

迅投 QMT-VBA 模型编辑 使用手册

北京睿智融科控股股份有限公司

二〇一九年二月

[illegible]

目 录

1. 概述.....	1
2. 详述.....	1
2.1. 模型的编写规则.....	1
2.1.1. 数据的引用.....	3
2.1.2. 特殊数据引用.....	3
2.1.3. 公式体构成结构.....	5
2.2. 迅投极速策略交易系统的控制语句.....	7
2.2.1. 序列变量与数组.....	7
2.2.2. 循环语句.....	9
2.2.3. 条件语句.....	13
2.3. 序列模式和逐 K 线模式.....	16
2.3.1. 控制语句在两种不同模式下的运行特点.....	16
2.3.2. 关于模型运行时这两种模式的选择.....	18
2.4. 交易系统函数.....	19
2.4.1. 下单函数.....	19
2.4.2. 监控函数.....	32
2.4.3. 控制函数.....	33
2.4.4. 线型描述.....	33
2.4.5. 逻辑函数.....	35
2.4.6. 动态行情.....	38
2.4.7. 绘图函数.....	38
2.4.8. 字符串函数.....	41
2.4.9. 引用函数.....	46
2.4.10. 控制函数.....	55
2.4.11. 指标函数.....	58
2.4.12. 统计函数.....	58
2.4.13. 时间函数.....	67
2.4.14. 数学函数.....	71
2.4.15. 行情函数.....	75
2.4.16. 扩展数据.....	79
2.4.17. 组合模型.....	80
2.4.18. 交易函数.....	84
2.4.19. 系统函数.....	96
2.4.20. 附加函数.....	99

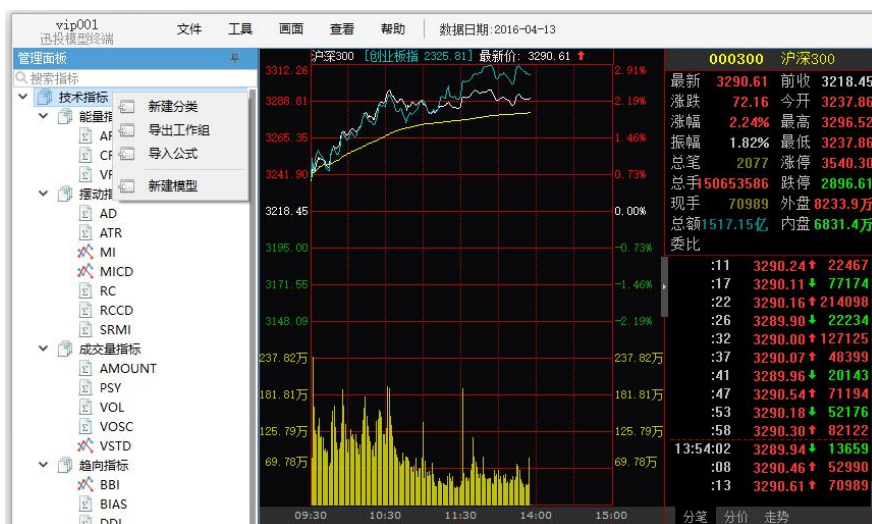
1. 概述

本教程主要介绍迅投极速策略交易系统的公式编写系统，重点介绍模公式系统的编写规则、序列模式和逐 K 线模式下的运行原理及特点。本篇教程的读者需要有一定的 VBA 语言编写经验。

