

# matlab编程基础

## 1 程序设计概述

- MATLAB是第四代编程语言
  - 程序简洁
  - 可读性很强
  - 调试容易
  - 编程效率高
  - 易移植和维护
- 通过编程可求解复杂性更高或特殊的问题

## 2 MATLAB程序结构

- 按照程序设计的观点,任何算法功能都可以通过由程序模块组成的三种基本程序结构的组合来实现:
  - 顺序结构: 程序按程序语句或模块在执行流中的顺序逐个执行。
  - 选择结构: 程序按设定的条件实现程序执行流的多路分支。
  - 循环结构: 程序按给定的条件重复地执行指定的程序段或模块。

### 2.1 顺序结构

语句后面加“;”号, 不显示运算结果

%开头表示是注释语句

赋值语句

变量=表达式

表达式 (相当于将值付给 ans 变量)

#### 1. 数据的输入

从键盘输入数据, 则可以使用input函数来进行, 该函数的调用格式为:

**A=input(提示信息, 选项);**

其中提示信息为一个字符串, 用于提示用户输入什么样的数据。

如果在input函数调用时采用's'选项(默认把输入作为一个表达式看待), 则允许用户输入一个字符串。例如, 想输入一个人的姓名, 可采用命令:

```
xm=input('What's your name?','s');
```

#### 2. 数据的输出

MATLAB提供的命令窗口输出函数主要有disp函数, 其调用格式为

**disp(输出项)**

其中输出项既可以为字符串, 也可以为矩阵。

- 例4-1 输入x,y的值, 并将它们的值互换后输出。

程序如下:

```
x=input('Input x please. ');  
y=input('Input y please. ');  
z=x;
```

```

x=y;
y=z;
disp(x);
disp(y);

```

- 例4-2 求一元二次方程 $ax^2+bx+c=0$ 的根。

程序如下：

```

a=input('a=?');
b=input('b=?');
c=input('c=?');
d=b*b-4*a*c;
x=[(-b+sqrt(d))/(2*a),(-b-sqrt(d))/(2*a)];
disp(['x1=',num2str(x(1)),'x2=',num2str(x(2))]);

```

### 3. 程序的暂停

暂停程序的执行可以使用`pause`函数，其调用格式为：

```
pause(延迟秒数)
```

如果省略延迟时间，直接使用`pause`，则将暂停程序，直到用户按任一键后程序继续执行。

若要强化中止程序的运行可使用`Ctrl+C`命令。

## 2.2 选择结构

- 在MATLAB中，选择结构由两种语句结构实现：

- if语句
- switch语句

### 1. if 语句

```

if表达式
    程序模块1
end

if 表达式
    程序模块1
else
    程序模块2
end

```

例4-3 计算分段函数的值。

程序如下：

```

x=input('请输入x的值:');
if x<=0
    y=(x+sqrt(pi))/exp(2);
else
    y=log(x+sqrt(1+x*x))/2;
end
y

```

实现多分支选择结构

```

If 表达式
    程序模块1
elseif 表达式2
    程序模块2
.....
elseif 表达式m
    程序模块m
else
    程序模块n
end

```

【例 1】一个简单的多分支结构。

```

cost=10;
number=12;
if number>8
    sums=number*0.95*cost;
elseif number>5
    sums=number*0.85*cost;
else
    sums=number*0.5*cost;
end
sums

```

```

sums =
    114.0000

```

## 2. switch语句

```

switch 表达式
    case 数值1
        程序模块1;
    case 数值2
        程序模块2;
    .....
    otherwise
        程序模块n
    .....
end

```

当表达式的值等于**数值1**的值时，执行语句组1，当表达式的值等于**数值2**的值时，执行语句组2，...，当表达式的值等于**数值m**的值时，执行语句组m，当表达式的值不等于case所列的表达式的值时，执行语句组n。当任意一个分支的语句执行完后，直接执行switch语句的下一句。

```

i=input('请输入数字 0-5:');
switch i %5 分制转等级制
    case 5
        Rank='优秀';
    case 4

