

| | |
|-------------------|---|
| 一. 流程控制 | 1 |
| 1.1 如果命令 | 1 |
| 1.2 如果真命令 | 2 |
| 1.3 判断命令 | 2 |
| 1.4 判断循环首命令 | 2 |
| 1.5 判断首命令 | 3 |
| 1.6 计次循环首命令 | 3 |
| 1.7 变量循环首命令 | 4 |
| 1.8 到循环尾命令 | 4 |
| 1.9 跳出循环命令 | 5 |
| 1.10 返回命令 | 5 |
| 1.11 结束命令 | 6 |

一. 流程控制

1.1 如果命令

操作系统支持：Windows、Linux、Unix 所属类别：流程控制

本命令根据提供的逻辑参数的值，来决定是否改变程序的执行位置，如果提供的逻辑参数值为真，程序继续顺序向下执行，否则跳转到左侧箭头线指向的命令处去执行。

语法： 无返回值 如果 （条件）

使用：.版本 2

.如果 (编辑框 1.内容=真)

 否则

 .如果结束

1.2 如果真命令

操作系统支持：Windows、Linux、Unix 所属类别：流程控制

本命令根据提供的逻辑参数的值，来决定是否改变程序的执行位置，如果提供的逻辑参数数值为真，程序继续顺序向下执行，否则跳转到左侧箭头线指向的命令处去执行。同“如果”命令相比，本命令没有为假时的程序部分。

语法： 无返回值 如果真 （条件）

使用：版本 2

```
.如果真 (编辑框 1.内容 = “”)  
.    如果真结束
```

1.3 判断命令

操作系统支持：Windows、Linux、Unix 所属类别：流程控制

本命令根据提供的逻辑参数的值，来决定是否改变程序的执行位置，如果提供的逻辑参数数值为真，程序继续顺序向下执行，否则跳转到下一分支处去继续判断。

语法： 无返回值 判断 （条件）

使用：.版本 2

```
.子程序 _按钮 1_被单击  
.参数 变量 1, 整数型  
    变量 1 = 1  
.判断开始 (变量 1 = 1)  
输出调试文本 (“变量 1 的值是 1”)  
.默认  
.判断结束
```

1.4 判断循环首命令

操作系统支持：Windows、Linux、Unix 所属类别：流程控制

本命令根据提供的逻辑参数的值，来决定是否进入循环。如果提供的逻辑参数数值为真，程序顺序执行下一条命令进入循环，否则跳转到本命令所对应的“判断循环尾”命令的下一条命令处以跳出循环。

语法： 无返回值 判断循环首 （条件）

使用：版本 2

```
输出调用文本（子程序 2（））  
. 子程序 子程序 2, 整数型  
. 局部变量 和值, 整数型  
.局部变量 计次, 整数型
```

```
.判断循环首 (计次 ≤ 20)
    和值 = 和值 + 计次
    计次 = 计次 + 1
.判断循环尾 ()
返回 (和值)
```

1.5 判断首命令

操作系统支持: Windows、Linux、Unix 所属类别: 流程控制
执行本命令后将顺序执行下一条命令进入程序循环。

语法: 无返回值 循环判断首 ()

使用: .版本 2

```
.局部变量 变量 1, 整数型
' 写 10 以内的双数
.循环判断首 ()
变量 1 = 变量 1 + 2
画板 1.滚动写行 (变量 1)
.循环判断尾 (变量 1 ≠ 10)
```

1.6 计次循环首命令

操作系统支持: Windows、Linux、Unix 所属类别: 流程控制

执行本命令将把循环体内的命令重复执行指定的次数。提供的整数参数值决定将重复执行的次数。

语法: 无返回值 计次循环首 (循环次数, [已循环次数记录变量])

使用: 版本 2

```
.程序集 窗口程序集 1
.子程序 __启动窗口_创建完毕
.局部变量 a, 整数型
.局部变量 次数, 整数型
.局部变量 b, 整数型
a = 0
b = 50
.计次循环首 (b, 次数)
    a = a + 次数
.计次循环尾 ()
信息框 (“1 到” + 到文本 (b) + “的和为: ” + 到文本 (a), 0,)
```

1.7 变量循环首命令

操作系统支持：Windows、Linux、Unix 所属类别：流程控制

执行本命令将利用变量对循环体内的命令进行循环执行。第一次执行此命令时将使用“变量起始值”参数初始化“循环变量”参数所指定的变量。每次（包括第一次）执行到此命令处都将判断循环变量内的值是否已经超过“变量目标值”参数所指定的值，如已超过，则跳转到对应的“变量循环尾”命令的下一条命令处继续执行，否则顺序执行下一条语句继续进行循环。