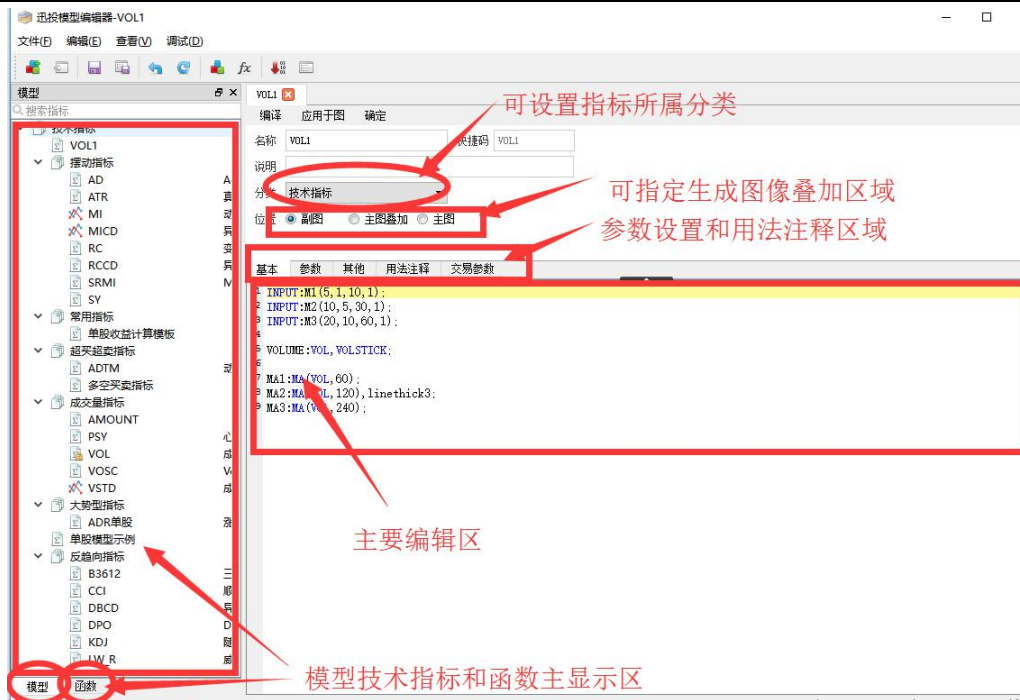
2. 详述

2.1. 模型的编写规则

在系统面板左侧模型树中右键，可对模型策略进行新建，编辑和导入等操作。



选“新建模型”，出现下图模型编辑器主界面。



通过该界面我们可以了解该系统的公式设定的内容和相关规则：

(1) 每一个指标公式必须有一个名称，这个名称由中文、字母和数字组成，公式名称在同类公式中必须是唯一的；公式描述是一段文字，用来简单描述该公式的含义，在公式列表时显示这段文字，这段文字不宜过长；同时该界面也定义了该指标显示的位置，是在主图上与K线叠加还是显示在副图上，一般来讲，只有少数几个主图指标会设定为主图叠加，例如MA均线、BOLL线等。

(2) 参数：计算参数用来替代公式中所需要的常数，在使用时可以方便地调节参数，不必修改公式就可以对计算方法进行调节。参数包括参数参数名、最小值、最大值、缺省值、步长五个部分，参数名称用于标识参数，计算公式时采用缺省值计算，而最小值和最大值是参数的调整范围。

(3) 其他：包括加密、快速计算、刷新闻隔等内容。模型终端支持公式加密功能和凭密码导出公式功能，如果您不想让别人看到您的公式内容，可以通过此功能对公式进行加密。

(4) 用法注释：模型注释是一段文字，相对于模型描述而言它可以很长，主要用来模型一个公式如何使用、注意事项、计算方法等等。

(5) 交易参数：主要用于在交易中，单股模型通过接口单直接下单的交易参数设置。

所有的公式系统都是遵守统一的运算法则，统一的格式进行函数之间的计算，所以我们掌握了技术指标公式的基本原理，其他的公式也不会出脱其外。

例如我们在指标公式系统内写下公式：



A: =X+Y; B: =A/Z; C: =B*0.618;

分析以上公式，我们可以引出以下相关的格式和法则的结论：

2.1.1. 数据的引用

(1) 数据来源

公式中的基本数据来源于接收的每日行情数据，这些数据有行情函数从数据库中按照一定的方式提取，例如，高开低收，成交量，成交额等等。

(2) 数据类型

按照公式使用的数据类型，系统可以处理的数据分为两类：变量和常量。

所谓变量就是一个随着时间变化而变化的数据，例如成交量；常量就是一个永远不变的数据。例如3，每个函数需要的参数可能是变量也可能是常量，不能随便乱用，函数计算的结果一般是一个变量。

例如计算收盘价均线MA(CLOSE, 5)，MA函数要求第一个参数为变量，而CLOSE函数返回的正好是一个变量；MA函数要求的第二个参数是常量，5就是一个常量，所以我们就不能这样书写：MA(5, CLOSE)。

