

## fopen , fprintf 和 sprintf 在 Matlab 中的应用

matlab 中 fopen 函数在指定文件打开的实例如下：

\*1) “ fopen打开文件，赋予文件代号。

语法 1：FID= FOPEN ( filename,permission )

用指定的方式打开文件

FID=+N(N 是正整数 )：表示文件打开成功，文件代号是 N.

FID=-1： 表示文件打开不成功。

FID 在此次文件关闭前总是有效的。

如果以读方式打开， matlab 首先搜索工作目录，其次搜索 matlab 的其他目录， “ permission 是打开方式参数。

打开方式参数由以下字符串确定：

r 读出

w 写入（文件若不存在，自动创建）

a 后续写入（文件若不存在，自动创建）

r+ 读出和写入（文件应已存在）

w+ 重新刷新写入，（文件若不存在，自动创建）

a+ 后续写入（文件若不存在，自动创建））

w 重新写入，但不自动刷新

a 后续写入，但不自动刷新

文件的存储格式：文件打开的默认方式是：二进制。以文本方式打开，可以在方式参

数 “ permission 中加入 “ 文件将，如 “ rt, ”“ wt+ ”

matlab 中 fprintf 函数的具体使用方法实例如下：

fprintf 函数可以将数据按指定格式写入到文本文件中。其调用格式为：

数据的格式化输出： fprintf(fid, format, variables)

按指定的格式将变量的值输出到屏幕或指定文件

fid 为文件句柄，若缺省，则输出到屏幕

1 for standard output (the screen) or 2 for standard error. If FID is omitted, output goes to the screen.

format 用来指定数据输出时采用的格式

%d 整数

%e 实数：科学计算法形式

%f 实数：小数形式

%g 由系统自动选取上述两种格式之一

%s 输出字符串

fprintf ( fid , format , A )

说明： fid 为文件句柄，指定要写入数据的文件， format 是用来控制所写数据格式的格式符，与 fscanf 函数相同， A 是用来存放数据的矩阵。

例 6.9 创建一个字符矩阵并存入磁盘，再读出赋值给另一个矩阵。

```
>> a='string';
```

```
>> fid=fopen('d:\char1.txt','w');
```

```
>> fprintf(fid,'%s',a);
```

```
>> fclose(fid);
```

```
>> fid1=fopen('d:\char1.txt','rt');
```

```
>> b=fscanf(fid1,'%s')
```

```
b =
```

```
string
```

matlab 读 txt 文件

```
fid=fopen('fx.txt','r');
```

%得到文件号

```
[f,count]=fscanf(fid,'%f %f',[12,90]);
```

%把文件号 1 的数据读到 f 中。其中 f 是 [12 90] 的矩阵

%这里 '%f %f' 表示读取数据的形势，他是按原始数据型读出

```
fclose(fid);
```

%关闭文件

另外有的 txt 文件还可以用 load 来打开

其语句为

```
f=load('fx.txt')
```

matlab 中 sprintf 和 fprintf 函数的对比实例如下：

字符串转换

abs

字符串到 ASCII 转换

dec2hex

十进制数到十六进制字符串转换

fprintf

把格式化的文本写到文件中或显示屏上

hex2dec

十六进制字符串转换成十进制数

hex2num

十六进制字符串转换成 IEEE 浮点数

int2str

整数转换成字符串

lower

字符串转换成小写

num2str

数字转换成字符串

setstr

ASCII 转换成字符串

sprintf

用格式控制，数字转换成字符串

sscanf

用格式控制，字符串转换成数字

str2mat

字符串转换成一个文本矩阵

str2num

字符串转换成数字

upper

字符串转换成大写

在许多情况下，希望把一个数值嵌入到字符串中。几个字符串转换可完成这个任务。

```
? rad=2.5; area=pi*rad^2;
```

```
? t=[' A circle of radius ' num2str(rad) ' has an area of ' num2str(area) ' . ' ] ;
```

```
? disp(t)
```

A circle of radius 2.5 has an area of 19.63.