语法： 无返回值 变量循环首 (变量起始值, 变量目标值, 变量递增值, [循环变量])

使用：.版本 2

.子程序 子程序 2, 整数型

.局部变量 和值, 整数型

.局部变量 计次, 整数型

.变量循环首 (0, 20, 2, 计次)

*变量起始值：0

*变量目标值：20

*变量递增值：2

*循环变量：计次

和值 = 和值 + 计次

.变量循环尾 ()

返回 (和值)

1.8 到循环尾命令

操作系统支持：Windows、Linux、Unix 所属类别：流程控制

本命令转移当前程序执行位置到当前所处循环体的循环尾语句处。

语法： 无返回值 到循环尾 ()

使用：.版本 2

.子程序 __启动窗口_创建完毕

.局部变量 a, 整数型

.局部变量 b, 整数型

.局部变量 c, 整数型

```
.计次循环首 (8, a)
  .计次循环首 (7, b)
    .计次循环首 (7, c)
      .如果真 (a ≥ b 或 b ≥ c)
        .如果真结束
          到循环尾 ()
        .计次循环尾 ()
      .计次循环尾 ()
    .计次循环尾 ()
  .计次循环尾 ()
输出调试文本 (a)
  本程序输出的结果: a=9
```

1.9 跳出循环命令

操作系统支持: Windows、Linux、Unix 所属类别: 流程控制
本命令转移当前程序执行位置到当前所处循环体循环尾语句的下一条语句处。
语法: 无返回值 跳出循环 ()

使用: .版本 2

```
.子程序 __启动窗口_创建完毕
.局部变量 a, 整数型
```

```
.计次循环首 (5, a)
  a = a + 1
  跳出循环 ()
```

```
.计次循环尾 ()
```

输出调试文本 (a)

该程序调试得到的结果: a=2

因为“跳出循环 ()”该循环语句只是循环了一次, 即初始值 a=1, a=a+1, 所以最后得 a=2

1.10 返回命令

操作系统支持: Windows、Linux、Unix 所属类别: 流程控制
本命令转移当前程序执行位置到调用本子程序语句的下一条语句处,并可根据需要返回一个值到子程序调用处。
语法: 无返回值 返回 ([返回到调用方的值])

使用：.版本 2

.子程序 __启动窗口_创建完毕

.局部变量 a, 整数型

.计次循环首 (5, a)

 a = a + 1

.计次循环尾 ()

返回 ()

编辑框 1.内容 = 到文本 (a)

运行程序得不到结果！！

因为在内容输出到编辑框 1 前，就返回了，所以得不到结果！

假如：把程序改动一下：

版本 2

.子程序 __启动窗口_创建完毕

.局部变量 a, 整数型

.计次循环首 (5, a)

 a = a + 1

 .计次循环尾 ()

编辑框 1.内容 = 到文本 (a)

返回 ()

此时运行结果为：6

1.11 结束命令

操作系统支持：Windows、Linux、Unix 所属类别：流程控制

本命令结束当前程序的运行。

语法：无返回值 结束 ()

使用：.版本 2

.子程序 __启动窗口_创建完毕

.局部变量 a, 整数型

.计次循环首 (5, a)

 a = a + 1

.计次循环尾 ()

编辑框 1.内容 = 到文本 (a)

结束 ()

| | |
|--------------------------|----|
| 二. 算术运算..... | 8 |
| 2.1 相乘 命令..... | 8 |
| 2.2 相加 命令..... | 9 |
| 2.3 相除 命令..... | 9 |
| 2.4 相减命令..... | 10 |
| 2.5 整除 命令..... | 10 |
| 2.6 求余数 命令..... | 11 |
| 2.7 负 命令..... | 11 |
| 2.8 取符号 命令..... | 12 |
| 2.9 取绝对值 命令..... | 12 |
| 2.10 取整/取绝对整 命令..... | 13 |
| 2.10.1 取整..... | 13 |
| 2.10.2 绝对取整 命令..... | 13 |
| 2.11 四舍五入 命令..... | 14 |
| 2.12 求次方 命令..... | 14 |
| 2.13 求平方根 命令..... | 15 |
| 2.14 求正弦/余弦 命令..... | 15 |
| 2.15 求正切/反正切 命令..... | 16 |
| 2.16 求自然对数 命令..... | 17 |
| 2.17 求反对数 命令..... | 17 |
| 2.18 是否运算正确 命令..... | 18 |
| 2.19 置随机数种子/取随机数 命令..... | 18 |