2.1.2. 特殊数据引用

(1) 指标数据引用

经常地编制公式的过程当中，需要使用另外一个指标的值，如果按照通常的做法，重新编写过这个指标显得很麻烦，因此有必要学习使用如何调用别的指标公式。

基本格式为：“指标.指标线”（参数）

a、指标和指标线之间用顿号分开，一个指标不一定只有一条指标线，所以有必要在指标后标注指标线的名称，但是如果缺失则表示引用最后一条指标线。

b、参数在表达式的末尾，必须用括号括起来，参数之间用逗号分开，通过参数设置可以选择设定该指标的参数，如果参数缺失则表示使用该指标的默认参数设置。

c、整个表达式用引号引在其中，除参数以外。

例如：

“MACD.DEF”（26, 12, 9）：表示计算MACD指标的DEA指标线，计算参数为26、12、9；

“MACD”（26, 12, 9）：表示该指标的最后一条指标线，计算参数是26、12、9；

“MACD”：表示该指标的最后一条指标线并且使用公式的默认参数。



(2) 跨周期引用指标数据

在迅投极速策略交易系统决策交易系统当中允许使用不同分析周期上的指标数据，并且支持与自身长短不同的任意周期引用。

a、基本格式为：“指标. 指标线#周期”（参数），格式上只是比上面指标引用多了一个周期设定，其他内容和方法一样，在周期调用上存在以下对应关系：

MIN1：1分钟 MIN5：5分钟.....DAY：日线 WEEK：周线 MONTH：月线 YEAR：年线

如上所示，MIN1表示的分析周期为1分钟，其它依次类推。例如：当前周期为日线，那么在公式中使用“MACD.DEA#WEEK”（26, 12, 9）表示使用了当天所在的本周的MACD指标中的数据。

b、以上格式的扩展格式为：“指标. 指标线##周期”（参数），该格式比基本格式采用了不同的对齐方式，简而言之，就是说“#”的格式调用的本周期所在的上一级周期的指标数据，那么“##”的格式则表示调用了前一种格式的前一周期的指标数据，举上例而言，

“MACD.DEF##WEEK”（26, 12, 9）表示的是从当天看来的上一周的数据，而基本格式就是当天看来的本周的数据。

注：在用户翻看一个品种时极速策略交易系统是可以自动补数据的，但是无法自动补被引用品种的或者该品种不同周期的数据，在首次使用迅投极速策略交易系统或者在不不确定被引用数据是否齐全时，请手工进行数据补充工作，手工补充数据方法在《迅投极速策略交易系统使用说明》中第二章数据管理中有详细说明，此处不再做过多解释。

此处，为了让您能更好的理解关于跨周期引用指标数据，我们添加了一个关于跨周期引用指标基本格式以及如何使用“##”来引用其他指标周期问题的示例。

跨周期引用指标基本格式----“指标. 指标线#周期”（参数）；

跨周期引用指标扩展格式----“指标. 指标线##周期”（参数）；

例1.1：

第一步：新建一个指标，命名为H，在H中写入下面这句代码

H1:high;

L1:low;

第二步：新建第二个指标，命名为TEST，在TEST中写入下面代码：

preDayHigh:“H.H1##day”; //昨日最高价

preDayLOW:“H.L1##day”; //昨日最低价



第三步：将指标“TEST”运行应用于盘面，您将在K线图上看到该品种昨日最高价和昨日最低价。

迅投模型编辑器中还提供了STKINDI函数——引用任意品种任意周期的任意指标输出，具体请参考模型编辑器自带的函数列表里的函数说明。

（3）其他数据引用

使用以下的格式可以在当前的分析界面下引用大盘的数据或者其他个股的数据实现横向上的对比。

引用个股数据引用个股数据时使用下列格式：“品种代码\$数据”，在以上格式当中调用CLOSE，VOL，AMOUNT等等！例如“SZ000002\$VOL”表示000002该股本周期的成交量，“SH000001\$CLOSE”同样也可以表示为大盘本周期的收盘价，此时的大盘被视为一只个股。

迅投模型编辑器还提供了CALLSTOCK函数——可引用其他证券或合约的部分基础数据，具体请参考模型编辑器自带的函数列表里的函数说明。

2.1.3. 公式体构成结构

A、公式语句

所有的公式体由若干语句按照一定的格式组成，每个语句表示一个计算结果，根据各个语句的功能分为两大类语句，一类是赋值语句，一类是中间表达式。

B、赋值语句

在技术指标“B:A/Z”和“C:B*0.618”就是分别两条指标线，语言间用冒号隔开。该语句被称为赋值语句，在技术指标当中，赋值语句的计算结果将会被计算机执行并形成相应的图形。每个语句可以有一个名称，该名称写在语句的最前面，并用一个冒号将它与语句分隔开。例如：ST:MA(CLOSE, 5)；表示该语句求收盘价的五日均线，语句的名称为ST，在该语句后的语句中可以直接用ST来替代MA(CLOSE, 5)，例如：MA(ST, 5)表示对收盘价的五日均线再求五日平均。