```

```

        Rank='良好';
    case 3
        Rank='及格';
    otherwise
        Rank='不及格';
    end
Rank

```

### 3. try-catch 结构

try-catch 结构应用实例。

```

clear,N=4;A=magic(3);          %设置 3 行 3 列矩阵 A。
try
    A_N=A(N,:),                %取 A 的第 N 行元素
catch
    A_end=A(end,:),            %如果取 A(N,:)出错，则改取 A 的最后一行。
end
lasterr                         %显示出错原因
A_end =
     4     9     2
ans =
    Index exceeds matrix dimensions.

```

## 2.3 循环结构

● 在MATLAB中，循环结构可以由两种语句结构实现：

- for语句循环结构
- while语句循环结构

### 1. for语句循环结构

**For** 循环变量=起始值：步长：终止值  
循环体

**end**

例4-8：使用for结构计算 $1+2+3+\dots+100$

```

clear
sum=0;
for i=1:100
    sum=sum+i;
end
sum

```

1 到 100 偶数和

```

sum=0;
for i=2:2:100
    sum=sum+i;
end
sum

```

例 一个三位整数各位数字的立方和等于该数本身则称该数为水仙花数。输出全部水仙花数。

程序如下：fix零方向取整，floor- $\infty$ 方向，ceil+ $\infty$ 方向,round四舍五入

```
for m=100:999
    m1=fix(m/100);           %求m的百位数字
    m2=rem(fix(m/10),10);    %求m的十位数字 rem求余数
    m3=rem(m,10);           %求m的个位数字
    if m==m1*m1*m1+m2*m2*m2+m3*m3*m3
        disp(m)
    end
end
```

## 2. while语句循环结构

**while**表达式

循环体

**end**

【例】Fibonacci 数组的元素满足 Fibonacci 规则： $a_{k+2} = a_k + a_{k+1}$ ， $(k = 1, 2, \dots)$ ；且  $a_1 = a_2 = 1$ 。现要求计算出该数组中第一个大于 10000 的元素。

```
a(1)=1;a(2)=1;i=2;
while a(i)<=10000
    a(i+1)=a(i-1)+a(i);    %当现有的元素仍小于 10000 时，求解下一个元素。
    i=i+1;
end;
i,a(i)
i =
    21
ans =
    10946
```

## 3 M文件

### ● M文件概述

用MATLAB语言编写的程序，称为M文件。M文件可以根据调用方式的不同分为两类：

1. 命令文件(Script File)
2. 函数文件(Function File)。

### 3.1 命令文件

- 又称为脚本文件，是一串指令的集合。
- 用户只需键入该命令文件的文件名，系统就会自动逐行执行该文件中的各条语句。
- 适用于小规模运算。

### 3.2 函数文件

- 函数文件由**function**语句引导，函数文件基本格式：

**function** [输出形参列表]=函数名(输入形参列表)  
    注释说明语句段  
    程序语句段

### 3.3 M文件的建立与打开

- M文件是一个文本文件，它可以用任何编辑程序来建立和编辑，而一般常用且最为方便的是使用**MATLAB**提供的文本编辑器。

#### 1. 建立新的M文件

为建立新的M文件，启动MATLAB文本编辑器有3种方法：

- (1) 菜单操作。从MATLAB主窗口的**File**菜单中选择**New**菜单项，再选择**M-file**命令，屏幕上将出现MATLAB 文本编辑器窗口。
- (2) 命令操作。在MATLAB命令窗口输入命令**edit**，启动MATLAB文本编辑器后，输入M文件的内容并存盘。
- (3) 命令按钮操作。单击MATLAB主窗口工具栏上的New M-File命令按钮，启动MATLAB文本编辑器后，输入M文件的内容并存盘。