二. 算术运算

2.1 相乘 命令

操作系统支持: Windows、Linux、Unix 所属类别: [算术运算](#)

返回两个数值的乘积, 运算符号为“*”或“×”。

语法: 双精度小数型 相乘 (被乘数, 乘数, ...)

| 参数名 | 描述 |
|-----|--------------------|
| 被乘数 | 必需的; 双精度小数型。 |
| 乘数 | 必需的; 可扩充的; 双精度小数型。 |

使用: .版本 2

.子程序 _按钮_1_被单击

.局部变量 被乘数, 整数型, ,, 被乘数 1-9

.局部变量 乘数, 整数型, ,, 乘数 1-9

.局部变量 要显示的内容, 文本型

.计次循环首 (9, 被乘数)

.变量循环首 (1, 被乘数, 1, 乘数)

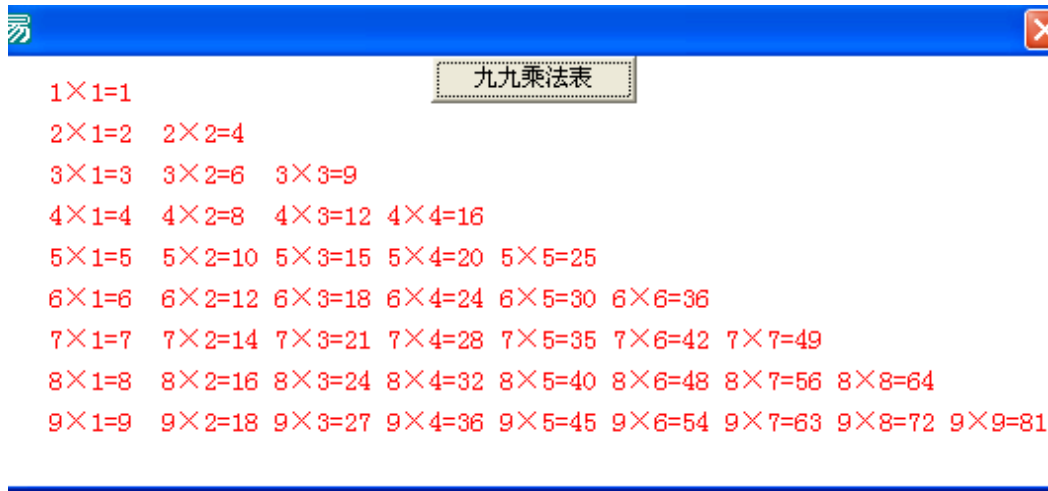
要显示的内容 = 到文本 (被乘数) + “×” + 到文本 (乘数) + “=” + 到文本 (被乘数 × 乘数)

画板 1.定位写出 (乘数 × 57 - 30, 被乘数 × 21 - 10, 要显示的内容)

.变量循环尾 ()

.计次循环尾 ()

程序运行结果显示:



2.2 相加 命令

操作系统支持：Windows、Linux、Unix 所属类别：[算术运算](#)

运算符号为“+”，用途为：1、返回两个数值的和；2、将两个文本首尾连接起来，返回连接后的文本；3、将两个字节集首尾连接起来，返回连接后的字节集。

语法：通用型 相加 （被加数或文本或字节集， 加数或文本或字节集， ... ）

| 参数名 | 描述 |
|------------|-----------------|
| 被加数或文本或字节集 | 必需的； 通用型。 |
| 加数或文本或字节集 | 必需的； 可扩充的； 通用型。 |

使用：.版本 2

.子程序 __启动窗口_创建完毕

.局部变量 B, 双精度小数型

.局部变量 A, 整数型

A = 1

B = 3

编辑框 1.内容 = 到文本 (A + B)

结果为：4

2.3 相除 命令

操作系统支持：Windows、Linux、Unix 所属类别：[算术运算](#)

返回两个数值的商，运算符号为“/”或“÷”。

语法：双精度小数型 相除 （被除数， 除数， ... ）

| 参数名 | 描述 |
|-----|--------------------|
| 被除数 | 必需的； 双精度小数型。 |
| 除数 | 必需的； 可扩充的； 双精度小数型。 |

使用：.版本 2

.子程序 __启动窗口_创建完毕

.局部变量 B, 双精度小数型

.局部变量 A, 整数型

A = 1

B = 3

编辑框 1.内容 = 到文本 (A/B)

结果为：0.3333333

2.4 相减命令

操作系统支持：Windows、Linux、Unix 所属类别：[算术运算](#)

返回两个数值的差，运算符号为“-”