C、中间语句

一个语句如果不需要显示，可以将它定义为中间语句，例如在上例当中的第一句“A:=X+Y;”，这样该语句就不会被系统辨认为是指标线了，中间语句用“:=”替代冒号，其他与一般语句完全一样，使用中间语句可以有效降低公式的书写难度，还可以将需要重复使用的语句定义成中间语句以减少计算量。在每个模型中，中间公式数量没有限制，但是所有语

句之间需要使用分号隔开。

D、公式计算符

公式计算符将函数连接成为公式，计算分为算术计算符和逻辑计算符。

a、算术计算符：

包括+、-、*、/，它们分别对计算符两边的数据进行加减乘除计算，这同一般意义上的算术计算没有差异。

b、逻辑计算符

包括>、<、<>、>=、<=、=、AND、OR八种，分别表示大于、小于、不等于、大于等于、小于等于、等于、逻辑与、逻辑或运算，如果条件成立计算结果就等于1，否则等于0，例如：3+4=7，4>3都等于1，3<=12就等于0，“逻辑与”表示两个条件都成立时结果才成立；“逻辑或”表示两个条件中只要有一个成立结果成立。

例如，4>3 AND 12>=4的结果等于1，4>3 OR 3>12的结果等于1。

注意：运算符的优先级直接决定了表达式执行的先后顺序，用户使用多个算术和逻辑运算符时，要注意运算符的优先级，如果不太确定，最好用括号把紧密的表达式括起来。

E、线形描述符

对于技术指标公式可以在语句加上线形描述符，用来表示如何画该语句描述的指标线。

线形描述符号包括以下7种。描述符写在语句后分号前，用逗号将它们与语句分隔开，例如在上例当中加入一句线形描述符，C: B*0.618, COLORSTICK; 该语句在被执行时，会在图中添加色彩柱线，该功能在编制MACD等指标的时候会显出它的用处。

STICK: 柱状线

COLORSTICK: 彩色柱状线，当值为正时显示红色，否则显示绿色

COLORRED: 为线形色，RED表示红色

COLORBLUE: 为线形色，BLUE表示蓝色

COLORYELLOW: 为线形色，YELLOW表示黄色

VOLSTICK: 成交量柱状线，当股价上涨时显示红色空心柱，否则绿色

LINESTICK: 同时画出柱状线和指标线

LINETHICK: 对线体的粗细作出描述

CROSSDOT: 小叉线

CIRCLEDOT: 小圆圈线

POINTDOT: 小圆点线

a、COLORRED等三个线形描述符还可以自定义颜色，格式为COLOR+“BBGGRR”；BB、GG、RR表示蓝色，绿色和红色的分量，每种颜色的取值范围是00-FF，采用了16进制，例如：MA5:MA(CLOSE, 5)，COLOR00FFFF表示纯红色与纯绿色的混合色；COLOR808000表示淡蓝色和淡绿色的混合色。

b、LINETHICK可以允许对线型的粗细进行自定义的描述，格式“LINETHICK+ (0/7)”：参数的取值范围在0-7之间，“LINETHICK0”表示最细的线，而“LINETHICK7”表示最粗的线。

2.2. 迅投极速策略交易系统的控制语句

迅投模型平台是个强大的脚本执行平台，除了我们在基础教程部分介绍的顺序执行语句外，可以执行带条件分支和循环以及无条件跳转等功能，迅投极速策略交易系统在工作时，由于内部运行机制不同，分为序列模式以及逐K线模式，我们这里先从序列模式介绍公式系统的运行机理，逐K线模式我们在后面会另有介绍。下面我们将逐步向大家介绍如何使用迅投模型中的控制语句编写模型。

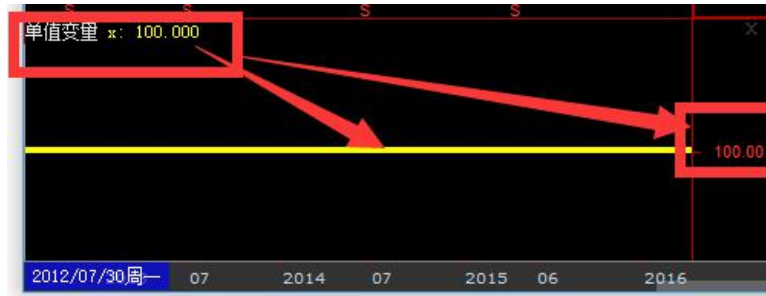
2.2.1. 序列变量与数组

在迅投模型公式系统中，需要大量运用并区分数组、单值变量及序列变量的概念，这些概念也是进一步学习编程所必需的，因此有必要简单描述并初步掌握这些概念。

2.2.1.1. 常数与单值变量

常数，在迅投模型编辑器中，就是不允许改变的数值，在我们平常写公式时，为了公式更加灵活，大量使用了参数，这些参数就是常数，不允许在公式中改变，比如参数n(1, 1, 25)，表示参数n最小值是1、最大值是25、默认值是1，如果在公式中再写一行“n:=30;”就是非法的。

