

MATLAB 小结

第一讲、Matlab 矩阵运算	1
1.1 Matlab 基础	1
1.2 矩阵运算	3
1.3 常用数学函数	5
第二讲、Matlab 绘图	8
2.1 平面曲线	8
2.2 空间曲线	9
2.3 空间曲面	9
2.4 符号绘图	9
第三讲、符号运算	11
3.1 符号运算基础	11
3.2 常见符号运算	11
第四讲、编程基础	12
4.1 M 文件	12
4.2 关系运算与逻辑运算	12
4.3 输入输出	12
4.4 选择与循环	13
4.2 函数文件	13

第零讲、MATLAB 介绍

MATLAB 是 MATrix LABoratory 的缩写，是由美国 MathWorks 公司出品的商业数学软件。MATLAB 是一种高级计算语言和交互式环境，并带有为数众多的工具箱 (Toolbox)，分别适合不同领域的应用。除了数值计算、符号计算和系统仿真等主要功能外，MATLAB 还可以与其它语言 (如 C, C++, FORTRAN 等) 进行混合编程。

1970 年代末到 80 年代初，时任美国新墨西哥大学教授的 Cleve Moler (克里夫·莫勒尔) 为了让学生更方便地使用 LINPACK 及 EISPACK (需要通过 FORTRAN 编程来实现，但当时学生们并无相关知识)，利用业余时间编了写 EISPACK 和 LINPACK 的接口程序，并取名为 MATLAB，即矩阵(matrix)和实验室(labotatory)两个英文单词的前三个字母的组合。这个版本的 MATLAB 只能进行简单的矩阵运算，例如矩阵转置、计算行列式和特征值等。在之后的数年里，MATLAB 在多所大学里作为教学辅助软件使用，并作为面向大众的免费软件广为流传。

1983 年春天，Cleve Moler 到 Standford 大学讲学。工程师 Jack Little 对 MATLAB 产生了很大的兴趣。他敏锐地觉察到了 MATLAB 在工程领域的广阔应用前景。同年，他和 Cleve Moler, Steve Bangert 一起，用 C 语言开发了第二代 MATLAB，并于 1984 年合作成立了 MathWorks 公司，正式把 MATLAB 推向市场。

MATLAB 最初是由 Moler 用 FORTRAN 编写的，Little 和 Bangert 花了约一年半的时间用 C 重新编写了 MATLAB，并增加了一些新功能，同时，Little 还开发了第一个系统控制工具箱，其中一些代码到现在仍然在使用。MATLAB 1.0 在拉斯维加斯举行的 IEEE 决策与控制会议 (IEEE Conference on Decision and Control) 上正式推出，它的第一份订单只售出了 10 份拷贝。而根据 MathWorks 自己的数据，目前 (2014 年) 全球有超过 5000 家的大专院校采用 MathWorks 的解决方案来开展多种技术学科的教学和研究工作。

第一讲、Matlab 矩阵运算

1.1 Matlab 基础

- Matlab 语句的一般形式:

变量 = 表达式	计算表达式的值，并将结果赋给赋值号“=”左边的变量
-----------------	---------------------------

- Matlab 命令的执行: **回车**
- Matlab 所有命令和内置函数都是由小写字母构成
- Matlab 变量的命名规则:
 - ① **以字母开头**;
 - ② 由字母、数字和下划线组成;
 - ③ 字母**区分大小写**;
 - ④ 长度不要太长，一般不要超过 63 个字符
- 常用功能键

Enter(回车)	运行命令或语句
Ctrl+C	强制中止运行中的语句或程序
Esc	删除当前行的所有内容
Tab	命令补全功能
↑	调出前面一个输入的命令或语句
Ctrl+K	删除从光标处到行尾的内容
Shift+Enter	换行 (不运行输入的命令或语句)

● 系统预定义的变量:

pi	圆周率
inf, Inf	无穷大
i, j	虚部单位
eps	双精度浮点运算相对误差 (机器精度), 大约为 2.2204×10^{-16}
nan, NaN	Not-a-number, 不定值

● 特殊变量: **ans**

● Matlab 中的数:

