  8  4
```

A =

```
1  2  3  4
5  6  7  8
9 10 11 12
```

B =

```
10 11 12
6  7  8
2  3  4
```

+	加法运算，适用于两个数或两个同阶矩阵相加.
-	减法运算
*	乘法运算
.*	点乘运算
/	除法运算
./	点除运算
^	乘幂运算
.^	点乘幂运算
\	反斜杠表示左除

⑩ 矩阵加减法

$$C=A+B$$

$$D=A-B$$

- 注意维数是否相等
- 注意其一为标量的情形

⑩ 矩阵乘法

$$C = A*B$$

- 注意维数相容性

⑩ 矩阵除法

- 矩阵左除： $AX = B$ ，求 X

MATLAB 求解： $X=A \setminus B$

- 若 A 为非奇异方阵，则 $X=A^{-1}B$
- 若 A 不是方阵（最小二乘解）

- 矩阵右除： $XA = B$ ，求 X

MATLAB 求解： $X=B/A$

- 若 A 为非奇异方阵，则 $X=BA^{-1}$
- 若 A 不是方阵（最小二乘解）

例： $AX = B$

>> $A \setminus B$

Warning: Rank deficient, rank = 2, tol = 7.347630e-15.

ans =

-1.3158 -1.3421 -1.3684

0 0 0

1.2632 1.3684 1.4737

0 0 0

0 0 0

>> rank(A)

ans =

2

A =

2	3	10	6	2
6	7	11	7	3
10	11	12	8	4

B =

10	11	12
6	7	8
2	3	4

例 : $XA = B$

`>> B/A`

Warning: Rank deficient, rank = 2, tol = 1.047382e-14.

ans =

-0.2678 0 1.0479

0.0681 0 0.5964

`>> rank(A)`

ans =

2

A =

2	3	10	6	2
6	7	11	7	3
10	11	12	8	4

B =

10	11	12	3	4
6	7	8	5	2

⑩ 点运算——矩阵对应元素的直接运算

MATLAB 实现: $C=A.*B$ $C=A./B$

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

```
>> B = A.*A
```

B =

1	4	9
16	25	36
49	64	81

```
>> C = B./A
```

C =

1	2	3
4	5	6
7	8	9

⑩ MATLAB中对数值型变量来说，非 0 表示逻辑 1

⑩ 逻辑运算（相应元素间的运算）

- 与运算 $A \& C$
- 或运算 $A | C$
- 非运算 $\sim A$
- 异或运算 $\text{xor}(A, C)$

⑩ 例:

>> A&C

ans =

1 0 1

1 1 1

1 1 0

>> A|C

ans =

1 1 1

1 1 1

1 1 1

>> xor(A,C)

ans =

0 1 0

0 0 0

0 0 1

>> ~C

ans =

0 1 0

0 0 0

0 0 0

A =

1 2 3

4 5 6

7 8 0

C =

1 0 3

4 5 6

7 8 9

⑩ MATLAB支持的比较运算符包括:

$>$, $>=$, $<$, $<=$, $==$, $\sim=$

```
>> A = rand(1,10)
```

```
A =
```

```
    0.1067    0.9619    0.0046    0.7749    0.8173    0.8687    0.0844  
    0.3998    0.2599    0.8001
```

```
>> B = find(A>0.5)
```

```
B =
```

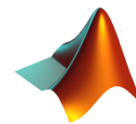
```
     2     4     5     6    10
```

```
>> C = A(B)
```

```
C =
```

```
    0.9619    0.7749    0.8173    0.8687    0.8001
```

- ⑩ MATLAB 简介
- ⑩ MATLAB 编码规则和规范
- ⑩ MATLAB 数学运算
- ⑩ MATLAB 程序流程控制
- ⑩ MATLAB 函数定义及程序调试
- ⑩ MATLAB 二维绘图



MATLAB 程序流程控制



⑩ for循环 (for-loop)

```
for 变数 = 向量  
    表达式;  
end
```

例 :

```
for iFile = 1 : nFiles  
    for jPosition = 1 : nPositions  
        ...  
    end  
end
```

⑩ while 循环 (while-loop)

```
while 条件式  
    表达式;  
end
```

例 :

```
while iter < MAX_ITER  
    ...  
end
```


循环结构 (2)

49

● M文件

The image displays the MATLAB R2012a software interface. The 'File' menu is open, and the 'Script' option is highlighted with a red box, with the keyboard shortcut 'Ctrl+N' visible. The background shows the MATLAB workspace with a file explorer on the left listing files like 'worker.bat', 'ProductRoots', and 'mw_mpiexec.bat'. The central editor window shows a script with MATLAB code, and the bottom Command Window displays the prompt 'fx >>'.



循环结构 (3)

⑩ 例:

$$S = \sum_{i=1}^{100} i$$

Sum01.m:

```
s = 0;
for iCount = 1:100
    s = s + iCount;
end

>> s

s =

    5050
```

$$\min_i \left(\sum_{i=1}^{100} i > 1000 \right)$$

sum02.m:

```
s = 0;
iCount = 1;
while(s < 1000)
    s = s + iCount;
    iCount = iCount + 1;
end

[s,iCount]
>> sum2
ans =

    1035    46
```



⑩ 例：a, b, n为常数，x为向量，求解：
$$g_n = \prod_{j=1}^n \frac{a - x_j}{b - x_j}$$

Loop01.m

```
n = 100;  
x = rand(1,n);  
a = 0;  
b = 1;
```

```
g = 1;
```

```
for j = 1:n  
    g = g*(a - x(1,j))/(b-x(1,j));  
end
```

g =

4.8586e+06



循环结构（5）

⑩ 例：a, n为常数, x为向量, 求解：

$$g_n = \sum_{k=1}^n \left(\prod_{\substack{j=1 \\ j \neq k}}^n \frac{a - x_j}{x_k - x_j} \right)$$

```
% Loop02.m
```

```
n = 100;  
x = rand(1,n);  
a = 1;
```

```
g = 0;  
for k=1:n  
    h = 1;           % 累乘和  
    for j = 1:n  
        if j ~= k  
            h = h*(a - x(1,j))/(x(1,k)-x(1,j));  
        end  
    end  
    g = g + h;  
end
```



循环结构 (6)

⑩ 例: a, n 为常数, x, y 为向量, 求解:

$$f = (a - x_1) \cdots (a - x_n) \sum_{i=1}^n \frac{y_i}{\prod_{j=1, j \neq i}^n (x_i - x_j)}$$

```
% Loop03.m
```

```
n = 100;  
a = 1;  
x = rand(1,n);  
y = x.^2;
```

```
linearComb = 1;  
sum = 0;
```

```
for iOrder = 1:n  
    denom = 1;  
    for jOrder = 1:n  
        if iOrder ~= jOrder  
            denom = denom*(x(1,iOrder) - x(1,jOrder));  
        end  
    end  
    sum = sum + y(1,iOrder)/denom;  
    linearComb = linearComb*(a - x(1,iOrder));  
end  
f = linearComb * sum;
```



⑩ If - else – end