单值变量，即只有一个数值，不随时间而改变，比如“x:100;”，就是定义了一个单值变量x=100，这个值不随时间而改变，做成副图指标看，今天x是100明天x也是100，直到最后1根K线也是100。



2.2.1.2. 数组

系统中，和常规数组概念不同并非构建单纯的有序数据的集合，通过数组可以建立有序的一一对应关系。通过这种对应关系，数组结构可以定义多样的对应关系，如常规数组 $A\{1, 2, 3, 4, 5, \dots\} \sim \{34, 235, 234, 626, 666, \dots\}$ ，从而得到 $A[1]=34, A[2]=235, \dots$ ；也可以定义不同的数据类型下标和数组值来构成对应关系，如例如对于数组 $B\{'SH600000'\ 'SZ000001'\ \dots\} \sim \{'上海'\ '深圳'\ \dots\}$ ，从而有 $B[SH600000]=上海, \dots$

一般来说，数组的使用，要先定义，然后初始化（即赋予数组的每个元素一个初始的数据）。在极速策略交易系统中，目前只支持一维数组，并且下标是从 1 开始的。

在极速策略交易系统中，如何定义、初始化并使用数组？下面简单示例说明：

例 1、在极速策略交易系统中，定义一个含 10 个元素的数值型数组

定义数组的语句是 `variable`，公式代码如下：

`variable:A[10]=0;` //定义一个含 10 个元素的数值型数组 A，并把所有元素初始化为 0；

例 2：定义一个 3 个元素的字符串型的数组

公式代码如下：

`variable:B[3]='abc';` //定义一个含 3 个元素的字符串型数组 B，并把所有元素初始化为 'abc'

例 3：把 {1, 3, 5, 7, 9} 定义为数组

分析：这组数据共有 5 个数值型数据，因此应定义长度为 5 的数组，并给每个元素赋值。公式代码如下：

`variable:A[5]=0;`

`A[1]:=1;`

`A[2]:=3;`



```
A[3]:=5;
```

```
A[4]:=7;
```

```
A[5]:=9;
```

前面所举的关于数组例子，都是概念性的，单纯数组的应用一般都比较复杂，比如，从每天的分笔成交中统计，各个成交价位成交的笔数。等到后面讲完循环有机会时再写具体的代码。

注意：对于使用 for 循环类的函数对数组执行循环赋值时，可以不设置下标顺序，系统会依据数组下标在系统中存储的值的大小顺序依次遍历。

2.1.3 序列变量：

我们平时接触最多的是收盘价 close，写一行代码“fc:close”，就是定义了一个变量 fc，并把收盘价赋值给变量 fc。这里变量 fc 就是序列变量，因为 fc 不是一个值，而是一序列随时间而变化的值(收盘价)，在 K 线图中，从第 1 根 K 线到最后 1 根 K 线都有一个收盘价，因此 fc 跟单值变量不同，是由一系列的值组成的变量，因此我们称之为序列变量。事实上，把“fc:close”这行代码做成一个指标，就可以看出 fc 是一条变化的曲线而不是一条水平直线。

迅投极速策略交易系统公式编辑系统中，可以把序列变量等同于一个数组，这是一个特殊的数组，这个数组的最小下标是从序列变量的起始有效位置开始，数组的最大下标是 K 线数量，其中 K 线的数量，可以从 datacount 函数得到。

比如“fc:close;”，这行代码，可以把 fc 看成是一个下标从 1 开始直到 datacount 的数组。如果我们想知道第 1、2、5、最后 1 根 K 线的收盘价，可以写成如下代码：

代码：

```
fc:=close;//定义一个序列变量，相当于是一个数组
```

```
k1:fc[1];//第 1 根 K 线的收盘价
```

```
k2:fc[2];//第 2 根 K 线的收盘价
```

```
k3:fc[5];//第 5 根 K 线的收盘价
```

```
k_end:fc[datacount];//最后 1 根 K 线的收盘价
```

2.2.2. 循环语句

极速策略交易系统循环语句有两个类别，一个是 FOR TO …，另一个是 REPEAT UNTIL …，这里我们重点介绍 FOR 循环，REPEAT UNTIL 的例子，用户请自行在迅投极速策略交