语法：双精度小数型 相减 （被减数， 减数， ... ）

| 参数名 | 描述 |
|-----|--------------------|
| 被减数 | 必需的； 双精度小数型。 |
| 减数 | 必需的； 可扩充的； 双精度小数型。 |

使用：.版本 2

.子程序 __启动窗口_创建完毕

.局部变量 A, 双精度小数型

.局部变量 B, 双精度小数型

A = 1.3

B = 3

编辑框 1.内容 = 到文本 (B - A)

结果为：1.7

2.5 整除 命令

操作系统支持：Windows、Linux 所属类别：[算术运算](#)

求出两个数值的商，并返回其整数部分，运算符号为“\”

语法：双精度小数型 整除 （被除数， 除数， ... ）

| 参数名 | 描述 |
|-----|--------------------|
| 被除数 | 必需的； 双精度小数型。 |
| 除数 | 必需的； 可扩充的； 双精度小数型。 |

使用：.版本 2

.子程序 __启动窗口_创建完毕

.局部变量 B, 双精度小数型

.局部变量 A, 双精度小数型

A = 1.1

B = 3.6

编辑框 1.内容 = 到文本 (B \ A)

结果为：3

2.6 求余数 命令

操作系统支持：Windows、Linux 所属类别：[算术运算](#)

求出两个数值的商，并返回余数部分，运算符号为“%” 或 “Mod”

语法： 双精度小数型 求余数 （被除数， 除数， ... ）

| 参数名 | 描述 |
|-----|--------------------|
| 被除数 | 必需的； 双精度小数型。 |
| 除数 | 必需的； 可扩充的； 双精度小数型。 |

使用：.版本 2

.子程序 __启动窗口_创建完毕

.局部变量 B, 双精度小数型

.局部变量 A, 双精度小数型

A = 1.1

B = 3

编辑框 1.内容 = 到文本 (B % A)

运行结果：0.8

2.7 负 命令

操作系统支持：Windows、Linux、Unix 所属类别：[算术运算](#)

反转一个数值的正负符号并返回反转后的值，运算符号为“-”（放在表达式首）。

语法： 双精度小数型 负 （数值）

| 参数名 | 描述 |
|-----|--------------|
| 数值 | 必需的； 双精度小数型。 |

使用：.版本 2

.子程序 __启动窗口_创建完毕

.局部变量 B

B = 3

编辑框 1.内容 = 到文本 (-B)

运行结果：-3

2.8 取符号 命令

操作系统支持：Windows、Linux 所属类别：[算术运算](#)

返回一个整数，如果小于零，表明给定数值为负；如果等于零，表明给定数值为零；如果大于零，表明给定数值为正。

语法： 整数型 取符号 （欲取其符号的数值）

| 参数名 | 描述 |
|----------|--------------|
| 欲取其符号的数值 | 必需的； 双精度小数型。 |

使用：.版本 2

.子程序 __启动窗口_创建完毕

.局部变量 B, 整数型

B = 9

编辑框 1.内容 = 到文本 (取符号 (B))

运行结果：1（表示所取的值为正数）

假如：B=-6 呢？（运行结果为：-1 为什么??）请读者自己理解！

B=0 呢？（运行结果为：0 为什么??）请读者自己理解！

2.9 取绝对值 命令

操作系统支持：Windows、Linux 所属类别：[算术运算](#)

如果所提供数值为字节型，则将直接返回该数值。

语法： 双精度小数型 取绝对值 （欲取其绝对值的数值）

| 参数名 | 描述 |
|-----------|--------------|
| 欲取其绝对值的数值 | 必需的； 双精度小数型。 |

使用：.版本 2

.子程序 __启动窗口_创建完毕

.局部变量 B, 整数型

B = -9

编辑框 1.内容 = 到文本 (取绝对值 (B))

运行结果：9

2.10 取整/取绝对整 命令

2.10.1 取整

操作系统支持：Windows、Linux 所属类别：[算术运算](#)

返回一个小数的整数部分。本命令与“绝对取整”命令不相同之处为：如果给定小数为负数，则本命令返回小于或等于该小数的第一个负整数，而“绝对取整”命令则会返回大于或等于该小数的第一个负整数。

例如，本命令将 **-7.8** 转换成 **-8**，而“绝对取整”命令将 **-7.8** 转换成 **-7**。

语法： 整数型 取整 （欲取整的小数）

| 参数名 | 描述 |
|--------|--------------|
| 欲取整的小数 | 必需的； 双精度小数型。 |

2.10.2 绝对取整 命令

操作系统支持：Windows、Linux 所属类别：[算术运算](#)

