

MATLAB App Designer 设计预备知识

MATLAB 是矩阵实验室(Matrix Laboratory)的简称,它是一种高级编程语言,也是面向 科学计算、可视化及交互式程序设计的高科技计算环境。它将数值分析、矩阵计算、科学数据 可视化及非线性动态系统的建模和仿真等诸多强大功能集成在一个易于使用的视窗环境中,并 在很大程度上摆脱了传统非交互式程序设计语言的编程模式。

🔅 本章要点

- (1) MATLAB 软件介绍。
- (2) MATLAB 工作环境。
- (3) MATLAB 中的函数类型。
- (4) MATLAB 程序设计结构。

🔕 学习目标

- (1) 了解 MATLAB 软件的组成及特点。
- (2) 熟悉 MATLAB 工作环境。
- (3) 掌握 MATLAB 中的几种函数类型的使用方法及特点。
- (4) 掌握 if、switch 语句实现选择结构的方法。
- (5) 掌握 for、while 语句实现循环结构的方法。

1.1 MATLAB 软件介绍

MATLAB 主要由核心部分和各种应用工具箱两部分组成。

MATLAB核心部分由MATLAB开发环境、MATLAB编程语言、MATLAB数学函数库、MATLAB图形处理系统和MATLAB应用程序接口5部分组成。①MATLAB开发环境是一个集成的工作环境,包括MATLAB命令行窗口、文件编辑调试器、工作区、数组编辑器和在线帮助文档等;②MATLAB编程语言是一种面向科学与工程计算的高级语言,允许按照数学习惯的方式编写程序;③MATLAB数学函数库包含了大量的计算算法,包括基本函数、矩阵运算和复杂算法等;④MATLAB图形处理系统能够将二维和三维数组的数据用图形表示出来,并可以实现图像处理、动画显示和表达式作图等功能;⑤MATLAB应用程序接口使MATLAB语言能与C或FORTRAN等其他编程语言进行交互。

MATLAB工具箱包括功能性工具箱和科学性工具箱两类。①功能性工具箱主要用来 扩充其符号计算功能、图示建模仿真功能、文字处理功能以及与硬件实时交互功能,适用 于多种学科;②科学性工具箱专业性较强,如 control toolbox、communication toolbox等, 这些工具箱都是由该领域内学术水平很高的专家编写的,所以用户不用自己编写学科范围 内的基础程序即可直接进行高、精、尖的研究。

1.2 MATLAB 工作环境

MATLAB 既是一种语言,又是一种编程环境。在这种环境中,系统提供了许多编写、 调试和执行 MATLAB 程序的便利工具。下面主要介绍 MATLAB 中的命令行窗口、图形 窗口、工作空间窗口、M 文件编辑窗口和帮助系统窗口。

1.2.1 命令行窗口

通过命令行窗口输入 MATLAB 的各种命令并读出相应的结果。但要注意,每一条命 令或命令行输入后都要按 Enter 键,命令才会被执行,例如,输入如下程序:

a=or	nes(4,	4)			
然后	按 Ent	er 键,	即可信	创建一个 4×4 且元素值为 1 的矩阵,并显示如下运行结果:	
a =	1 1 1	1 1 1	1 1 1	1 1 1	
	1	1	1	1	

在 MATLAB 中,命令行窗口常用的命令及功能如表 1-1 所示。

表 1-1 命令行窗口常用的命令及功能

命令	功能
clc	清除命令行窗口中所有内容,光标回到屏幕左上角
clear	清除工作空间中所有的变量
clear all	从工作空间清除所有变量和函数
clear 变量名	清除指定的变量
clf	清除图形窗口内容
delete <文件名 >	从磁盘删除指定的文件
help < 命令行 >	查询所示命令的帮助信息
which < 文件名 >	查找指定文件路径
who	显示当前工作空间中的变量
whos	列出当前工作空间的变量及信息
what	列出当前目录下的 .m 文件和 .mat 文件
load name	加载 name 文件中的所有变量到工作空间
load name x,y	加载 name 文件中的变量 x, y 到工作空间
save name	保存工作空间变量到文件 name.mat 中
save name x, y	保存工作空间变量 x, y 到文件 name.mat 中
Home 或 Ctrl+A	光标移动到行首
End 或 Ctrl+E	光标移动到行尾
Esc 或 Ctrl+U	清除一行
Del 或 Ctrl+D	清除光标后的字符
Backspace 或 Ctrl+H	清除光标前的字符
Ctrl+K	清除光标至行尾字
Ctrl+C	中断程序运行