#### 2. 打开已有的M文件

打开已有的M文件，也有3种方法：

- (1) 菜单操作。从MATLAB主窗口的**File**菜单中选择**Open**命令，则屏幕出现Open对话框，在Open对话框中选中所需打开的M文件。在文档窗口可以对打开的M文件进行编辑修改，编辑完成后，将M文件存盘。
- (2) 命令操作。在MATLAB命令窗口输入命令：**edit 文件名**，则打开指定的M文件。
- (3) 命令按钮操作。单击MATLAB主窗口工具栏上的Open File命令按钮，再从弹出的对话框中选择所需打开的M文件。

- 函数文件的创建要求：

1. 文件名与函数名必须相同，如  $\sin(x)$  必有 **sin.m** 函数文件存在。
2. 要求**实参和形参位置一一对应**。形参在工作空间中不会存在。
3. 可以编写递归函数，可以嵌套其他函数。
4. 可以用 **return** 命令返回，也可以执行到终点返回

例4.10 建立一个命令文件将变量**a,b**的值互换，然后运行该命令文件。

首先建立命令文件并以文件名**exch.m**存盘：

```
clear;  
a=1:10;  
b=[11,12,13,14;15,16,17,18];  
c=a; a=b; b=c;  
a  
b
```

然后在MATLAB的命令窗口中输入**exch**，将会执行该命令文件。

例4.11 建立一个函数文件将变量a,b的值互换，然后在命令窗口调用该函数文件。

首先建立函数文件exch.m:

```
function [a,b]=exch(a,b)
```

```
c=a;a=b;b=c;
```

然后在MATLAB的命令窗口调用该函数文件:

```
clear;
```

```
x=1:10; y=[11,12,13,14;15,16,17,18];
```

```
[x,y]=exch(x,y)
```

### 3.4 函数调用

- 函数调用的一般格式是:

[输出实参表]=函数名(输入实参表)

要注意的是，函数调用时各实参出现的顺序、个数，应与函数定义时形参的顺序、个数一致，否则会出错。函数调用时，先将实参传递给相应的形参，从而实现参数传递，然后再执行函数的功能。

- 例4-12 利用函数文件，实现直角坐标(x,y)与极坐标( $\rho, \theta$ )之间的转换。

函数文件tran.m:

```
function [rho, theta]=tran( x, y)
```

```
rho=sqrt( x*x + y*y );
```

```
theta=atan( y / x );
```

调用tran.m的命令文件main1.m:

```
x=input('Please input x=:');
```

```
y=input('Please input y=:');
```

```
[rho, the]=tran( x, y );
```

```
rho
```

```
the
```

### 3.5 全局变量与局部变量

- 全局变量用global命令定义，格式为:

global 变量名

例4-15 全局变量应用示例。

先建立函数文件wadd.m，该函数将输入的参数加权相加。

```
function f=wadd( x , y )
```

```
global ALPHA BETA
```

```
f=ALPHA*x+BETA*y;
```

在命令窗口中输入:

```
global ALPHA BETA
```

```
ALPHA=1;
```

```
BETA=2;
```

```
s=wadd(1,2)
```

### 3.6 程序调试

- 程序调试概述

一般来说，应用程序的错误有两类，一类是语法错误，另一类是运行时的错误。语

法错误包括词法或文法的错误，例如函数名的拼写错、表达式书写错等。

程序运行时的错误是指程序的运行结果有错误，这类错误也称为程序逻辑错误。

- 调试器

1. Debug菜单项

该菜单项用于程序调试，需要与Breakpoints配合使用。

选择 Debug 菜单，其各项命令功能如下：

Step :逐步执行程序。

Step in :进入子程序中逐步执行调试程序。

Step out :跳出子程序中逐步执行调试程序。

run:执行 M-文件。

Go Until Cursor :执行到光标所在处。

Exit Debug Mode :跳出调试状态。