返回一个小数的整数部分。本命令与“取整”命令不相同之处为：如果给定小数为负数，则本命令返回大于或等于该小数的第一个负整数，而“取整”命令则会返回小于或等于该小数的第一个负整数。例如，本命令将 **-7.8** 转换成 **-7**，而“取整”命令将 **-7.8** 转换成 **-8**。

语法： 整数型 绝对取整 （欲取整的小数）

| 参数名 | 描述 |
|--------|--------------|
| 欲取整的小数 | 必需的； 双精度小数型。 |

使用：.版本 2

.子程序 __启动窗口_创建完毕

.局部变量 B, 双精度小数型

B = -9.65

编辑框 1.内容 = 到文本 (取整 (B))

编辑框 2.内容 = 到文本 (绝对取整 (B))

运行结果：-10 -9

2.11 四舍五入 命令

操作系统支持：Windows、Linux 所属类别：[算术运算](#)

返回按照指定的方式进行四舍五入运算的结果数值。

语法： 双精度小数型 四舍五入 (欲被四舍五入的数值, [被舍入的位置])

| 参数名 | 描述 |
|-----------|--|
| 欲被四舍五入的数值 | 必需的； 双精度小数型。 |
| 被舍入的位置 | 可选的； 整数型。如果大于 0，表示小数点右边应保留的位数；如果等于 0，表示舍入到整数；如果小于 0，表示小数点左边舍入到的位置。例如：四舍五入 (1056.65,1) 返回 1056.7； 四舍五入 (1056.65,0) 返回 1057； 四舍五入 (1056.65,-1) 返回 1060。如果省略本参数，则默认为 0。 |

使用：.版本 2

.子程序 __启动窗口_创建完毕

.局部变量 B, 双精度小数型

B = -9.65

编辑框 1.内容 = 到文本 (四舍五入 (B,1))

*被赋值的变量或变量数组：编辑框 1.内容

*用作赋予的值或资源：到文本 (四舍五入 (B, 1))

*待转换的数组：四舍五入 (B, 1)

*欲被四舍五入的数值：B

*被四舍五入的位置：1 (正数表示四舍五入小数点后几位)
(负数表示四舍五入小数点前几位)

运行结果：-9.7

2.12 求次方 命令

操作系统支持：Windows、Linux 所属类别：[算术运算](#)

返回指定数值的指定次方。

语法： 双精度小数型 求次方 (欲求次方数值, 次方数)

| 参数名 | 描述 |
|--------|--------------------------------|
| 欲求次方数值 | 必需的； 双精度小数型。参数值指定欲求其某次方的数值。 |
| 次方数 | 必需的； 双精度小数型。参数值指定对欲求次方数值的运算指数。 |

使用：.版本 2

.子程序 __启动窗口_创建完毕

.局部变量 B, 双精度小数型

B = 4

编辑框 1.内容 = 到文本 (求次方 (B, -1))

运行结果：0.25

2.13 求平方根 命令

操作系统支持：Windows、Linux 所属类别：[算术运算](#)

返回指定参数的平方根。

语法： 双精度小数型 求平方根 （欲求其平方根的数值）

| 参数名 | 描述 |
|-----------|------------------------------|
| 欲求其平方根的数值 | 必需的； 双精度小数型。参数值如果小于零将导致计算错误。 |

使用：.版本 2

.子程序 __启动窗口_创建完毕

.局部变量 B, 双精度小数型

B = 4

编辑框 1.内容 = 到文本 (求平方根 (B))

运行结果：2

2.14 求正弦/余弦 命令

操作系统支持：Windows、Linux 所属类别：[算术运算](#)

返回指定角的正弦值。

语法： 双精度小数型 求正弦/求余弦 （欲进行计算的角）

| 参数名 | 描述 |
|---------|---|
| 欲进行计算的角 | 必需的； 双精度小数型。所使用单位为弧度。为了将角度转换成弧度，请将角度乘以 $\pi/180$ 。为了将弧度转换成角度，请将弧度乘以 $180/\pi$ 。如果参数值大于等于 2 的 63 次方，或者小于等于 -2 的 63 次方，将导致计算溢出。 |

使用：.版本 2

.程序集 窗口程序集 1

.子程序 __启动窗口_创建完毕

.局部变量 B, 双精度小数型

$B = \#pi \div 6$

编辑框 1.内容 = 到文本 (求正弦 (B))

编辑框 2.内容 = 到文本 (求余弦 (B))