1.2.2 图形窗口

图形窗口用来显示 MATLAB 所绘制的图形,这些图形可以是二维图形,也可以是三维图形。只要执行了任意一种绘图命令,图形窗口就会自动产生,输入如下程序:

x=1:0.2:10; y=sin(x); plot(x,y) 运行结果如图 1-1 所示。

1.2.3 工作空间窗口

工作空间窗口用来显示当前计算机内存中 MATLAB 变量信息,包括变量的名称、数据结构、字节数及其类型等,如图 1-2 所示,所显示的变量信息可通过单击右上角"倒三角"图形,单击【选择列】按钮进行设置。同时,在 MATLAB 中可以对变量进行观察、编辑、保存和删除操作,若要查看变量的具体内容,可以双击该变量名称。



1.2.4 M 文件编辑窗口

MATLAB 命令行窗口适用于编写短小的程序,对于大型、复杂程序的编写,应采用 文件编辑方法,即编辑 M 文件。在 MATLAB 命令行窗口中输入 edit,即可启动编辑器, 并打开空白的 M 文件编辑窗口,如图 1-3 所示。



图 1-3 编辑器

MATLAB 文件编辑器具有编辑 M 文件和调试 M 文件两大功能。其中,编辑功能的选择、复制与粘贴、查找与替换等方法与 Windows 编辑程序类似,此处不再赘述,只对下列几点进行特别说明。

(1) 注释: Ctrl+R 快捷键增加注释, Ctrl+T 快捷键删除注释。

(2) 缩进:调整缩进格式以增加 M 文件的可读性。增加缩进量用 Ctrl+]键,减少缩进量用 Ctrl+[键。当一段程序比较乱时,使用快捷键 Ctrl+I 进行自动整理,也是一种很好的选择。

M 程序调试器的热键设置和 VC 有些类似,下面列出来一些常用的调试方法。

(1) 设置 / 清除断点: 使用快捷键 F12。

(2) 执行: 使用快捷键 F5。

(3) 单步执行: 使用快捷键 F10。

(4) step in: 遇到函数时,进入函数内部,使用快捷键 F11。

(5) step out: 执行流程跳出函数,使用快捷键 Shift+F11。

1.2.5 帮助系统窗口

MATLAB 拥有完善的帮助系统,熟练的程序开发人员会充分利用帮助系统所提供的 信息。常用的帮助命令有 help、demo、doc、who、whos、what、which、lookfor 等。

如果用户知道某个函数的名称,并想了解该函数的具体用法,只需在命令行窗口中

输入:

help+函数名

例如:在窗口中输入 help cos 就可以获得 cos 函数的信息。

```
help cos
cos Cosine of argument in radians.
cos(X) is the cosine of the elements of X.
See also acos, cosd, cospi.
cos 的文档
名为 cos 的其他函数
```

如果用户不知道函数的确切名称,此时 help 命令就无能为力了,但可以使用 lookfor 命令,即在使用 lookfor 命令时,用户只需知道某个函数的部分关键字,在命令行窗口 输入:

lookfor+关键字

1.3 MATLAB 中的函数类型_

MATLAB中的 M 文件有两种形式,即脚本文件和函数文件,其中,函数文件包括主函数、子函数、嵌套函数、私有函数和重载函数,下面分别进行介绍。

1.3.1 主函数

M 文件中的第一个函数称为主函数,一个 M 文件只能包含一个主函数,主函数之后 可附随多个子函数。并且, M 文件名应该与主函数定义名相同,外部函数只能对主函数进 行调用。M 文件主函数是相对其内部的子函数和嵌套函数而言的,一个 M 文件中除了主 函数外,还可包含多个嵌套函数或子函数。