- ① 默认的是双精度实数;
- ② 可用浮点形式或科学计数法表示;
- ③ 所能表示的实数的范围为: $10^{-308} \sim 10^{308}$;
- ④ 浮点运算的相对误差为 **eps**, 大约为 2.2×10^{-16} ;
- ⑤ 复数的输入: **z=3+4i**, 注意虚部与 i 之间不能有空格

● 基本运算及优先级

运算符		优先级
()	小括号	高
', .'	转置, 共轭转置	↓
^, .^	幂, 数组幂运算	↓
+, -, ~	正号, 负号, 逻辑非	↓
, /, \, . , ./, .\	乘除, 数组乘除	↓
+, -	加减	↓
:	冒号运算	↓
>, >=, <, <=, ==, ~=	关系运算	↓
&	逻辑与	↓
 	逻辑或	↓
=	赋值	低

● 特殊符号

:	冒号, 等差序列	()	小括号/圆括号
[]	方括号, 矩阵输入	{ }	花括号, cell
@	函数句柄	...	续行号
,	逗号, 命令分隔符	=	赋值号
;	分号, 命令分隔符	%	注释符
'	单引号, 字符串	!	调用操作系统命令

● 命令分隔符: 逗号或分号

● 若不想在屏幕上显示结果, 可用分号作为语句结束符

● 系统操作命令

cd	改变当前工作目录	dir	列出当前目录中的文件
pwd	显示当前工作目录	mkdir	创建子目录
.	当前工作目录	..	父目录
computer	显示计算机类型	diary	储存命令窗口内容

● 工作空间和搜索路径管理

clear	清除变量和函数	pack	合并工作内存中的碎块
who	列出变量	edit	打开或编辑 M 文件

whos	列出变量细节	type	显示文件内容
what	列出当前目录中的 Matlab 文件	path	显示 Matlab 搜索路径
which	显示指定函数的位置	addpath	添加搜索路径
exit	退出 MATLAB	pathtool	修改搜索路径

- Matlab 中所有标点符号必须在英文状态下输入!

1.2 矩阵运算

- Matlab: 矩阵实验室, 以矩阵为基本操作单位, 在 Matlab 中, 一切皆为矩阵
- 矩阵的直接输入: 如 **A=[1 2 3; 4,5,6]**
 - ① 用方括号 “[]”;
 - ② 同一行中元素之间用空格或逗号分隔;
 - ③ 行与行之间用分号隔开
- 冒号的特殊作用: **a:b:c %** 生成一个满足等差数列的向量
 - ① **a** 是首项, **b** 是公差, **c** 确定最后一项 (不一定就是最后一项);
 - ② 若公差为 1, 则可简写为 **a:c**
- 矩阵元素的引用

x(i)	向量 x 中的第 i 个元素
x(i:j)	向量 x 中的第 i 到第 j 个元素
x(i:end)	向量 x 中的第 i 个到最后一个元素
x(:)	由 x 的所有元素组成的列向量
A(i,j)	矩阵 A 中的第 i 行, 第 j 列元素
A(i:j,m:n)	由第 i 至 j 行和第 m 至 n 列组成的子矩阵
A(i:end,m:n)	第 i 行到最后一行与 m 至 n 列组成的子矩阵
A(i:j,m:end)	第 m 列到最后一列与 i 至 j 行组成的子矩阵
A(:,k)	矩阵的第 k 列
A(i,:)	矩阵的第 i 行
A(i:j,:)	矩阵的第 i 行到第 j 行
A(:,m:n)	矩阵的第 m 行到第 n 列
A(:,:)	整个矩阵
A(:)	将矩阵的所有元素按列排成一个列向量
A([i1, i2, ..., ip],[j1, j2, .., jq])	第 i1,..., ip 行和第 j1,..., jq 列组成的子矩阵

- 删除指定的行或列

A(i,:)=[]	删除第 i 行
A(:,j)=[]	删除第 j 列
A(i:j,:)=[]	删除第 i 至第 j 行
A(:,i:j)=[]	删除第 i 至第 j 列
A(:,:)=[]	删除整个矩阵