易系统公式编辑器中软件函数列表中查看使用方法。

2.2.1 FOR递增循环

先从一个的实例开始，求当前股票收盘价的2日算术平均价，为了方便起见，以刚上市不久的603798康普顿来说明，首先定义序列变量fc为收盘价，根据上一讲我们知道，fc可以看成是一个数组，因此，数组fc的每一个元素如下表所示：

603798	康普顿	
	时间	收盘
fc[1]	2016/04/06	20.64
fc[2]	2016/04/07	22.70
fc[3]	2016/04/08	24.97
fc[4]	2016/04/11	27.47
fc[5]	2016/04/12	30.22
fc[6]	2016/04/13	33.24

显然，2日平均收盘价必须是从第2根K线即上市后第2日开始，到最后一天结束，设2日平均股价用序列变量ma2来表示，由于2日平均收盘价等于当日收盘价加昨收盘除以2，如果手工一行一行来写代码计算，则：

```
ma2[2]:=(fc[1]+fc[2])/2; //今收盘与昨收盘之和除以2，等于今2日平均收盘价
ma2[3]:=(fc[2]+fc[3])/2;
ma2[4]:=(fc[3]+fc[4])/2;
ma2[5]:=(fc[4]+fc[5])/2;
ma2[6]:=(fc[5]+fc[6])/2;
```

分析上面代码，2日平均价的计算特点是：

- ① 从第2根K线开始计算的
- ② 每天的均价都等于(今收盘+昨收盘)/2
- ③ 直到最后1根K线结束

容易知道，如果有1000根K线，则要写999行代码，显然效率太低了。

我们来看看规律，如果是中间的某天即第i天，则上面的表达式可以写为

```
ma2[i]:=(fc[i-1]+fc[i])/2;
```



这个i是从第2根K线开始直到最后1根K线。这些计算都是重复同样的计算，只不过i不同罢了。如果有一种方法，可以自动重复计算这些表达式，并且每重复一次，i自动增加1，就可以达到目的了。由此我们定义一种循环语句，可以自动循环、重复运行某一行代码。在迅投模型编辑系统中，循环语句有2种，下面先介绍FOR循环语句

语法：

```
FOR var=n1 TO n2 DO expr;
```

{从 var=n1 开始，直到 var=n2，开始循环执行 expr 语句，每执行一次var加1。这里，var称之为循环变量。}

用循环语句表达上面计算2日平均收盘价，公式代码如下：

```
fc:=close; //定义序列变量fc为收盘价  
for i=2 to datacount do ma2[i] :=(fc[i-1]+fc[i])/2;  
//从i=2到i=datacount循环执行ma2[i] :=(fc[i-1]+fc[i])/2
```

2.2.2.1. FOR 递减循环

上面的FOR循环，循环变量是每次递增1，可称之为递增FOR循环。还有一种FOR循环是递减FOR循环，循环变量是每次递减1，语法如下：

```
FOR var=n1 DOWNT0 n2 DO expr2;
```

{从 var=n1 开始到 var=n2 开始循环执行 expr 语句，每执行一次var减1}

上面我们设计2日平均收盘价的公式时，是从前面往后面计算的，仔细想想，其实也可以从后面往前面计算，公式代码如下：

```
fc:=close;  
for i=datacount downto 2 do ma2[i] :=(fc[i-1]+fc[i])/2;
```

2.2.3复合语句

复合语句注释：

把多条语句看作一条语句

语法:BEGIN...END

这里，begin和end是成对出现的，被begin和end包围起来的语句可以有很多条，这些语句可以

看成是一条复合语句。

下面我们用begin...end来改写递增循环计算2日平均收盘价的公式：

```
fc:=close; //定义序列变量为收盘价
for i=2 to datacount do
begin
    a:= fc[i-1]+fc[i]; //定义一个临时的单值变量a，保存中间计算结果
    ma2[i] := a/2;
end;
```

这段代码，就是由2行代码组成的复合语句，被循环执行若干次。为了代码容易分辨，我们特别把复合语句中的2行代码，都向右缩了4格，表明这是2行复合语句，是被循环语句所控制的。以前对于这类分层次的语句，都要进行缩格，便于看懂代码，特别是复杂的代码，如果不进行缩格，时间久了，恐怕连自己都很难看懂，大家一开始就要养成好习惯。

有人会问，书写代码不缩格行吗？不缩格公式会不会出错？答案是，缩格书写代码，仅仅是为了方便看清程序代码的逻辑层次，对公式的运行没有影响。

有了复合语句，循环的功能就更加强大了，可以轻松实现多重循环，即循环中套循环。在计算N日的平均价时会用到，如果事先不知道N是多少，就要用到二重循环。对于循环中要执行的语句，如果重复太多，也可以使用多重循环来简化。以二重循环为例，大致结构如下：

```
for i=n1 to n2 do
begin
    语句;
    ...
    for j=m1 to m2 do
    begin
        语句;
        ...
    end;
    语句;
    ...
end;
```

end;