1.3.2 子函数

一个 M 文件中可能包含多个函数,主函数之外的函数都称为子函数,所有的子函数 都有自己独立的声明、帮助和注释等结构,只需要在位置上注意位于主函数之后即可,而 各个子函数则没有前后顺序,可以任意放置。

M 文件内部发生函数调用时,MATLAB 首先检查该文件中是否存在相应名称的子函数,然后检查这一 M 文件所在目录的子目录下是否存在同名的私有函数,然后按照MATLAB 路径,检查是否存在同名的 M 文件或内部函数。

1.3.3 嵌套函数

任一 M 函数体内所定义的函数称为外部函数的嵌套函数, MATLAB 支持多重嵌套函数, 即在嵌套函数内部继续定义下一层的嵌套函数, 形如:

```
function x=A(p1,p2)
function y=B(p3)
...
end
...
```

MATLAB 函数体通常不需要 end 结束标记,但如包含嵌套函数,则该 M 文件内的所 有函数(主函数和子函数)都需 end 标记。

嵌套函数的调用规则: ①父级函数可调用下一层嵌套函数; ②相同父级的同级嵌套函数可相互调用; ③处于低层的嵌套函数可调用任意父级函数。

1.3.4 私有函数

end

私有函数是具有限制性访问权限的函数,是位于私有目录 private 下的函数文件,这些私有函数的构造与普通 M 函数完全相同,访问条件是:私有函数只能被存放于该 private 子目录上一层父目录中的函数文件调用。

1.3.5 重载函数

重载是计算机编程中非常重要的概念,它经常用于处理功能类似,但是参数类型或个 数不同的函数编写中。例如实现两个相同的计算功能,输入变量数量相同,不同的是其中 一个输入变量为双精度浮点类型,另一个输入变量为整型,这时用户就可以编写两个同名 函数,一个用来处理双精度浮点类型的输入参数,另一个用来处理整型的输入参数。

MATLAB的内置函数中有许多重载函数,放置在不同的文件路径下,文件夹名称以 @开头,然后跟一个代表 MATLAB 数据类型的字符。

1.4 MATLAB 程序设计结构

计算机编程语言程序设计结构主要有三大类:顺序结构、选择结构和循环结构。这一 点 MATLAB 与其他编程语言完全一致。

1.4.1 顺序结构

顺序结构是按照语句出现顺序执行的一种设计结构,即按语句由上到下的书写顺序执

行,只有一个入口和一个出口。在 MATLAB 的函数中,变量主要有输入变量、输出变量 及函数内所使用的变量。

1. 数据输入

可使用 input 函数,实现从键盘输入数据。

1) 输入数据

调用格式如下:

x=input('提示信息')

例如输入 x=input ('please input a number: '),运行结果如下:

2) 输入字符串

调用格式如下:

x=input(' 提示信息 ','s')

例如输入 x=input ('please input a string: ', 's'),运行结果如下:

```
x=input('please input a string:','s')
please input a string:this is a string
x =
```

```
'this is a string'
```

2. 数据输出

MATLAB 提供的命令行窗口输出函数主要有自由格式输出 disp 函数和格式化输出 fprintf 函数。

1) disp 函数

调用格式如下:

disp(输出项)

其中,输出项既可以是字符串,也可以是矩阵。

▶【例 1-1】disp 函数数据输出实例。

程序命令如下:

```
A=20+300-20*2;
B=[12 13 14;15 16 17;18 19 20];
C='this is a string';
disp(A);
disp(B);
disp(C);
运行结果如下:
280
```

12 13 14 15 16 17 18 19 20 this is a string 2) fprintf 函数

fprintf 函数可以将数据按指定格式输出到屏幕或指定文本文件中。

▶【例 1-2】fprintf 函数数据输出实例。

```
程序命令如下:
clc;
clear;
age=18;
name='洋洋';
fprintf('%s的年龄是%d\n',name,age)
运行结果如下;
```