```
if 条件式  
    表达式;  
else  
    表达式;  
end
```

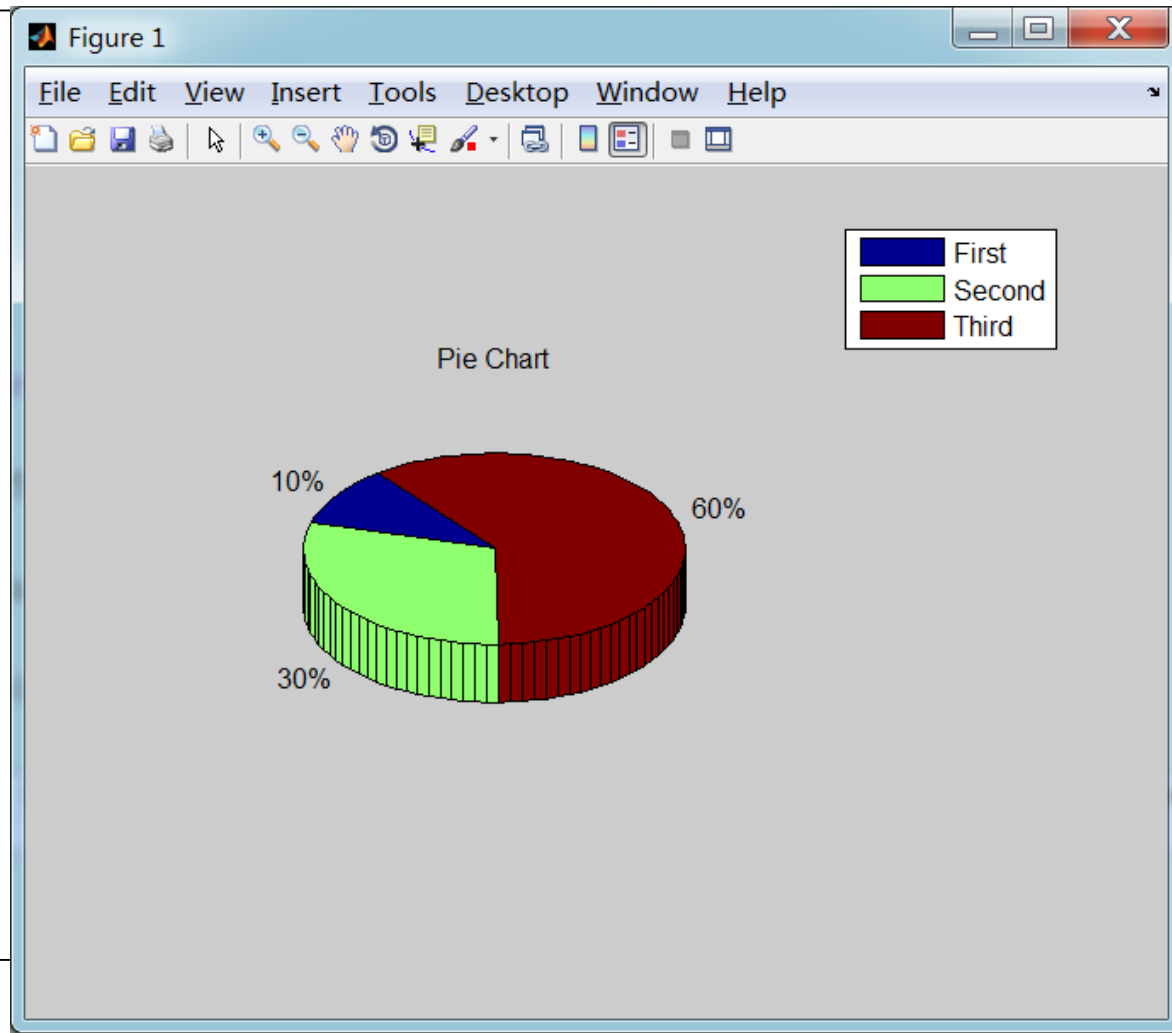
⑩ If - elseif - else - end

```
if 条件式  
    表达式;  
elseif 条件式  
    表达式;  
else  
    表达式;  
end
```

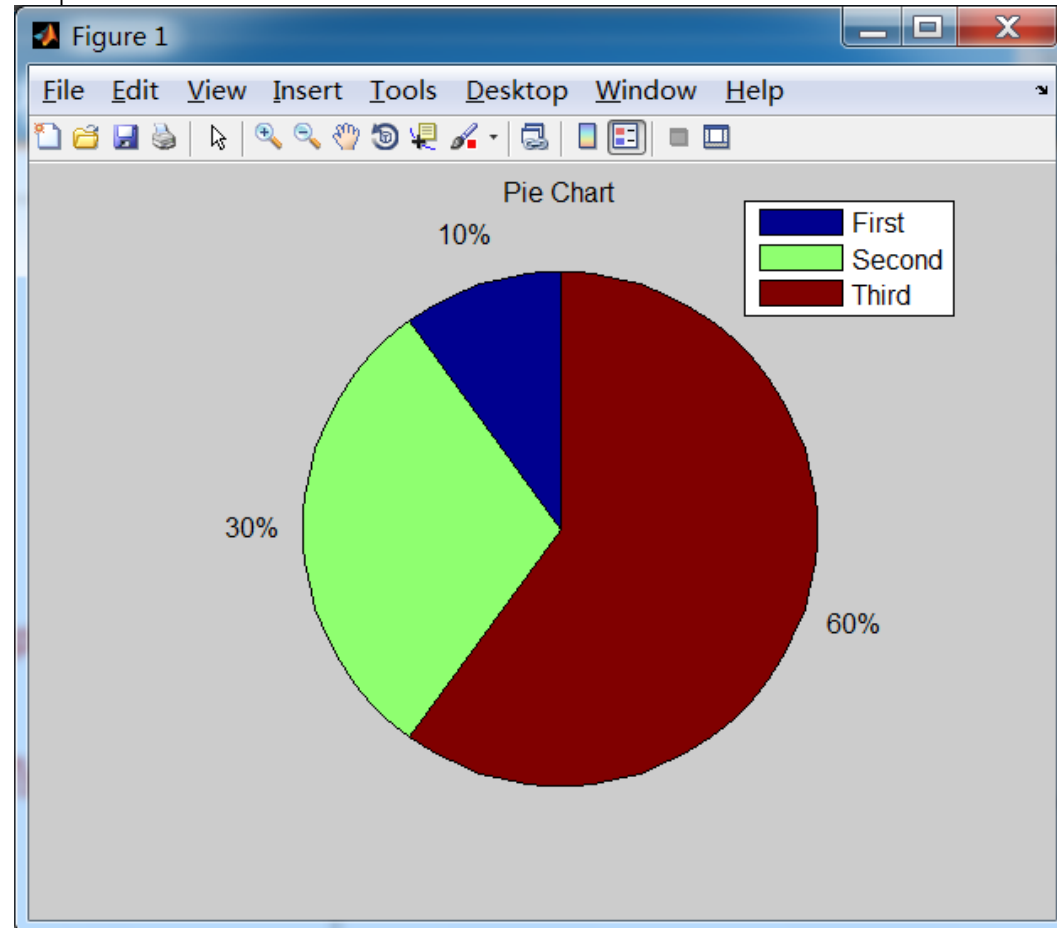
⑩ 开关结构形式如下：

```
switch 开关表达式  
    case 条件表达式1  
        表达式1;  
    case 条件表达式2  
        表达式2;  
    .....  
    otherwise  
        表达式n;  
end
```

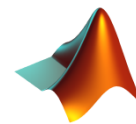
```
x = [10, 30, 60];  
plotType = 'pie';  
innerType = 2;  
switch plotType  
    case 'bar'  
        bar(x);  
        title('Bar Graph');  
    case 'pie'  
        switch innerType  
            case 1  
                pie(x);
```




```
x = [10, 30, 60];  
plotType = 'pie';  
innerType = 1;  
switch plotType  
    case 'bar'  
        bar(x);  
        title('Bar Graph');  
    case 'pie'  
        switch innerType  
            case 1  
                pie(x);
```



- ⑩ MATLAB 简介
- ⑩ MATLAB 编码规则和规范
- ⑩ MATLAB 数学运算
- ⑩ MATLAB 程序流程控制
- ⑩ MATLAB 函数定义及程序调试
- ⑩ MATLAB 二维绘图



MATLAB 函数定义及程序调试