- 常见的矩阵生成函数

zeros(m,n)	生成一个 m 行 n 列的零矩阵
zeros(n)	m=n 时可简写为 zeros(n)
ones(m,n)	生成一个 m 行 n 列的元素全为 1 的矩阵
ones(n)	m=n 时可简写为 ones(n)
eye(m,n)	生成一个主对角线全为 1 的 m 行 n 列矩阵

eye(n)	m=n 时可简写为 eye(n)，即为 n 维单位矩阵
rand(m,n) rand(n)	产生 0~1 间均匀分布的随机矩阵 m=n 时简写为 rand(n)
randn(m,n) randn(n)	产生均值为 0，方差为 1 的标准正态分布随机矩阵 m=n 时简写为 randn(n)
magic(n)	生成一个 n 阶魔方矩阵
hilb(n)	生成一个 n 阶 Hilbert 矩阵
invhilb(n)	n 阶 Hilbert 矩阵的逆矩阵
pascal(n)	生成一个 n 阶 pascal 矩阵
vander(x)	生成一个 Vandermonde 矩阵
wilkinson(n)	生成一个 n 阶 Wilkinson 矩阵（主要用于特征值测试）
diag(X)	若 X 是矩阵，则 diag(X) 为 X 的主对角线向量 若 X 是向量，diag(X) 产生以 X 为主对角线的对角矩阵
diag(X,k)	若 X 是矩阵，则为提取 X 的第 k 条对角线 若 X 是向量，则生成以 X 为第 k 条对角线的对角矩阵 k 是一个整数，可正可负，k=0 时即为 diag(X)
tril(A)	提取一个矩阵的下三角部分
triu(A)	提取一个矩阵的上三角部分
gallery	一些供测试用的小矩阵

- 矩阵基本运算
 - ① 加减运算：**A+B**，**A-B**，对应分量进行运算
 - ② 普通乘法：**A*B**
 - ③ 除法：**A\B** 等价于 $A^{-1}B$ ，**A/B** 等价于 AB^{-1}
 - ④ 幂运算：**A^p**，即 p 个 A 相乘，A 是方阵，p 是正整数
 - ⑤ 矩阵的转置：**A'**（共轭转置），**A.'**（普通转置，不取共轭）
- 矩阵的数组运算：对应元素进行运算，包括点乘（**.***），点除（**./** **.**），点幂（**.^**）
 - ① 点与算术运算符之间不能有空格
 - ② 参与运算的对象必须具有相同的形状
- 矩阵的 Kronecker 乘积：**kron(A,B)**
- 矩阵的翻转与旋转

flip1r(A)	左右翻转
flipud(A)	上下翻转
rot90(A)	逆时针旋转 90 度
rot90(A,k)	逆时针旋转 k*90 度

- 改变矩阵的形状和查看矩阵的大小

reshape(A,m,n)	将矩阵元素按列方向重新排列，生成一个 $m \times n$ 的新矩阵
size(A)	返回矩阵 A 的行数和列数
size(A,1)	返回矩阵 A 的行数
size(A,2)	返回矩阵 A 的列数
length(x)	若 x 是向量，则返回 x 的长度
length(X)	若 X 是矩阵，则返回行数和列数中大的一个
numel(A)	返回 A 的元素的个数

- 函数作用在矩阵上的取值：设 **f** 是函数，**x** 是变量（可以是向量或矩阵），则

- ① $f(x)$ 是作用在 x 的每个分量上;
- ② 若 x 是矩阵, 则 $f(x)$ 是一个与 x 同形状的矩阵
- 矩阵与数的运算
 - ① 加减: 矩阵的每个元素都与数作加减运算;
 - ② 矩阵乘以一个数: 矩阵的每个元素都与数作乘法运算;
 - ③ 矩阵除以一个数: 每个元素都除以这个数;
 - ④ 数与矩阵的点幂运算: 采用数组运算, 如 $2.^{[1,2,3;4,5,6]}$
- 变量的输出格式: **format**
 - ① 在屏幕上输出运算结果时, 可以通过 **format** 命令指定输出格式;
 - ② **format** 只改变输出格式, 不会改变变量的值;

format short	短格式 (缺省格式), 小数点后 4 位
format long	长格式, 双精度输出小数点后 15 位, 单精度数为 7 位
format shortE	短 E 格式 (科学计数格式)
format longE	长 E 格式
format shortG	短 G 格式
format longG	长 G 格式
format rat	分数形式
format compact	紧凑格式
format loose	宽松格式
format hex	十六进制格式
format bank	银行格式
format +	分别用 +, -和空格表示正数, 负数和零

