

## 7.3 Mathematica 语言编程

### 1. 过程

一般形式为：

`Module[<局部变量名表>, 表达式 1 ; 表达式 2 ; ...表达式 n]`

例如：

```
unit[x_, y_, z_] := Module[{len}, len = Sqrt[x^2 + y^2 + z^2];  
N[{x/len, y/len, z/len} ]
```

`Block[<局部变量名表>, 表达式 1 ; 表达式 2 ; ...表达式 n]`

### 2. 控制选择

(i) 顺序控制 分号 “ ; ”

(ii) 条件控制

If 语言结构。

`If[逻辑表达式, 表达式]`

`If[逻辑表达式, 表达式 1, 表达式 2]`

`If[逻辑表达式, 表达式 1, 表达式 2, 表达式 3]`

例如：

```
f[x_] := If[(x > 0) || (x=0), N[Sqrt[x]], Print[“x is negative.”],  
Print[“x is not numerical.”]
```

Which 语句结构。

`Which[条件1, 表达式1, 条件2, 表达式2, ..., 条件n, 表达式n]`

例：`Which[ 2==3, x, 3==3, y ]`

其结果为y。这是由于条件2==3的结果为False, 而条件3==3的结果为True.

Switch 语句结构。

`Switch[判别表达式, 模式1, 表达式1, 模式2, 表达式2, ..., 模式n,`

表达式n]

例： $i=1$

`Switch[i^2, 0, x, 1, y, 2, z]`

最后结果为y。

(iii) 循环控制

Do语句结构。

`Do[表达式, {循环描述}]`

例：`Do[Print[2^i], {i, 1, 5}]`

该指令的结果是循环打印出 $2^i$ , ( $i=1, 2, 3, 4, 5$ ) 的值2, 4, 8, 16, 32。

For语句结构。

`For[初始表达式, 条件, 步进表达式, 循环表达式]`

例如：`For[ i=0, i<=10, ++1, Print[i] ]`  
值为0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10。

While语句结构。

例如：  
`i=0`  
`While[ i<=10, Print[i]; i++ ]`

### 3. 程序包结构

一般的程序包具有如下的基本框架：

`BeginPackage[“程序包名称”, “”]`

名字::usage=“字符串, 程序包中定义在包外可以使用的函数、变量等的名字和使用说明。”

.....

`Begin[“Private”]`

程序包主体.(包括外部可用的函数及变量,一些内部函数变量的定义.)

.....

End[ ]

EndPackage[ ]