```
function [函数返回变量列表] = 函数名(输入变量列表)
% 注释说明段
函数体
end % M文件中仅有一个函数时可省略
```

⑩ 隐藏变量：nargin, nargout

分别表示输入和返回变量的实际个数，此为MATLAB保留变量，只要进入该函数，MATLAB就将自动生成这两个变量。

例:

```
function [mean,stdev] = Stat(x)
```

```
% Calculate the mean and standard deviation of a vector x
```

```
% input:
```

```
% x - the vector
```

```
% output:
```

```
% mean - the mean of x
```

```
% stdev - the standard deviation of x
```

```
n = length(x);
```

```
mean = sum(x)/n;
```

```
stdev = sqrt(sum((x-mean).^2/n));
```

```
>> x = randn(1,100);  
>> [mean,stdev] = Stat(x)  
mean =  
    -0.0727  
  
stdev =  
    1.0000
```



- ⑩ MATLAB是一种解释和执行同时进行的语言，这使得程序的调试变得相对便利，尤其是MATLAB具有良好的所见即所得特性。
- ⑩ MATLAB还提供了专门的调试器，即M文件编译器，用户能够完成大部分的程序调试工作。

调试的基本任务

- ⑩ 程序调试（Debug）的基本任务就是要找到并去除程序中的错误。程序的错误大致可以分为如下三类：
- 语法错误：由于程序员疏忽、输入不正确等原因而造成的代码违背程序语言规则的错误。
 - 运行错误：由于对所求解问题的理解差异，导致程序流程出错或对程序本身的特性认识有误而造成的程序执行结果错误的情况。
 - 程序异常：程序执行过程中由于不满足条件而造成的程序执行错误。

Error1

错误：文件:Error1.m 行:4 列:8

不应为 MATLAB 表达式。

例：

```
% Error1.m
```

```
% 定义矩阵A
```

```
A = (1 2 3,4 5 6,7 8 9);
```

```
% 定义矩阵B
```

```
B = (1 2 3 4,5 6 7 8,9 10 11 12,13 14 15 16);
```

```
% C为矩阵A和B相乘
```

```
C = A*B;
```


例：

% 求解方程： $A*x = B$

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

$B = [9 \ 8 \ 7; 6 \ 5 \ 4; 3 \ 2 \ 1];$

$x = B/A;$

>>

$x =$

-2.6667 0 1.6667

-2.1667 0 1.1667

-1.6667 0 0.6667

>> $A*x - B$

ans =

-21.0000 -8.0000 -1.0000

-37.5000 -5.0000 12.5000

-54.0000 -2.0000 26.0000

运行错误 (2)

例:

% 求解方程: $A*x = B$

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

$B = [9 \ 8 \ 7; 6 \ 5 \ 4; 3 \ 2 \ 1];$

$x = A \setminus B;$

>>

$x =$

-27 -26 -17

42 41 24

-16 -16 -8

>> $A*x - B$

ans =

0 0 0

0 0 0

0 0 0

例:

```
function [mean,stdev] = Stat(x)
```

```
% Calculate the mean and standard deviation of a vector x
```

```
% input:
```

```
% x - the vector
```

```
% output:
```

```
% mean - the mean of x
```

```
% stdev - the standard deviation of x
```

```
n = length(x);
```

```
mean = sum(x)/n;
```

```
stdev = sqrt(sum((x-mean).^2/n));
```

```
>> x = randn(3,3);
```

```
>> [mean,stdev] = Stat(x)
```

Error using -

Matrix dimensions must agree.

Error in Stat (line 11)

```
stdev = sqrt(sum((x-mean).^2/n));
```

```
function LeapYear(startYear,endYear)
```

```
% Output the leap years between start year
```

```
% and end year.
```

```
% input:
```

```
% startYear - the start year
```

```
% endYear - the end year
```

```
for year = startYear:endYear
```

```
    sign = 0; %定义标志位
```

```
    a = rem(year,100); %求year除以100后的剩余数
```

```
    b = rem(year,4); %求year除以4后的剩余数
```

```
    c = rem(year,400); %求year除以400后的剩余数
```

```
%以下根据a、b、c是否为0对标志变量sign进行处理
```

```
>> LeapYear(2000,2015)
```

错误：文件:LeapYear.m 行:14
列:10

等号左侧的表达式不是用于赋值
的有效目标。

```
if a = 0
```

```
    sign = sign + 1;
```

```
end
```

```
if b = 0
```

```
    sign = sign + 1;
```

```
end
```

```
if c = 0
```

```
    sign = sign - 1;
```

```
end
```

```
if sign = 1
```

```
    fprintf('%4d \n',year);
```

```
end
```

```
end
```



程序调试 (2)

69

```
function LeapYear(startYear,endYear)
```

```
% Output the leap years between start year
```

```
% and end year.
```

```
% input:
```

```
% startYear - the start year
```

```
% endYear - the end year
```

```
for year = startYear:endYear
```

```
    sign = 0;
```

%定义标志位

```
    a = rem(year,100);
```

%求year除以100后的剩余数

```
    b = rem(year,4);
```

%求year除以4后的剩余数

```
    c = rem(year,400);
```

%求year除以400后的剩余数

```
%以下根据a、b、c是否为0对标志变量sign进行处理
```

```
>> LeapYear(2007,2015)
```

```
2008
```

```
2012
```

```
    if a == 0
```

```
        sign = sign + 1;
```

```
    end
```

```
    if b == 0
```

```
        sign = sign + 1;
```

```
    end
```

```
    if c == 0
```

```
        sign = sign - 1;
```

```
    end
```

```
    if sign == 1
```

```
        fprintf('%4d \n',year);
```

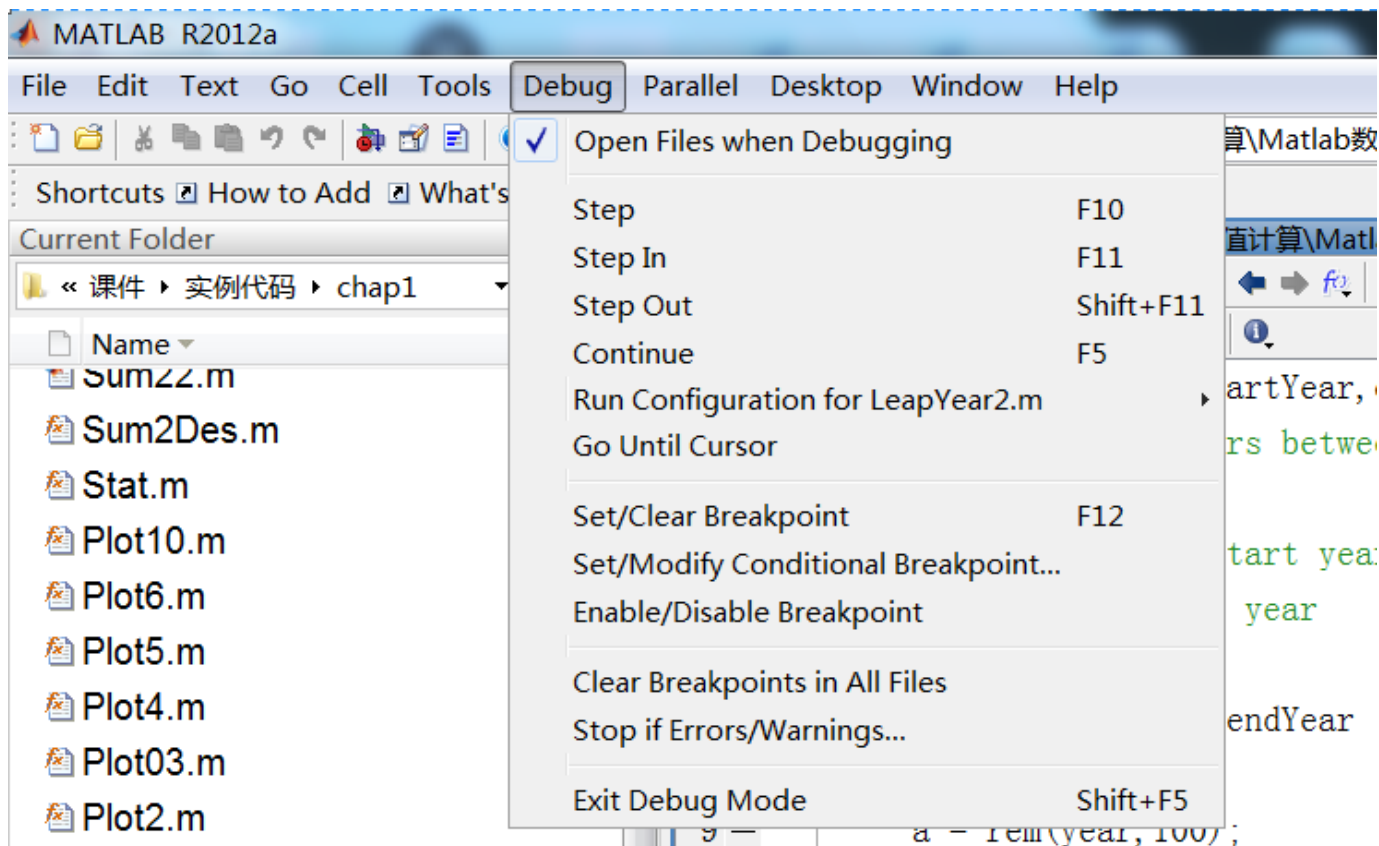
```
    end
```