- 变量的存储和读取

save	将所有变量保存到文件 matlab.mat 中
save fname save fname.mat	将所有变量保存到文件 fname.mat 中, 文件扩展名可以省略
save fname 变量列表	将指定变量保存到文件 fname.mat 中, 存储多个变量时, 变量之间用空格隔开
load fname	读取 fname.mat 中的所有变量
load fname 变量列表	从 fname.mat 中读取指定的变量

- 寻找帮助

help	显示指定命令的简短使用说明
doc	以网页形式显示完整的使用帮助
lookfor	根据指定的关键词查询与之相关的命令

- 几个小技巧

- ① Matlab 的命令记忆功能: 上下箭头键, 可以先输入命令的前几个字符, 再按上下键缩小搜索范围
- ② 命令补全功能: **Tab** 键;
- ③ **Esc** 键: 删除当前命令行的所有内容
- ④ 两个清屏命令: **home**、**clc**

1.3 常用数学函数

- 三角函数

sin	正弦	sec	正割	tanh	双曲正切
asin	反正弦	asec	反正割	atanh	反双曲正切
cos	余弦	csc	余割	coth	双曲余切
acos	反余弦	acsc	反余割	acoth	反双曲余切
tan	正切	sinh	双曲正弦	sech	双曲正割
atan	反正切	asinh	反双曲正弦	asech	反双曲正割
cot	余切	cosh	双曲余弦	csch	双曲余割
acot	反余切	acosh	反双曲余弦	acsch	反双曲余割

● 指数函数

exp(x)	指数函数	log(x)	自然对数（以 e 为底）
sqrt(x)	平方根	log2(x)	以 2 为底的对数
nthroot(x)	n 次方根	log10(x)	常用对数（以 10 为底）
power(x,y)	幂函数 x^y	pow2(x)	2^x

● 复数运算函数

abs(x)	绝对值或模	conj(z)	取复数共轭
real(z)	取复数实部	angle(z)	相角
imag(z)	取复数虚部	complex(x,y)	用实部和虚部构成复数

● 取整与求余

ceil(x)	朝正无穷大方向取整	mod(m,n)	模数求余
fix(x)	朝 0 方向取整	rem(m,n)	求余数
floor(x)	朝负无穷大方向取整	sign(x)	符号函数
round(x)	四舍五入取整	gcd(m,n)	最大公约数
factor(n)	分解成素数的乘积	lcm(m,n)	最小公倍数
factorial(n)	阶乘	primes(n)	不超过 n 的所有素数

● 数据分析

sum(x)	求和	sort(x)	排序
max(x)	最大值	sortrows(X)	按行排序
min(x)	最小值	hist(x)	统计频数直方图
mean(x)	平均值	histc(x)	直方图统计
median(x)	中值	std(x)	标准差
prod(x)	元素乘积	var(x)	求方差
intmin	能表示的最小整数	realmin	能表示的最小实数
intmax	能表示的最大整数	realmax	能表示的最大实数

● 矩阵分析

det	行列式	lu	LU 分解
inv	矩阵的逆	chol	Cholesky 分解
rank	矩阵的秩	qr	QR 分解
trace	矩阵的迹	schur	Schur 分解
norm	范数	eig	矩阵特征值和特征向量
cond	矩阵条件数	eigs	部分特征值（稀疏矩阵）
null	零空间	svd	奇异值分解
orth	值空间	svds	部分奇异值
poly	特征多项式	qz	广义特征值

- 多项式运算

conv	多项式相乘	polyint	多项式积分
deconv	多项式相除	polyval	多项式求值
poly	由根创建多项式	polyvalm	矩阵多项式求值
polyder	多项式微分	roots	多项式求根
polyfit	多项式拟合		