洋洋的年龄是18

注意 %d 表示输出整数; %e 表示输出实数, 科学记数法形式; %f 表示输出实数, 小数形式; %s 表示输出字符串。

3. 程序的暂停

可使用 pause 函数执行程序暂停,其调用格式如下:

pause(延迟秒数)

如果没有指定延迟时间,直接使用 pause 即可暂停程序,直到用户按任意键后程序继续执行。若要强行中止程序的运行可使用 Ctrl+C 快捷键。

1.4.2 选择结构

选择结构又称为分支结构,用于判断给定的条件是否成立,并根据判断的结果来控制 程序执行的流程。MATLAB语言中的条件判断语句主要是 if 语句、switch 语句和 try 语句。

1. if 语句

1) 单分支 if 语句

- 语句格式如下:
- if 条件

语句组

end

当条件成立时,则执行语句组,执行完之后继续执行 if 语句的后续语句,若条件不成 立,则直接执行 if 语句的后续语句。

2) 双分支 if 语句

语句格式如下:

```
if 条件
语句组 1
else
语句组 2
```

end

当条件成立时,执行语句组1;不成立时执行语句组2。语句组1或语句组2执行后,再执行 if 语句的后续语句。

▶【例 1-3】计算分段函数的值, $y = \begin{cases} 3x+6, x>0 \\ 5x-2, x \leq 0 \end{cases}$ 。 程序命令如下:

```
x=input(' 请输入 x 的值:');
if x>0
   y=3*x+6
else
   v = 5 * x - 2
end
运行结果如下:
请输入 x 的值:8
у =
   30
3) 多分支 if 语句
if 条件1
     语句组 1
elseif 条件 2
     语句组 2
         . . .
```

```
elseif 条件 m
语句组 m
else
语句组 n
```

程序命令如下:

end



```
)
频讲解
```

```
L=input(' 请输入三角形的 3 条边:');
if L(1)<=0|L(2)<=0|L(3)<=0
    disp(' 三角形的边长不能为 0 或为负 ');
elseif L(1)+L(2)<=L(3)|L(1)+L(3)<=L(2)|L(2)+L(3)<=L(1)
    disp('不能构成三角形 ');
elseif L(1)==L(2)&L(2)==L(3)
    disp('构成等边三角形 ');
elseif L(1)==L(2)|L(1)==L(3)|L(2)==L(3)
    disp('构成等腰三角形 ');
else
    disp('构成一般三角形 ');
end
运行结果如下:
```

请输入三角形的 3 条边:[1 1 2] 不能构成三角形

请输入三角形的3条边:[2 3 4] 构成一般三角形

请输入三角形的3条边:[3 3 3] 构成等边三角形

2. switch 语句

多分支 if 语句用于实现多分支选择结构。if-else-end 语句所对应的是多重判断选择,而 有时也会遇到多分支判断选择的问题。MATLAB 语言为解决多分支判断的问题,提供了 switch 语句。switch 语句根据表达式的取值不同,有条件地执行一组语句,其语句格式如下:

```
switch 表达式
case 表达式 1
语句组 1
case 表达式 2
语句组 2
...
case 表达式 m
语句组 m
otherwise
语句组 n
end
```

与其他的程序设计语言的 switch 语句不同的是,在 MATLAB 语言中,当其中一个 case 语句后的条件为真时, switch 语句不对其后的其他 case 语句进行判断,也就是说在 MATLAB 语言中,即使有多条 case 判断语句为真,也只执行遇到的第一条为真的语句。 这样就不必像 C 语言那样,在每条 case 语句后加上 break 语句以防止继续执行后面为真的 case 条件语句。