```
end
```



程序调试 (3)

70



- ⑩ step: 单步执行，不进入函数
- ⑩ step in: 进入子函数单步执行
- ⑩ step out: 在函数中的话跳出函数，否则直接跳入下个断点处
- ⑩ save and run: 存储，运行
- ⑩ go until cursor: 运行到光标处

程序调试 (5)

72

Editor - E:\我的工作\教学工作\matlab数值计算\Matlab数值计算方法与实践-冯辉\课件\实例代码\chap1\LeapYear2.m

```
7 for year = startYear:endYear
8     sign = 1; %定义标志位
9     a = rem(year, 100); %求year除以100后的剩余数
10    b = rem(year, 4); %求year除以4后的剩余数
11    c = rem(year, 400); %求year除以400后的剩余数
12
13    %以下根据a、b、c是否为0对标志变量sign进行处理
14    if a == 0
15        sign = sign + 1;
16    end
17    if b == 0
18        sign = sign + 1;
19    end
20    if c == 0
21        sign = sign - 1;
22    end
23    if sign == 1
24        fprintf('%4d \n', year);
25    end
26 end
```

Workspace

Name	Value	Min	Max
a	7	7	7
b	3	3	3
c	7	7	7
endYear	2015	20...	20...
sign	1	1	1
startYear	2007	20...	20...
year	2007	20...	20...

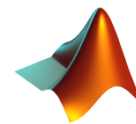
Command History

```
%-- 2015/3/27 10:00 --%
x = randn(3,3);
[mean,stdev] = stat(x)
[mean,stdev] = Stat(x)
x-mean
LeapYear(2000,2015)
LeapYear2(2000,2015)
LeapYear3(2000,2015)
LeapYear2
clear
clc
LeapYear2
clc
LeapYear2(2007,2015)
```

Command Window

```
>> LeapYear2(2007,2015)
7 for year = startYear:endYear
fx K>>
```


- ⑩ MATLAB 简介
- ⑩ MATLAB 编码规则和规范
- ⑩ MATLAB 数学运算
- ⑩ MATLAB 程序流程控制
- ⑩ MATLAB 函数定义及程序调试
- ⑩ MATLAB 二维绘图



MATLAB二维绘图



⑩ 强大的绘图功能是Matlab的特点之一

- Matlab提供了一系列的绘图函数，用户不需要过多的考虑绘图的细节，只需要给出一些基本参数就能得到所需图形，这类函数称为高层绘图函数。
- Matlab还提供了直接对图形句柄进行操作的低层绘图操作。这类操作将图形的每个图形元素（如坐标轴、曲线、文字等）看做一个独立的对象，系统给每个对象分配一个句柄，可以通过句柄对该图形元素进行操作，而不影响其他部分。

⑩ 基本二维绘图函数

指令	说明
plot	x 轴和 y 轴均为线性刻度(Linear Scale)
loglog	x 轴和 y 轴均为对数刻度(Logarithmic Scale)
semilogx	x 轴为对数刻度, y 轴为线性刻度
semilogy	x 轴为线性刻度, y 轴为对数刻度
plotyy	画出两个刻度不同的y轴



10 高级二维绘图函数

指令	说明
errorbar	在曲线加上误差范围
fplot、ezplot	绘制自定义函数和隐函数的图形
polar、ezpolar	极坐标图形
hist	直角坐标直方图(累计图)
rose	极坐标直方图(累计图)
compass	罗盘图
feather	羽毛图
area	面积图
stairs	阶梯图



10 plot函数

- `plot(Y)`
- `plot(X1,Y1,...,Xn,Yn)`
- `plot(X1,Y1,LineSpec,...,Xn,Yn,LineSpec)`
- `plot(...,'PropertyName',PropertyValue,...)`

⑩ 对向量绘图

例:

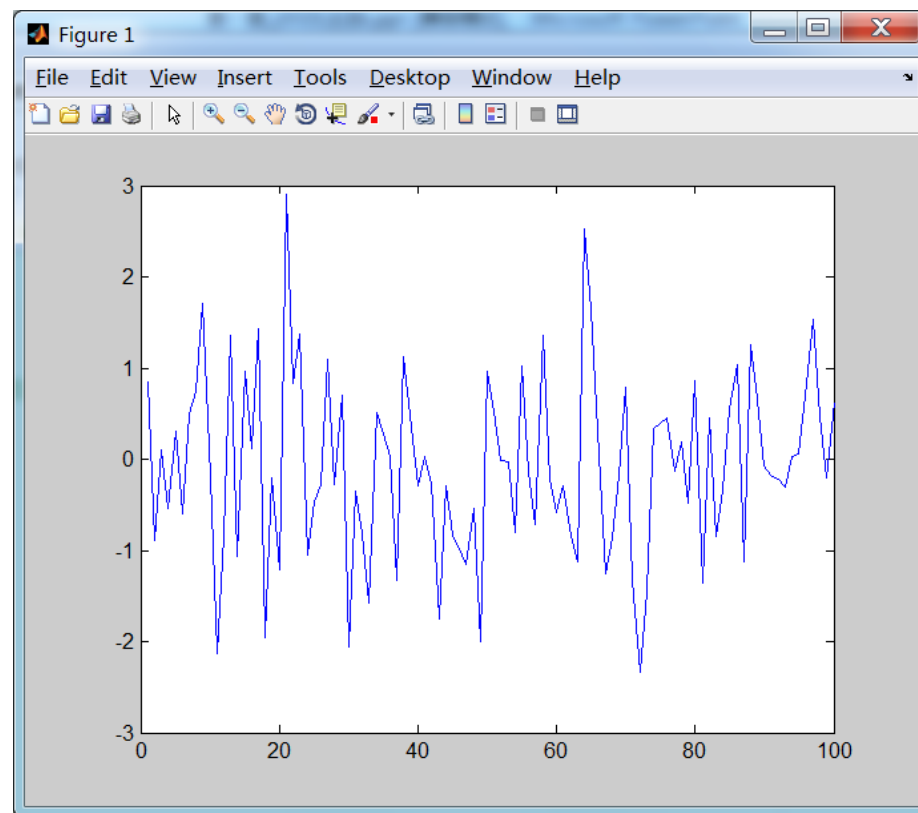
```
function Plot1
```