2.2.3. 条件语句

条件语句的语法如下：

IF条件语句

语法:IF cond THEN expr1 ELSE expr2

如果 cond 条件成立，则执行语句 expr1，否则执行 expr2 语句。

说明：

- 1、在条件判断比较简单的情况下，ELSE expr2 子句可以省略。
- 2、条件 cond 可以是单值变量，也可以为序列变量。当为序列变量时，cond将取最后一个周期的值做为条件判断语句。

条件语句的语法比较简单，但使用时却容易出错，下面举若干示例。

例1：修改成交量公式VOL，当流通盘不为零且当前周期为日以上周期时，显示换手率，代码参考如下（仿此，大家绕过指标模组，可以自行设计，“绑定”到周期、券种等的公式）

代码：

```
VOL, VOLSTICK;  
MA1:MA(VOL, M1);  
MA2:MA(VOL, M2);  
MA3:MA(VOL, M3);  
if capital>0 and DATATYPE>=6 then  
    换手率:vol/capital; //日以上周期及非指数个股，显示换手率
```

当切换到60分钟及以下周期，或者切换到大盘（此时流通盘=0），会发现“换手率”指标线、名称及数值都不显示。

以上是较简单的情况，没有使用 ELSE expr2 子句。上面代码稍加改进，使用复合语句，使之适用于任意周期：

例2，修改成交量公式，流通盘不为0时，显示换手率（60分钟及以下周期，计算当日最新

的换手率），代码参考如下：

代码：

```
VOL, VOLSTICK;  
MA1:MA (VOL, M1) ;  
MA2:MA (VOL, M2) ;  
MA3:MA (VOL, M3) ;  
  
IF CAPITAL>0 then //如果换手率>0, 则  
    IF DATATYPE>=6 then //如果周期为日及以上的较长周期, 则  
        b:=VOL/CAPITAL*100;  
    else //否则  
        begin //复合语句开始，即以下3条语句，视为1条语句，end表示复合语句结束  
            tj:=DAY>REF (DAY, 1) or BARSSINCE (CLOSE)=0;  
            ts:=BARSLAST (tj)+1;  
            b:=SUM (VOL, ts)/CAPITAL*100;  
        end;  
    换手率:b;
```

例3、通过参数N，控制调整均线数，代码参考如下：

代码：

```
input:p1 (5, 0, 300), p2 (10, 0, 300), p3 (20, 0, 300), p4 (30, 0, 300), n (4, 1, 4) ; {参数定义}  
IF n>0 then MA1:MA (CLOSE, P1) ; {如果n>=1则输出ma1指标线}  
IF n>1 then MA2:MA (CLOSE, P2) ; {如果n>=2则输出ma1指标线}  
IF n>2 then MA3:MA (CLOSE, P3) ; {如果n>=3则输出ma1指标线}  
IF n>3 then MA4:MA (CLOSE, P4) ; {如果n>=4则输出ma1指标线}
```



在使用条件语句“IF cond THEN”中，序列模式下，cond如果是序列变量，那么IF语句只取最后序列值做为条件判断，比如：

代码：

```
fc:=close;
fo:=open;
if fc>fo then //这里的fc、fo是序列变量，因此只取最后一个周期的条件做为判断依据
    xx:=1;
else
    xx:=0;
y:xx;
```

这里，if fc>fo then 这种写法的本意是，“如果收盘价大于开盘价则”，是针对序列变量的每个数据（相当于数组的每个元素），但是在序列模式下是不会得到执行结果的，在模型终端公式编辑器的序列模式运行中，正确的写法应该是这样的：

代码：

```
//如果xx是序列变量，则代码参考如下
fc:=close;
fo:=open;
for i=1 to datacount do
begin
    if fc[i]>fo[i] then
        xx[i]:=1; //请注意这里跟上面代码的不同
    else
        xx[i]:=0;
end
y:xx;
```

2.3. 序列模式和逐 K 线模式

迅投极速策略交易系统编辑器工作有两种模式，即序列和逐K线两种模式。序列模式公式系统每次刷新时解析公式按照序列或者常数计算返回结果，整个执行过程只解析一遍公式系统，我们前面所讲的控制语句的用法都是基于序列模式下运行的。逐K线模式为从第1个K线直到最后一个K线逐个解析公式系统，每根K线都会解析整个公式系统一遍，返回值也只有数值类型这一种，故这种模式运行时效率要比序列模式低，但此种模式下由于是逐根周期执行运算的，故我们可以在编写公式时使用各种更加灵活的控制语法。