这里函数 num2str 用来把数值转换成字符串，字符串连接用来把所转换的数嵌入到一个字符串句子中。按类似方式，

int2str 把整数转换成字符串。无论是 num2str 还是 int2str 都调用函数 sprintf ,它用类似 C 语言语法把数值转换成字符串。

函数 fprintf 经常是函数 disp 的一个有用替换，由于它提供了对结果更多的控制。当准备把格式化的数据写到一个文件中

去时，按缺省它在命令窗口显示结果。例如，

```
? fprintf(' See what this does ')
See what this does?
? fprintf(' See what this does\n ')
See what this does
```

在上面第一个例子里，`fprintf` 显示字符串，然后立即给出 MATLAB 提示符。相反，在第二个例子里，`\n` 插入一个新行字符，在 MATLAB 提示符出现之前创建一个新行。

无论 `fprintf` 还是 `sprintf` 以同样方式处理输入参量，但 `fprintf` 把输出送到显示屏或文件中，而 `sprintf` 把输出返回到一个字符串中。例如，上面的例子用 `num2str` 可重写为

```
? t=sprintf(' A circle of radius %.4g has an area of %.4g. ', rad, area);
? disp(t)
A circle of radius 2.5 has an area of 19.63.
? fprintf(' A circle of radius %.4g has an area of %.4g.\n ', rad, area)
A circle of radius 2.5 has an area of 19.63.
```

这里 `%.4g` 是用在函数 `num2str` 中的数据格式。`%.4g` 就是用指数或定点标记，不管哪一种更短些，只显示至 4 位数字。除了 `g` 格式，还可用 `e` (指数)和 `f` (定点)转换。表 6.2 表明在各种不同转换下，如何显示 `pi` 结果。

表 6.2  
数值格式转换例子

命令	
结果	
<code>fprintf(' %.0e\n ',pi)</code>	
	3e+00
<code>fprintf(' %.1e\n ',pi)</code>	
	3.1e+00
<code>fprintf(' %.3e\n ',pi)</code>	
	3.142e+00
<code>fprintf(' %.5e\n ',pi)</code>	
	3.14159e+00
<code>fprintf(' %.10e\n ',pi)</code>	
	3.1415926536e+00
<code>fprintf(' %.0f\n ',pi)</code>	
	3
<code>fprintf(' %.1f\n ',pi)</code>	
	3.1
<code>fprintf(' %.3f\n ',pi)</code>	
	3.142
<code>fprintf(' %.5f\n ',pi)</code>	
	3.14159
<code>fprintf(' %.10f\n ',pi)</code>	
	3.1415926536
<code>fprintf(' %.0g\n ',pi)</code>	
	3
<code>fprintf(' %.1g\n ',pi)</code>	
	3
<code>fprintf(' %.3g\n ',pi)</code>	
	3.14
<code>fprintf(' %.5g\n ',pi)</code>	
	3.1416
<code>fprintf(' %.10g\n ',pi)</code>	

```
3.141592654
```

```
fprintf(' %.8.0g\n ',pi)
```

```
3
```

```
fprintf(' %.8.1g\n ',pi)
```

```
3
```

```
fprintf(' %.8.3g\n ',pi)
```

```
3.14
```

```
fprintf(' %.8.5g\n ',pi)
```

```
3.1416
```

```
fprintf(' %.8.10g\n ',pi)
```

```
3.141592654
```

注意，对 e 和 f 格式，小数点右边的十进制数就是小数点右边要显示的多少位数字。相反，在 g 的格式里，小数点右边的十进制数指定了显示数字的总位数。另外，注意最后的五行，其结果指定为 8 个字符长度，且是右对齐。在最后一行，8 被忽略，因为指定超过了 8 位。

概括起来，当需要比缺省函数 disp，num2str 和 int2str 所提供的更多的控制时，fprintf 和 sprintf 是有用的。

函数 str2mat 把一系列的几个字符串转换成一个字符串矩阵。例如，

```
? a=' one '; b=' two '; c=' three ';
```

```
? disp(str2mat(a, b, c, ' four '))
```

```
one
```

```
two
```

```
three
```

```
four
```

从上面看不明显，上面的每行有同样数目的元素。较短行用空格补齐，使结果形成一个有效的矩阵

在逆方向转换中，有时是很方便的。

```
? s=' [1 2; pi 4] ' % a string of a MATLAB matrix
```

```
s =
```

```
[1 2; pi 4]
```

```
? str2num(s)
```

```
ans =
```

```
1.0000 2.0000
```

```
3.1416 4.0000
```

```
? s=' 123e+5 ' % a string containing a simple number
```

```
s =
```

```
123e+5
```

```
? str2num(s)
```

```
ans =
```

```
12300000
```

函数 str2num 不能接受用户定义的变量，也不能执行转换过程的算术运算。