```
close all;
```

```
% 生成服从正态分布的序列
```

```
x = randn(1,100);
```

```
plot(x);
```



⑩ 对矩阵绘图

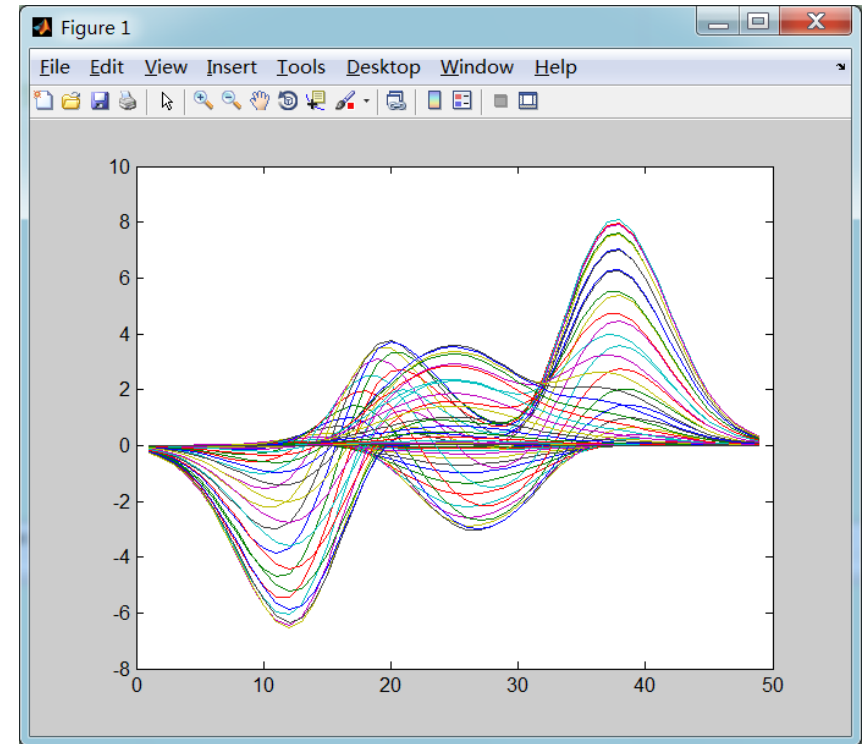
例：

% 产生一个 49×49 的矩阵

`x = peaks;`

% 对矩阵 `x` 的每一个行向量作图

`plot(x);`



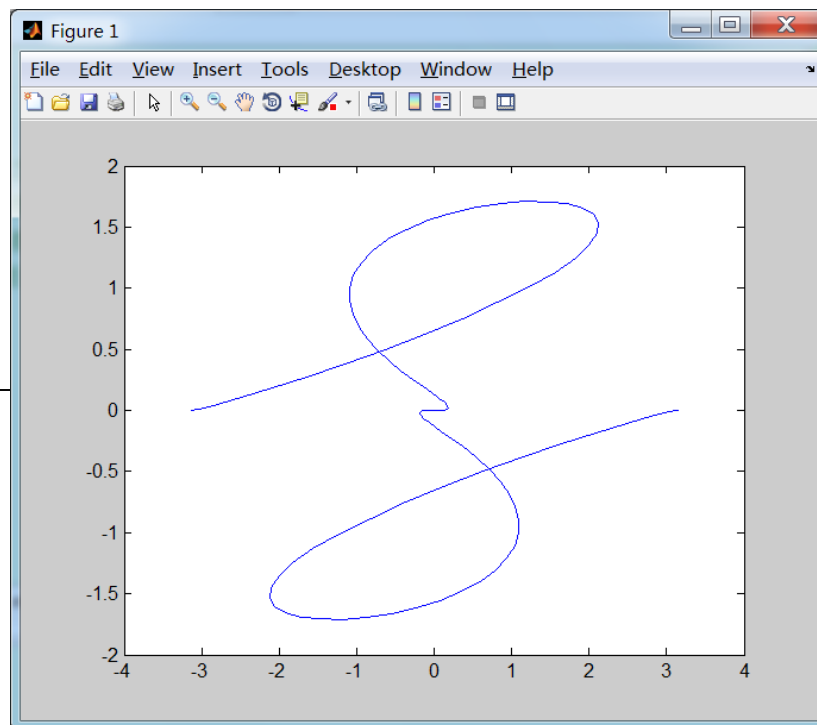
plot函数 (2)

例:

```
function Plot2  
close all;
```

```
t = linspace(-pi,pi);  
x = t.*cos(3*t);  
y = t.*sin(t).*sin(t);  
plot(x,y)
```

```
% 生成-n到n之间的线性序列，默认100个等分点  
% 计算 x 的函数值  
% 计算 y 的函数值  
% 进行二维平面描点作图
```



plot函数 (2)

82

例:

```
function Plot3
```

```
close all;
```

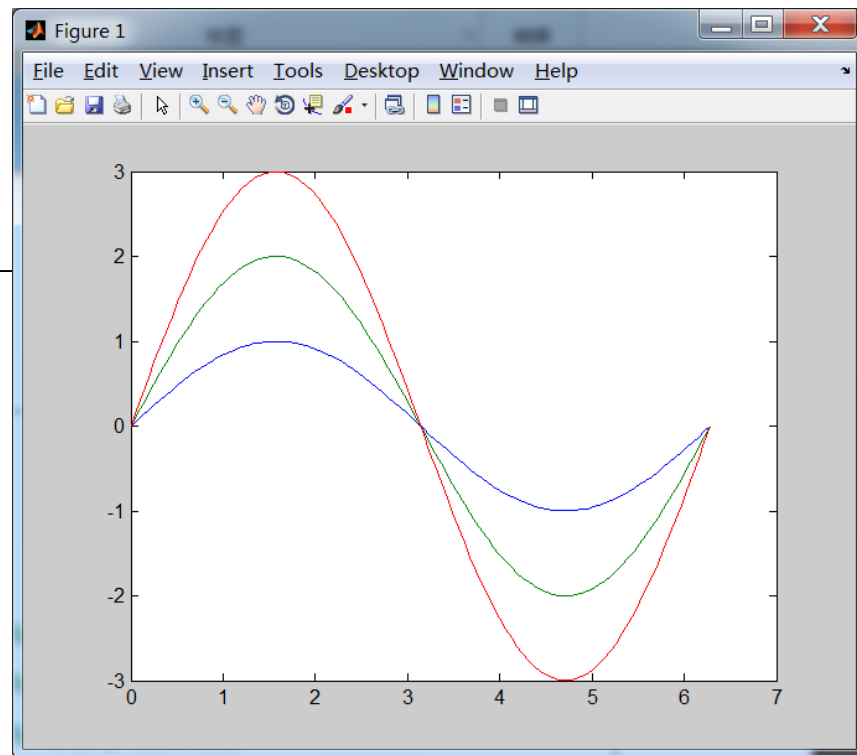
```
x = linspace(0,2*pi,100);
```

```
y1 = sin(x);           % 计算 y1 的函数值
```

```
y2 = 2*sin(x);         % 计算 y2的函数值
```

```
y3 = 3*sin(x);         % 计算 y3 的函数值
```

```
plot(x,y1,x,y2,x,y3);  % 进行二维平面描点作图
```



plot函数 (2)

83

例:

```
function Plot3
```

```
close all;
```

```
x = linspace(0,2*pi,100);
```

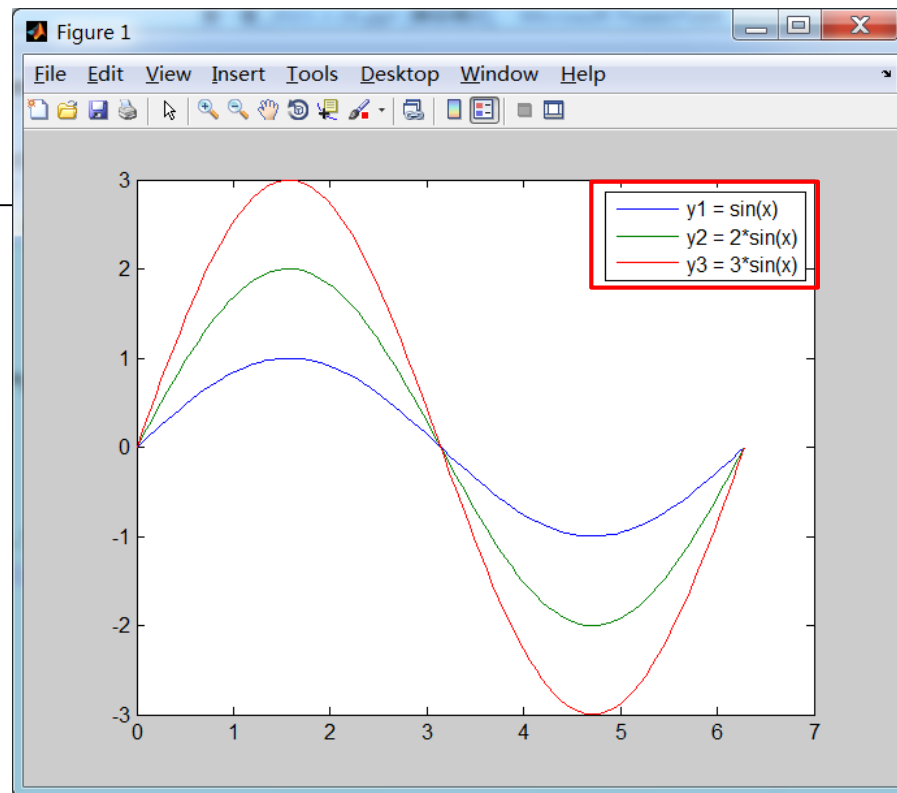
```
y1 = sin(x);           % 计算 y1 的函数值
```

```
y2 = 2*sin(x);        % 计算 y2的函数值
```

```
y3 = 3*sin(x);        % 计算 y3 的函数值
```

```
plot(x,y1,x,y2,x,y3); % 进行二维平面描点作图
```

```
legend('y1 = sin(x)','y2 = 2*sin(x)','y3 = 3*sin(x)');
```



颜色	线型	标记符号	
b 蓝色	- 实线	. 点	h 六角星
g 绿色	: 虚线	o 圆圈	v 下三角
r 红色	-. 点划线	x 叉叉	^ 上三角
c 青色	-- 双划线	+ 加号	< 左三角
m 品红		* 星号	> 右三角
y 黄色		s 方块	
k 黑色		d 菱形	
w 白色		p 五角星	

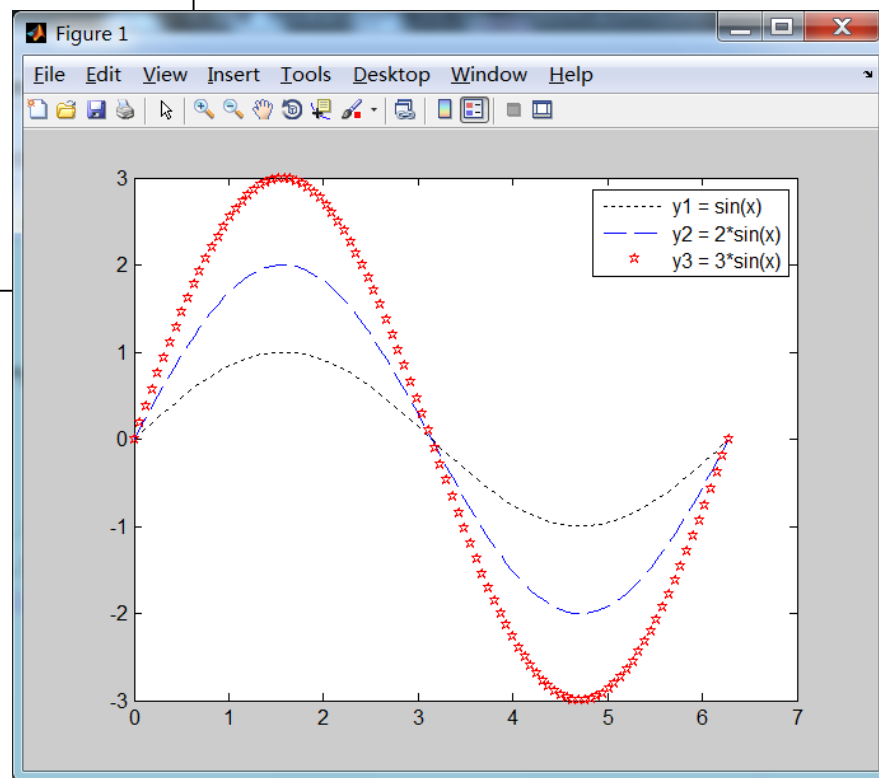
plot函数 (3)

85

例:

```
% plot(x,y1,x,y2,x,y3); % 进行二维平面描点作图
```

```
plot(x,y1,'k:',...  
      x,y2,'b--',...  
      x,y3,'rp');
```



semilogx函数

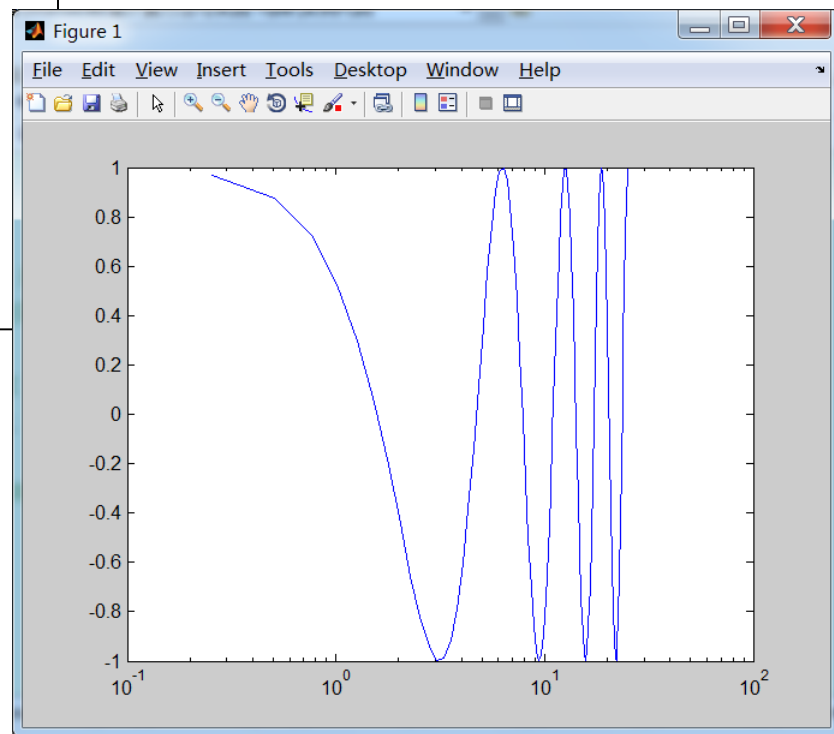
⑩ x 轴为对数刻度, y 轴为线性刻度

% 在 0 到 8 间, 等分取 100 个点