运行结果：0.49999987≈0.5 0.8660254037919

2.15 求正切/反正切 命令

操作系统支持：Windows、Linux 所属类别：[算术运算](#)

返回指定角的正切值。

语法： 双精度小数型 求正切 （欲进行计算的角）

语法： 双精度小数型 求反正切 （欲求其反正切值的数值）

| 参数名 | 描述 |
|---------|---|
| 欲进行计算的角 | 必需的； 双精度小数型。所使用单位为弧度。为了将角度转换成弧度，请将角度乘以 $\#pi / 180$ 。为了将弧度转换成角度，请将弧度乘以 $180 / \#pi$ 。如果参数值大于等于 2 的 63 次方，或者小于等于 -2 的 63 次方，将导致计算溢出。 |

使用：.版本 2

.程序集 窗口程序集 1

.子程序 __启动窗口_创建完毕

.局部变量 C, 双精度小数型

.局部变量 B, 双精度小数型

$B = \#pi \div 180 \times 45$

$C = 1$

编辑框 1.内容 = 到文本 (求正切 (B))

编辑框 2.内容 = 到文本 (求反正切 (C) × 180)

运行结果：0.9999999999551 141.3716694115

2.16 求自然对数 命令

操作系统支持：Windows、Linux 所属类别：[算术运算](#)

返回指定参数的自然对数值。自然对数是以 e 为底的对数。常量 $\#e$ 的值大约是 2.718282。

语法：双精度小数型 求自然对数 （欲求其自然对数的数值）

| 参数名 | 描述 |
|------------|--------------|
| 欲求其自然对数的数值 | 必需的； 双精度小数型。 |

使用：.版本 2

.程序集 窗口程序集 1

.子程序 __启动窗口_创建完毕

.局部变量 B, 双精度小数型

$B = \#e$ ‘(常数 e)

编辑框 1.内容 = 到文本 (求自然对数 (B))

运行结果：1.000000063106 \approx 1

2.17 求反对数 命令

操作系统支持：Windows、Linux 所属类别：[算术运算](#)

返回 e （自然对数的底）的某次方。

语法：双精度小数型 求反对数 （欲求其反对数的数值）

| 参数名 | 描述 |
|-----------|--|
| 欲求其反对数的数值 | 必需的； 双精度小数型。如果参数值超过 709.782712893，将导致计算溢出。 |

.版本 2

.程序集 窗口程序集 1

.子程序 __启动窗口_创建完毕

.局部变量 B, 双精度小数型

$B = 1$

编辑框 1.内容 = 到文本 (求反对数 (B)) ‘编辑框 1.内容= e^B

运行结果：2.718281828459= e

2.18 是否运算正确 命令

操作系统支持：Windows、Linux 所属类别：[算术运算](#)

对乘、除、“求次方”、“求平方根”、“求正弦值”、“求余弦值”、“求正切值”、“求反正切值”、“求自然对数”、“求反对数”等等数学运算命令所计算出来的双精度结果数值进行校验，如果该数值正确有效，返回真。如果该数值是运算错误或运算溢出后的结果，返回假。

语法：逻辑型 是否运算正确 （欲校验的计算结果）

| 参数名 | 描述 |
|----------|--------------|
| 欲校验的计算结果 | 必需的； 双精度小数型。 |

使用：.版本 2

.程序集 窗口程序集 1

.子程序 __启动窗口_创建完毕
.局部变量 数 1, 双精度小数型
.局部变量 数 2, 双精度小数型
.局部变量 数 3, 双精度小数型

数 1 = #pi

数 2 = 1.41421

数 3 = 数 1 × 数 2

_启动窗口.标题 = 到文本 (数 3)

信息框 (是否运算正确 (数 3), 0,)

运行结果：先信息框提示：真 然后窗口标题显示为数 3 的值

2.19 置随机数种子/取随机数 命令

操作系统支持：Windows、Linux 所属类别：[算术运算](#)

置随机数种子:为随机数生成器初始化一个种子值，不同的种子值将导致“取随机数”命令返回不同的随机数系列。

语法：无返回值 置随机数种子 （[欲置入的种子数值]）

取随机数: 返回一个指定范围内的随机数值。在使用本命令取一系列的随机数之前，应该先使用“置随机数种子”命令为随机数生成器初始化一个种子值。

语法：整数型 取随机数 （ [欲取随机数的最小值] ， [欲取随机数的最大值] ）

| 参数名 | 描述 |
|----------|------------------------------------|
| 欲置入的种子数值 | 可选的； 整数型。如果省略本参数，将默认使用当前计算机系统的时钟值。 |