- 数值积分

trapz	梯形法
quad	自适应 Simpson 积分法
quadl	高阶自适应 Lobatto 积分法
dblquadl	自适应 Simpson 二重积分
integral	全局自适应积分
integral2	二重积分
integral3	三重积分

- 常微分方程求解

ode45	高阶法解微分方程
ode23	低阶法阶微分方程
ode23s	低阶法解刚性微分方程
ode23t	解适度刚性微分方程
ode23tb	低阶法解刚性微分方程
ode113	变阶法解微分方程
ode15s	变阶法解刚性微分方程
ode15i	隐式微分方程求解

第二讲、Matlab 绘图

2.1 平面曲线

- 一般步骤:

设曲线方程 $y=f(x)$, $x \in [a,b]$

① 分割绘图区域: $x=a:h:b$ % 这里 h 为步长, 通常取为 $h=(b-a)/N$

② 计算函数值: $y=f(x)$ % 这里需要注意使用数组运算

③ 调用绘图函数: **plot(x,y)**

- 注意: 也可以使用参数方程绘图: $x=x(t)$, $y=y(t)$, $t \in [a,b]$

- 绘图函数

plot	直角坐标下线性刻度绘图
fplot	函数绘图
semilogx	X 轴半对数刻度绘图
semilogy	Y 轴半对数刻度绘图
polar	极坐标绘图
loglog	双对数刻度曲线
plotyy	双纵坐标图

- 指定图形的属性: 点的形状、线的形状、颜色

plot(x,y,'str')	str 为一字符串, 指定图形的属性
plot(x1,y1,'str1',x2,y2,'str2',...)	同时绘制多条曲线

- 点的形状、线的形状、颜色

点的形状	线的形状	颜色
. 点	- 实线	y 黄色
o 小圆圈	: 虚线	m 洋红/magenta
x 叉子符	-. 点划线	c 青色/cyan
+ 加号	-- 间断线	r 红色
* 星号	空白(不画线)	g 绿色
s 方格		b 蓝色
d 菱形		w 白色
^ 朝上三角		k 黑色
v 朝下三角		
> 朝右三角		
< 朝左三角		
p 五角星		
h 六角星		
缺省(不画点)		

- 图像注释

title('str')	图形标题
xlabel('str')	X 轴标注
ylabel('str')	Y 轴标注
legend('str1','str2',...)	图例
text(x,y,'str')	在图上标注文字

- 坐标轴控制等

<code>axis([x1,x2,y1,y2,z1,z2])</code>	设置坐标轴显示范围
<code>axis auto</code>	由系统自动设置坐标轴显示范围
<code>axis equal</code>	各坐标轴采用等长刻度
<code>axis square</code>	使绘图区域为正方形
<code>axis manual</code>	以当前的坐标限制图形的绘制（多图时）
<code>axis normal</code>	恢复缺省设置
<code>set(gca,'xtick',[x0,x1,x2,...])</code>	手工设置刻度
<code>hold on/off</code>	图形的保持
<code>grid on/off</code>	网格控制
<code>subplot(m,n,p)</code>	创建子图
<code>colormap(gray)</code>	着色方案
<code>figure(n)</code>	新建/选取绘图窗口
<code>close, close all</code>	关闭绘图窗口
<code>clf</code>	清空绘图窗口
<code>shg</code>	显示绘图窗口

2.2 空间曲线

- 一般步骤:

设曲线参数方程: $x=x(t), y=y(t), z=z(t), t \in [a,b]$

- ① 分割绘图区域: $t=a:h:b$ % 这里 h 为步长
- ② 计算函数值: $x=x(t); y=y(t); z=z(t)$ % 这里需要注意使用数组运算
- ③ 调用绘图函数: `plot3(x,y,z)`

- 注意:

- ① 只能使用参数方程绘图;
- ② `plot3` 的用法与 `plot` 类似

2.3 空间曲面

- 一般步骤:

设曲面方程: $z=z(x,y), x \in [a,b], y \in [c,d]$

- ① 分割绘图区域: $x=a:h_1:b; y=c:h_2:d$ % 这里 h_1, h_2 为步长
- ② 生成网格: `[X,Y]=meshgrid(x,y)`
- ③ 计算函数值: $Z=z(X,Y)$ % 这里需要注意使用数组运算
- ④ 调用绘图函数: `mesh(X,Y,Z)`

- 注意:

- ① 也可以使用参数方程绘图: $x=x(u,v), y=y(u,v), z=z(u,v), u \in [a,b], v \in [c,d]$
- ② 图形剪切: `find`

- 绘图函数

<code>mesh</code>	三维网格图	<code>surf</code>	三维表面图
<code>meshc</code>	带等高线的三维网格图	<code>surfz</code>	带等高线的三维表面图
<code>meshz</code>	带屏蔽面的三维网格图	<code>sphere</code>	单位球面

2.4 符号绘图

<code>ezplot(f(x), [a,b])</code>	函数 f 在 $[a,b]$ 区间内的图形
----------------------------------	-------------------------

<code>ezplot(f(x,y),[a,b,c,d])</code>	隐式方程绘图, $x \in [a,b], y \in [c,d]$
<code>ezplot(x(t),y(t),[a,b])</code>	参数方程绘图, $t \in [a,b]$
<code>ezplot3(x(t),y(t),z(t),[a,b])</code>	空间曲线, 只能使用参数方程
<code>ezpolar(r(t),[a,b])</code>	极坐标方程绘图
<code>ezmesh(z(x,y),[a,b,c,d])</code>	绘制空间曲面网格图, $x \in [a,b], y \in [c,d]$
<code>ezmesh(x(s,t),y(s,t),z(s,t),[a,b,c,d])</code>	参数方程绘图, $s \in [a,b], t \in [c,d]$
<code>ezsurf</code>	空间曲面表面图, 用法同 <code>ezmesh</code>

● 注意:

- ① `ezplot` 可以使用显式方程、参数方程或隐式方程绘制二维曲线
- ② `ezplot3` 只能使用参数方程绘制三维曲线
- ③ `ezmesh` 和 `ezsurf` 可以使用显式方程或参数方程绘制空间曲面图
- ④ 作图时需要注意自变量或参数的取值范围

第三讲、符号运算

3.1 符号运算基础

- 进行符号运算时，一般需要**先定义符号对象**（符号变量，符号表达式），然后再进行相应的符号运算
- 基本函数

sym	定义符号变量或表达式	subs	符号替换
syms	定义符号变量	vpa	指定精度
symvar	列出表达式中的符号变量		

3.2 常见符号运算

- 因式分解，函数展开，合并同类项，通分

factor(f), factor(n)	因式分解，也可以对整数进行分解
expand(f)	函数展开
collect(f,v)	对指定的符号变量进行合并同类项
[N,D]=numden(f)	通分

- 简化，符号级数求和

simplify(f)	简化
simple(f)	简化（尝试多种方法）
symsum	符号级数求和

- 微积分

limit(f,x,a)	计算极限 $\lim_{x \rightarrow a} f(x)$
limit(f,x,a,'left')	计算左极限
limit(f,x,a,'right')	计算右极限
diff(f,x)	计算 f 关于变量 x 的导数
diff(f,x,n)	计算 f 关于变量 x 的 n 阶导数
int(f,x,a,b)	计算定积分 $\int_a^b f(x) dx$
int(f,x)	计算不定积分 $\int f(x) dx$
taylor(f,x,a)	计算 f 在 x=a 处的五阶 Taylor 展开
taylor(f,x,a,'order',N)	计算 f 在 x=a 处的 N-1 阶 Taylor 展开

- 其它

horner(f)	写成 horner 多项式形式
finverse(f,v)	反函数
symsum(f,v,a,b)	符号级数求和
solve(f,v)	代数方程求解
dsolve	常微分方程求解

第四讲、编程基础

4.1 M 文件

- 用 Matlab 语言编写的程序称为 M 文件
- M 文件以 .m 为扩展名
- M 文件有两大类：脚本（script）和函数（function）

4.2 关系运算与逻辑运算

- 关系运算

<code>==</code>	等于	<code>~=</code>	不等于
<code><</code>	小于	<code>></code>	大于
<code><=</code>	小于或等于	<code>>=</code>	大于或等于