▶【例 1-5】利用 switch 语句实现简单计算器功能,即输入两个操作数和运算符号,即可进行加减乘除运算。

```
程序命令如下:
N=input('请输入操作数:');
S=input('请输入运算符号:','s');
switch S
   case'+'
     Y=N(1)+N(2)
   case'-'
      Y = N(1) - N(2)
   case'*'
     Y=N(1)*N(2)
   case'/'
      Y=N(1)/N(2)
   otherwise
      disp('error operator')
end
运行结果如下:
请输入操作数:[8 23]
```

```
请输入运算符号:-
Y =
-15
```

3. try 语句

try 语句是错误检查语句,当程序运行在复杂的环境下时,一些语句可能会产生错误,导致程序停止执行,这时用户需要将这些语句放在 try...catch 结构中,其一般形式如下:

```
try
程序段 A;
catch
程序段 B;
end
```

逐行运行程序段 A,如果运行出错,就跳过程序段 A 后面的语句,改为执行程序段 B,此时命令行并不显示出错信息,若程序段 A 运行完没有出现错误,则跳过程序段 B,继续执行后面的程序。该语句结构也可以只包含 try 语句,不含 catch 语句,其一般形式如下:

```
try
程序段 A;
end
```

逐行运行程序段 A, 若运行出错, 就跳过程序段 A 后面的语句, 继续执行后面的 程序。

▶【例 1-6】矩阵的乘法运算。

```
程序命令如下:
```

```
A=[1,2,3;4,5,6]; B=[7,8,9;10,11,12];

try
C=A*B
catch
C=A.*B
end
s=lasterror;
disp(s.message) %显示出错原因
运行结果如下:
C =
7 16 27
```

```
40 55 72
ans =
Error using ==> mtimes
Inner matrix dimensions must agree.
```

1.4.3 循环结构

程序中总会有对某些量的迭代运算,或对某个过程的重复处理,这就需要使用循环来简化程序。下面分别介绍 for 循环和 while 循环。

1. for 循环语句

for 循环语句用于重复次数确定的循环,调用格式如下:

```
for 循环变量 = 表达式 1: 表达式 2: 表达式 3
```

循环体语句

end

其中,表达式1的值为循环变量的初值,表达式2的值为步长,表达式3的值为循环 变量的终值。当步长为1时,表达式2可以省略。

```
▶【例 1-7】sum=1-2+3-4+5-6···+99-100,利用 for 循环语句求 sum 的值。
程序命令如下:
```

```
k=1;
sum=0;
for i=1:1:100
    sum=sum+k*i;
    k=-1*k;
end
fprintf('sum的值为%d\n',sum);
运行结果如下:
sum的值为-50
```

▶【例 1-8】输入 5 个整数,统计并输出其中正整数、负整数和零的个数。

```
程序命令如下:
al=0;bl=0;cl=0;
for i=1:5
    fprintf('请输入第 %d 个数 :',i);
    x=input('');
    if x>0
        al=al+1;
    elseif x<0
        bl=bl+1;
    else
        cl=cl+1;
    end
end
fprintf('共有正整数 %d 个、负整数 %d 个、零 %d 个 ',al,bl,cl);
```

```
运行结果如下:
```

```
请输入第1个数:1
请输入第2个数:0
请输入第3个数:0
请输入第4个数:-1
请输入第5个数:5
共有正整数2个、负整数1个、零2个
```