```
x = linspace(0, 8*pi);
```

% 使 x 轴为对数刻度, 并对其正弦函数作图

```
semilogx(x, cos(x));
```



plotyy函数

87

```
function Plot4
```

```
close all;
```

```
x = 0:0.01:20;
```

```
y1 = 200*exp(-0.05*x).*sin(x);
```

```
y2 = 0.8*exp(-0.5*x).*sin(10*x);
```

```
% 绘制刻度不同的纵轴
```

```
[AX,H1,H2] = plotyy(x,y1,x,y2,'plot');
```

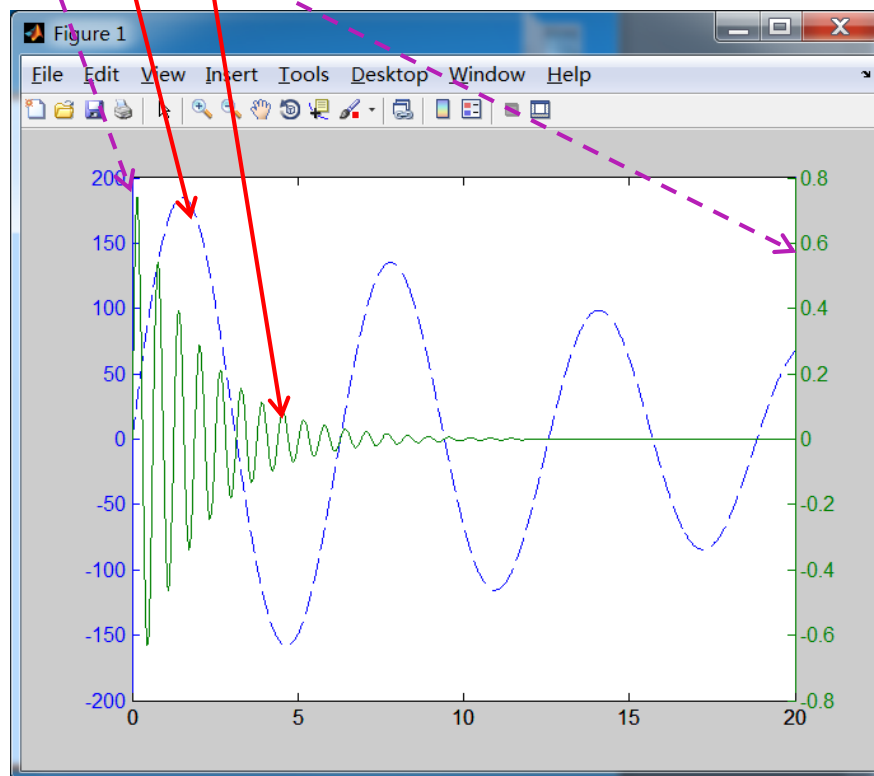
```
% 设置线型
```

```
set(H1,'LineStyle','--');
```

```
set(H2,'LineStyle','-');
```

[AX,H1,H2] = plotyy(X1,Y1,X2,Y2,function)

基本二维绘图函数



subplot函数

88

例:

function Plot5

close all;

x = 0:0.35:7;

y = 2*exp(-0.5*x);

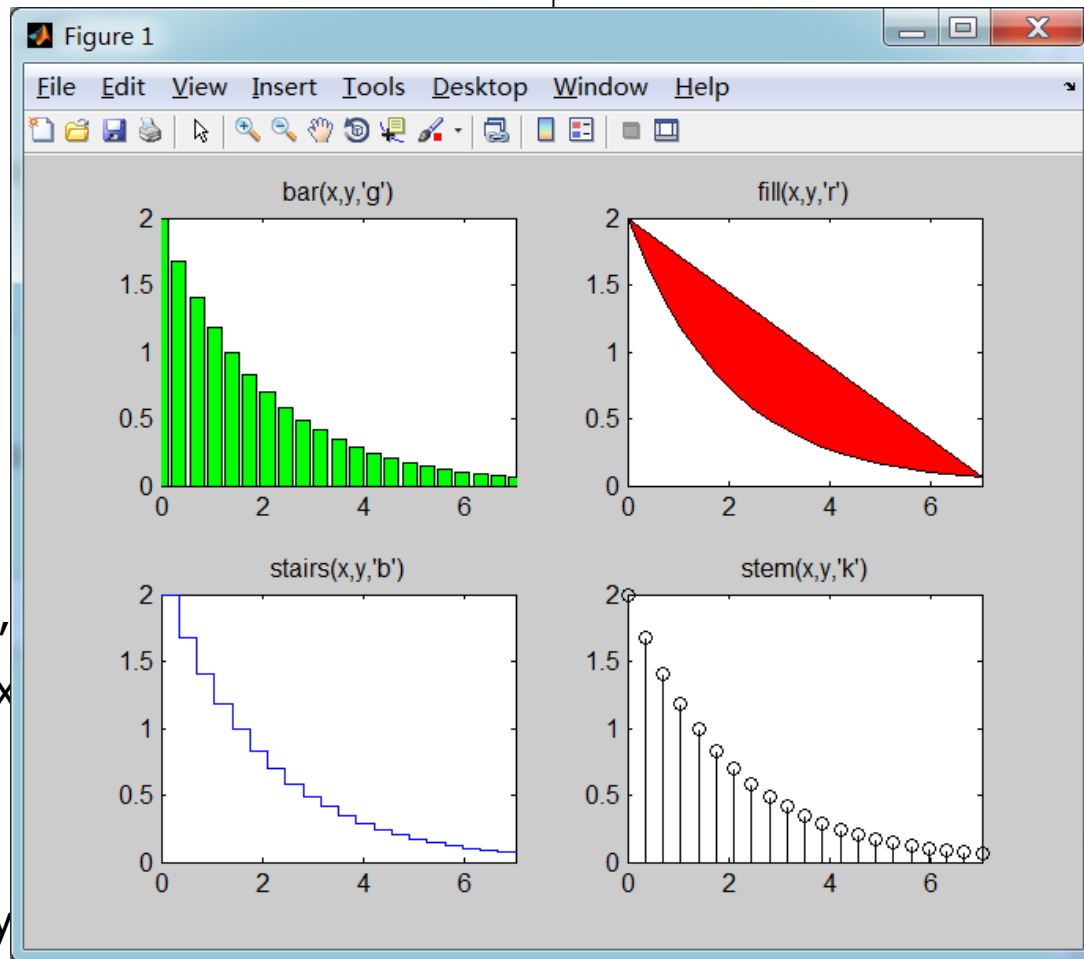
% 绘制第一组图形

subplot(2,2,1); bar(x,
title('bar(x,y,"g")'); ax

% 绘制第二组图形

subplot(2,2,2); fill(x,y

title('fill(x,y,"r")'); axis([0, 7, 0 ,2]);



坐标轴设置