- 逻辑运算

<code>&</code>	逻辑“与”	<code>&&</code>	逻辑“与”（短格式）
<code> </code>	逻辑“或”	<code> </code>	逻辑“或”（短格式）
<code>~</code>	逻辑“非”	<code>all</code>	所有元素非零时为真
<code>xor</code>	逻辑“异或”	<code>any</code>	有非零元素则为真
		<code>find</code>	找出非零元素所在的下标

- 测试函数

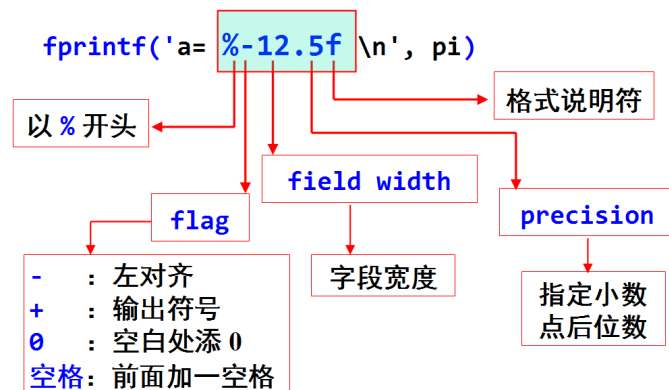
<code>isfinite</code>	是否为有限值	<code>ismatrix</code>	是否为矩阵
<code>isinf</code>	是否为无穷大	<code>isvector</code>	是否为向量
<code>isnan</code>	是否为不定值	<code>isscalar</code>	是否为标量
<code>isreal</code>	是否无虚部	<code>iscolumn</code>	是否为列向量
<code>isstr</code>	是否为一个字符串	<code>isrow</code>	是否为行向量
<code>isempty</code>	是否为空	<code>islogical</code>	是否为逻辑型数据
<code>isprime</code>	是否为素数	<code>isnumeric</code>	是否为数值型数据
<code>isequal</code>	对应元素是否相等	<code>isa</code>	是否为指定的数据类型
<code>ismember</code>	是否为某个数组的成员	<code>exist('A')</code>	变量 A 是否存在

4.3 输入输出

- 输入输出

<code>disp(A)</code>	在屏幕上输出向量的值
<code>x=input('str')</code>	提示键盘输入
<code>fprintf(fid,format,variables)</code>	格式化输出
<code>str=sprintf(fid,format,variables)</code>	将输出转化为字符串

- 格式化输出示例



- 格式化输出：常用格式说明符与转义字符

常用格式说明符		常用转义字符	
s	字符串	\b	后退一格
d	十进制整数	\n	换行
e	浮点数（科学计数法）	\t	水平制表符
f	浮点数（小数形式）	\\	反斜杠
g	浮点数（由系统自动决定）	'	单引号（两个连续的单引号）
o, x	八进制，十六进制整数	%	百分号（两个连续的百分号）

4.4 选择与循环

- 选择结构

if	条件执行语句	switch	多个条件分支
else	同 if 一起使用	case	同 switch 一起使用
elseif	同 if 一起使用	otherwise	同 switch 一起使用
end	结束 for, while, if 语句		

- 循环结构

for	for 循环	continue	结束本轮循环，进行下一轮循环
while	while 循环	break	终止最内层循环
pause	暂停	return	返回

- 计算运行

eval	字符串宏指令	builtin	执行内建的函数
evalc	执行 Matlab 字符串	run	执行脚本文件
feval	函数宏指令		

- 其它

global	定义全局变量	isglobal	是否为全局变量
persistent	定义永久变量	iskeyword	是否为关键字
mfilename	正在运行的 M 文件名		

4.2 函数文件

- 参数处理

nargin	实际输入参数的个数	inputname	实际调入变量名
nargout	实际输出参数的个数	varargin	输入参数
nargchk	输入变量个数检查	varargout	输出参数
nargoutchk	输出变量个数检查		