▶【例 1-9】利用嵌套的 for 循环语句实现输出 9×9 乘法表。

```
程序命令如下:
```

```
for i=1:9
    for j=1:9
        for j=1:9
        result=i*j;
        fprintf( "%d*%d=%-3d ",i,j,result);%-3d 表示左对齐,占3位
        end
        fprintf( " \n " );%每一行后换行
end

cf结果如图 1-4 所示。
2. while 循环语句
while 循环语句是依条件结束的语句。调用格式如下:
while(表达式)
        循环体语句
end
```





```
      1*1=1
      1*2=2
      1*3=3
      1*4=4
      1*5=5
      1*6=6
      1*7=7
      1*8=8
      1*9=9

      2*1=2
      2*2=4
      2*3=6
      2*4=8
      2*5=10
      2*6=12
      2*7=14
      2*8=16
      2*9=18

      3*1=3
      3*2=6
      3*3=9
      3*4=12
      3*5=15
      3*6=18
      3*7=21
      3*8=24
      3*9=27

      4*1=4
      4*2=8
      4*3=12
      4*4=16
      4*5=20
      4*6=24
      4*7=28
      4*8=32
      4*9=36

      5*1=5
      5*2=10
      5*3=15
      5*4=20
      5*5=25
      5*6=30
      5*7=35
      5*8=40
      5*9=45

      6*1=6
      6*2=12
      6*3=18
      6*4=24
      6*5=30
      6*6=36
      6*7=42
      6*8=48
      6*9=54

      7*1=7
      7*2=14
      7*3=21
      7*4=28
      7*5=35
      7*6=42
      7*7=49
      7*8=56
      7*9=63

      8*1=8
      8*2=16
      8*3=24
      8*4=32
      8*5=40
      8*6=48
      8*7=56
      8*8=64
      8*9=72

      9*1=9
      9*2=18
      9*3=27
      9*4=36
      9*5=45
      9*6=54
      9*7=63
      9*8=72
      9*9=81
```

其中,表达式为某种形式的逻辑判断语句,当该表达式的值为真时,反复执行循环体 语句,直到表达式的逻辑值为假时,退出循环。

在 while 循环语句中必须有可以修改循环变量的命令,否则该循环语句将陷入死循环中。或者循环语句中有能控制退出循环的命令,例如 break 命令或 continue 命令。当程序流程运行至该命令时,将退出当前循环,执行循环后的其他语句。

▶【例 1-10】用 while 循环语句求出所有的水仙花数。水仙花数是一个 3 位数的自然数, 该数各位数的立方和等于该数本身。

见频讲解

程序命令如下:				
k=100;				
while(k<=999)				
x=fix(k/100);				
y=rem(fix(k/10),10);				
z=rem(k,10);				
if k==x*x*x+y*y*y+z*z*z				
<pre>fprintf('%d\n',k);</pre>				
end				
k=k+1;				
end				
运行结果如下:				
153				
370				

371 407

印卢人人加丁

3. continue、break 和 return

(1) continue: 用于循环控制。当不想执行循环体的全部语句,只想在做完某一步后 终止当前循环,直接返回到循环头,进行下一次循环时,可在此处插入 continue。continue 后面的语句将被跳过。

(2) break: 用在 for 或 while 循环中,立即结束本层循环,并继续执行循环之后的下一条语句。嵌套循环中,它只跳出所在层的循环。

(3) return:终止当前函数的继续执行,控制权交给调用函数或键盘。

1 本章小结

本章先介绍了 MATLAB 软件概述,进而介绍了 MATLAB 软件的工作环境和函数类

型,最后介绍了三大程序设计结构。本章是全书的基础,重点做到熟悉 MATLAB 编程环境和基本操作,为后续核心技术与工程应用的学习打下良好的基础。

」 习 题

1-1 输入一个整数,若为奇数则输出其平方根,否则输出其立方根。

1-2 输入5个数,求其中最大数和最小数,分别用循环结构和调用 max 函数、min 函数来实现。

1-3 计算1到999之间的所有偶数之和。

1-4 计算分段函数 $y = \begin{cases} \cos(x+1) + \sqrt{x^2+1}, x=10 \\ x \sqrt{x+\sqrt{x}}, x \neq 10 \end{cases}$

1-5 编写 M 文件,等待键盘输入,输入密码 20240101,密码正确,显示输入密码正确,程序结束;否则提示,重新输入。

1-6 有一群鸡和兔子,加在一起头的数量是 66,脚的数量是 220,编写 M 文件解答 鸡和兔子数量各是多少?

1-7 编写函数文件, 求 $y = \sum_{n=0}^{20} n!$, 将 *n* 作为函数的输入参数, *y* 作为函数的输出参数。

1-8 编写程序,求 $S_n = a + aa + aaa + \dots + \underbrace{aa \cdots a}_{n \uparrow}$ 的值,其中,数字a和表达式中位数最 多项a的个数n由键盘输入。