```
function Plot6
```

```
x = 0:0.01:20;
```

```
y1 = 200*exp(-0.05*x).*sin(x);
```

```
plot(x,y1);
```

```
% 坐标轴上下限设置
```

```
axis([-inf, inf, -150, 150]);
```

```
% 坐标轴（y轴）刻度设置
```

```
set(gca, 'ytick', [-100 -50 50 100]);
```

```
% 改变坐标轴刻度的文字
```

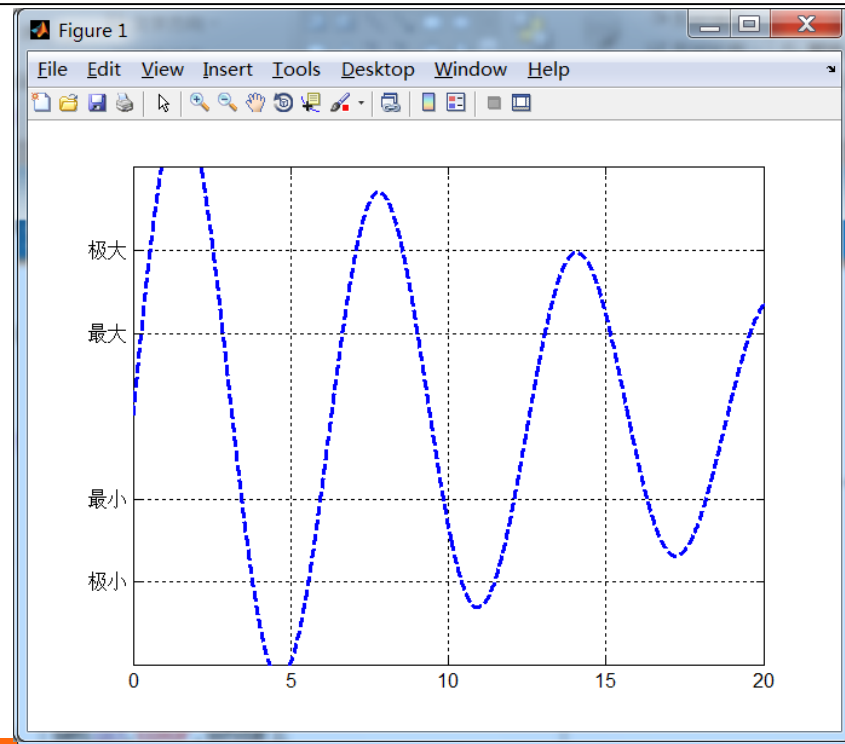
```
set(gca, 'yticklabel', {'极小','最小','最大','极大'});
```

```
% 改变画布背景色
```

```
set(gcf,'color','white');
```

```
% 开启网格
```

```
grid on;
```



网格及边框设置

- ⑩ grid: 绘制网格开关 (on/off)
- ⑩ box: 绘制外边框开关 (on/off)

指令	说明
grid on	画出网格线
grid off	取消网格线
box on	画出图轴的外围长方形
box off	取消图轴的外围长方形

⑩ 在图形或图轴加入说明文字

指令	说明
title	图形的标题
xlabel	x 轴的说明
ylabel	y 轴的说明
zlabel	z 轴的说明(适用于立体绘图)
legend	多条曲线的说明
text	在图形中加入文字
gtext	使用鼠标决定文字的位置

说明文字 (2)

92

% 设置图形标题、坐标轴标题、
明文字

```
title('Multiple Decay Rates');
```

```
xlabel('Time (\musec)');
```

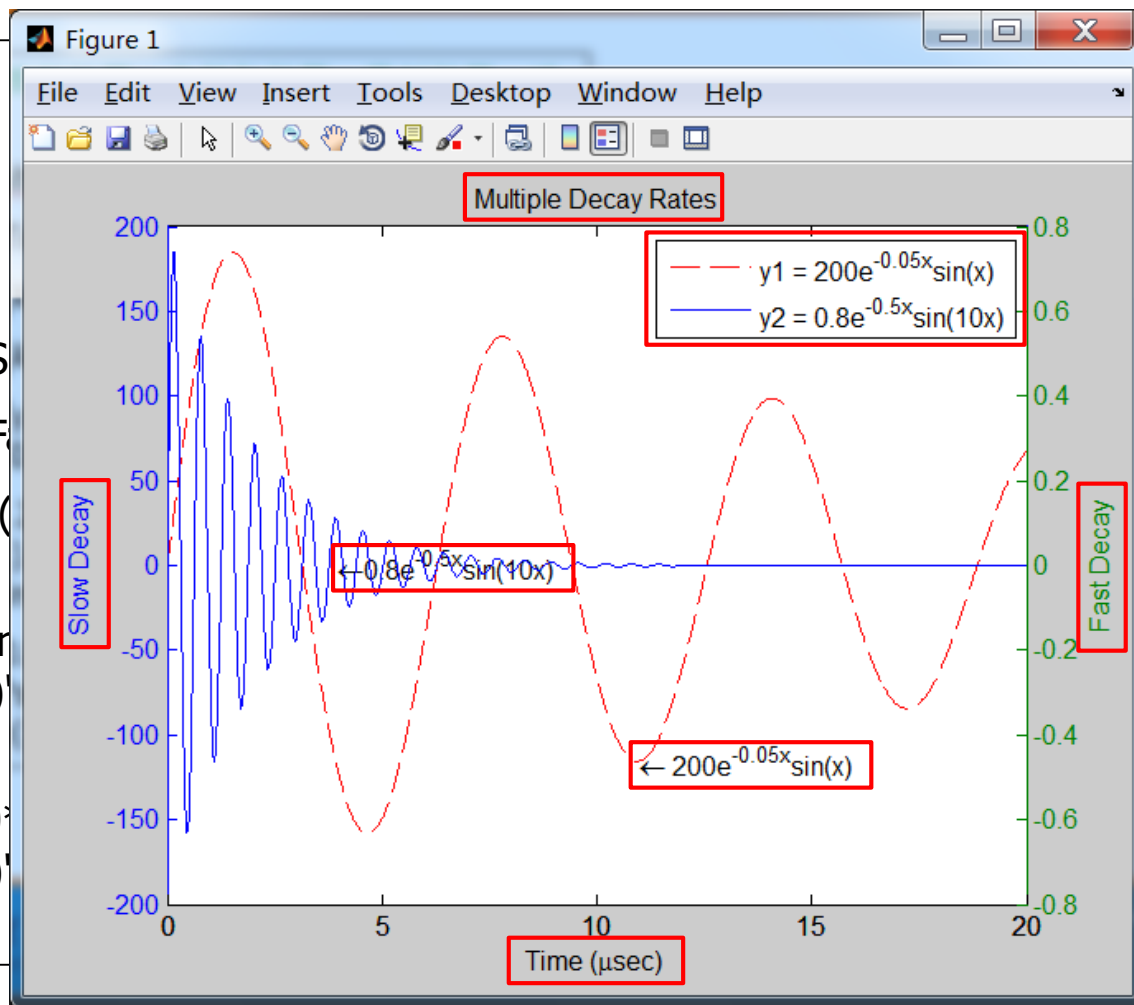
```
set(get(AX(1),'Ylabel'),'String','S
```

```
set(get(AX(2),'Ylabel'),'String','F
```

```
legend('y1 = 200e^{-0.05x}sin(x)  
0.8e^{-0.5x}sin(10x)');
```

```
text(11, 200*exp(-0.05*11).*sin  
'\leftarrow 200e^{-0.05x}sin(x)  
'HorizontalAlignment', 'left');
```

```
text(9, 0.8*exp(-0.5*9).*sin(10  
'\leftarrow 0.8e^{-0.5x}sin(10x)  
'HorizontalAlignment', 'right');
```



- ⑩ 安装Matlab 2014a及以上版本
- ⑩ 把第一章课堂上讲解的**所有**代码，**亲自**输入到Matlab应用程序并运行
- ⑩ 参考Matlab内置文档，学会基本**数据运算、程序流控制、函数和高级二维绘图函数**的使用，自导数据，把图形绘制结果导出到word文档（以“**第X次 姓名 班级 学号.doc**”命名，见群中模板），汇总至**组长**，打包发到老师邮箱（截止**下周二3.13晚12:00**前）

Nerver Copy
& Paste!





武汉理工大学
WUHAN UNIVERSITY OF TECHNOLOGY

Q & A