使用：.版本 2

.程序集 窗口程序集 1

.子程序 __启动窗口_创建完毕

.子程序 _按钮 1_被单击

置随机数种子 ()

标签 1.标题 = 到文本 (取随机数 (1, 1000))

运行结果：随机显示 1~1000 之间的数值

三.逻辑比较

3.1 等于 命令

操作系统支持：Windows、Linux、Unix 所属类别：[逻辑比较](#)

被比较值与比较值相同时返回真，否则返回假，运算符号为“=”或“==”。

语法： 逻辑型 等于 (被比较值, 比较值)

| 参数名 | 描述 |
|------|--------------------------------------|
| 被比较值 | 必需的； 通用型。 |
| 比较值 | 必需的；通用型。比较值的数据类型必须与“被比较值”一致或者可以相互转换。 |

使用：.版本 2

.程序集 窗口程序集 1

.子程序 __启动窗口_创建完毕

.局部变量 A, 整数型

A = 2

编辑框 1.内容 = 到文本 (A + 1)

运行结果：3

3.2 不等于 命令

操作系统支持：Windows、Linux、Unix 所属类别：[逻辑比较](#)

被比较值与比较值不相同返回真，否则返回假，运算符号为“<>”或“!=”或“≠”。

语法：逻辑型 不等于 （被比较值， 比较值）

| 参数名 | 描述 |
|------|--------------------------------------|
| 被比较值 | 必需的；通用型。 |
| 比较值 | 必需的；通用型。比较值的数据类型必须与“被比较值”一致或者可以相互转换。 |

使用：.版本 2

.程序集 窗口程序集 1

.子程序 __启动窗口_创建完毕

.子程序 _按钮 1_被单击

.如果 (编辑框 1.内容 ≠ 编辑框 2.内容 且 编辑框 1.内容 ≠ 编辑框 3.内容)

信息框 (“请重新输入!”, 0,)

运行结果：如果编辑框 1, 2, 3 里的内容不一样，且不满足 如果 命令，所得结果都是得出提示信息框，反之则不提示信息框

3.3 小于/大于 命令

操作系统支持：Windows、Linux、Unix 所属类别：[逻辑比较](#)

小于命令

被比较值小于比较值时返回真，否则返回假，运算符号为“<”。

语法：逻辑型 小于 （被比较值， 比较值）

大于命令

被比较值大于比较值时返回真，否则返回假，运算符号为“>”。

语法：逻辑型 大于 （被比较值， 比较值）

| 参数名 | 描述 |
|------|--|
| 被比较值 | 必需的；通用型。被比较值的数据类型可以为数值型、日期时间型、文本型。 |
| 比较值 | 必需的；通用型。比较值的数据类型可以为数值型、日期时间型、文本型，且必须与“被比较值”一致或者可以相互转换。 |

使用：.版本 2

.程序集 窗口程序集 1

.子程序 __启动窗口_创建完毕

.子程序 _按钮 1_被单击

.如果 (编辑框 1.内容 < 编辑框 2.内容)

 信息框 (到数值 (编辑框 1.内容), 0,)

.否则

.如果结束

.如果 (编辑框 3.内容 > 编辑框 2.内容)

 信息框 (到数值 (编辑框 3.内容), 0,)

运行结果：看 3 个编辑框内的数字大小，如果符合程序条件的，都有信息框提示，不符合就没信息框提示。

3.4 小于或等于/大于等于 命令

操作系统支持：Windows、Linux、Unix 所属类别：[逻辑比较](#)

小于或等于

被比较值小于或等于比较值时返回真，否则返回假，运算符号为“<=”或“≤”。

语法： 逻辑型 小于或等于 (被比较值, 比较值)

大于等于

被比较值大于或等于比较值时返回真，否则返回假，运算符号为“>=”或“≥”。

语法： 逻辑型 大于或等于 (被比较值, 比较值)

| 参数名 | 描述 |
|------|--|
| 被比较值 | 必需的；通用型。被比较值的数据类型可以为数值型、日期时间型、文本型。 |
| 比较值 | 必需的；通用型。比较值的数据类型可以为数值型、日期时间型、文本型，且必须与“被比较值”一致或者可以相互转换。 |

使用：.版本 2

.程序集 窗口程序集 1

.子程序 __启动窗口_创建完毕

.子程序 _按钮 1_被单击

.如果 (编辑框 1.内容 ≤ 编辑框 2.内容)

 信息框 (到数值 (编辑框 1.内容), 0,)

.否则

.如果结束

.如果 (编辑框 3.内容 ≥ 编辑框 2.内容)

 信息框 (到数值 (编辑框 3.内容), 0,)

运行结果：看 3 个编辑框内的数字大小，如果符合程序条件的，都有信息框提示，不符合就没信息框提示。

3.5 近似等于 命令

操作系统支持：Windows、Linux 所属类别：[逻辑比较](#)

当比较文本在被比较文本的首部被包容时返回真，否则返回假，运算符号为“?”或“≈”。

语法： 逻辑型 近似等于 (被比较文本, 比较文本)

| 参数名 | 描述 |
|-------|-----------|
| 被比较文本 | 必需的： 文本型。 |
| 比较文本 | 必需的： 文本型。 |

使用：

3.6 并且 命令

操作系统支持：Windows、Linux、Unix 所属类别：[逻辑比较](#)

如果所提供的两个参数逻辑值都为真则返回真，否则返回假，运算符号为“&&”或“And”或“且”。

语法： 逻辑型 并且 (逻辑值一, 逻辑值二, ...)

| 参数名 | 描述 |
|------|-----------------|
| 逻辑值一 | 必需的； 逻辑型。 |
| 逻辑值二 | 必需的； 可扩充的； 逻辑型。 |

使用：.版本 2

.程序集 窗口程序集 1

.子程序 __启动窗口_创建完毕

.子程序 _按钮 1_被单击

.如果 (编辑框 1.内容 = “我喜欢你!” 且 编辑框 2.内容 = “你喜欢我吗?”)

 信息框 (“I Love You!”, 0,)

 编辑框 3.内容 = “I Love You!”

运行结果：当在编辑框 1 中输入“我喜欢你!”，在编辑框 2 中输入数字“你喜欢我吗?”时，出现提示框“I Love You!” 编辑框 3 中就会出现“I Love You!”。

3.7 或者 命令

操作系统支持：Windows、Linux、Unix 所属类别：[逻辑比较](#)

如果所提供的两个参数逻辑值中任意有一个为真则返回真，否则返回假，运算符为“||”或“Or”或“或”。

语法： 逻辑型 或者 (逻辑值一, 逻辑值二, ...)

| 参数名 | 描述 |
|------|-----------------|
| 逻辑值一 | 必需的; 逻辑型。 |
| 逻辑值二 | 必需的; 可扩充的; 逻辑型。 |

使用：.版本 2

.程序集 窗口程序集 1

.子程序 __启动窗口_创建完毕

.子程序 _按钮 1_被单击

.如果 (编辑框 1.内容 = “” 或 编辑框 2.内容 = “”)

信息框 (“请输入账号和密码!”, 0,)

运行结果:

当编辑框 1 以及编辑框 2 中有一个没数据, 则会提示 “请输入账号跟密码!”

当编辑框 1 以及编辑框 2 中两个都有数据, 则不会出现提示!

3.8 取反 命令

操作系统支持：Windows、Linux、Unix 所属类别：[逻辑比较](#)

如果参数值为真则返回假, 如果参数值为假则返回真。

语法： 逻辑型 取反 (被反转的逻辑值)

| 参数名 | 描述 |
|---------|-----------|
| 被反转的逻辑值 | 必需的; 逻辑型。 |

使用：.版本 2

.程序集 窗口程序集 1

.子程序 _按钮 1_被单击

.局部变量 A, 逻辑型

A = 取反 (2 ≠ 3)

编辑框 1.内容 = 到文本 (A)

运行结果:

编辑框 1 中显示的内容为: 假

因为 $2 \neq 3$, 这个逻辑是真的, 即逻辑=真, 取反的话, 得到的逻辑结果为假。

四.位运算

4.1 位取反 命令

操作系统支持: Windows、Linux 所属类别: [位运算](#)

将指定数值每一个比特位的值取反后返回。

语法: 整数型 位取反 (欲取反的数值)

| 参数名 | 描述 |
|--------|-----------|
| 欲取反的数值 | 必需的; 整数型。 |

使用: .版本 2

.程序集 窗口程序集 1

.子程序 _按钮 1_被单击

.局部变量 A, 整数型

A =004324

编辑框 1.内容 = 到文本 (位取反 (A))

运行结果:

-4325

位取反 简单算法 位取反 (A) =-A-1

4.2 位与 命令

操作系统支持: Windows、Linux 所属类别: [位运算](#)

如两个数值有某共同比特位为 1, 则返回值的对应位也为 1, 否则为 0。返回计算后的结果。

语法: 整数型 位与 (位运算数值一, 位运算数值二, ...)

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。
如要下载或阅读全文，请访问：

<https://d.book118.com/845043134122011240>