2.3.1. 控制语句在两种不同模式下的运行特点

逐K线计算时，控制语句比如IF THEN，FOR等语句工作机制是每周期都去执行一次，因此在逐K线模式下，可以利用这种灵活的模式来设计我们的策略，比如加仓、减仓、资金管理策略等等。

序列模式计算时，控制语句条件允许使用序列变量，由于序列模式只执行一次控制语句解析，对于序列变量，仅取最后一个数据做为条件判断。

如：前面我们在序列模式下无法正常工作的公式

```
fc:=close;
fo:=open;
if fc>fo then //这里的 fc、fo 是序列变量，因此只取最后一个周期的条件做为判断依据
    xx:=1
else
    xx:=0;
```

但是在逐K线模式下，该公式是可以正常工作的。因为逐K线每根K线都得到了执行，故不需要向序列运行那样在后面用FOR循环重新赋值一遍了。

为了更能说明两种模式下的公式运行特点，特制作一个计算移动平均线的公式，如下：

```
//用于序列模式下运行的公式：
INPUT:N(5, 2, 500); //参数申明
```



```
RUNMODE:=1; //运行于序列模式
VARIABLE:I=0, S=0;
VAR1:=C; //变量申明
    FOR J=1 TO DATACOUNT DO BEGIN
        S:=S+VAR1[J];
        IF J>=N THEN BEGIN
            IF J>N THEN
                S:=S-VAR1[J-N];
            MA1[J]:=S/N; //实现MA(C, N)
            I:=0;
        END;
    END;
```

上述公式使用序列模式运行，但是如果在逐K线模式下运行上述公式就会变得异常缓慢，由于模型编辑器逐K线模式在每个周期上都要执行一遍这样的循环，效率自然就变得非常低了。鉴于模型编辑器的特点，如果将上述公式改进一下，则可以高效的在序列和逐K线模式同时高效运行，公式如下：

```
INPUT:N(5, 2, 500); //参数申明
VARIABLE:I=0, S=0; //全局变量申明
VAR1:=C; //模型编辑器下放在这里的变量为序列赋值
//为了加快运算速度，只有最后一个周期时才循环计算
IF NOT(ISLASTBAR) THEN EXIT;
FOR J=1 TO DATACOUNT DO BEGIN
    S:=S+VAR1[J];
    IF J>=N THEN BEGIN
        IF J>N THEN
            S:=S-VAR1[J-N];
        MA1[J]:=S/N; //实现MA(C, N)
        I:=0;
    END;
END;
```

END;

END;

上述公式中增加了 IF NOT (ISLASTBAR) 判断是否最后一个周期的指令，在逐 K 线模式下，由于是判断到最后一个周期才执行的下面循环，故效率是非常高的。对于序列模式，由于同样使用了 ISLASTBAR 控制，故符合控制语句取最后一个数据的特点，所以该公式同时可以在两种模式下得到正确执行。

为了更能说明逐 K 线的运行特点，计算移动平均线的公式还可以这样写

```
INPUT:N(5,2,500);           //参数申明
RUNMODE:=0;                  //工作于逐 K 线模式
IF BARPOS <= N THEN //从计算周期开始计算
    EXIT;
MA1:=C;
FOR J=1 TO N-1 DO
    MA1:=MA1+CLOSE[BARPOS-J];
MA1:=MA1/N;
```

这样的公式即保证了效率，也可以使编写公式的复杂程度大大降低，提高了公式的可读性。另外，逐 K 线模式下运行的代码，还可以配合 EXIT 指令，控制语句的执行流程，达到各种复杂的逻辑运算要求。

2.3.2. 关于模型运行时这两种模式的选择

我们在模型运行时尤其是新手用户往往面对如何选择这两种运行模式纠结，在通常情况下，我们推荐用户在序列模式下运行你的公式系统，因为这样会有很高的执行效率，只有在序列模式下无法表达编写出你的策略时，再考虑使用逐 K 线模式，因为逐 K 线可以精细的控制每跟 K 线周期的动作，所以灵活性较高，可以完成多数序列模式下无法完成的事情。

建议如下：在普通技术指标，选股指标，简单的图表程式化交易，以及公式中涉及到 BACKSET、REFX 等未来函数调用等，推荐使用序列模式；用户需要精细控制 K 线周期的操作时例如资金头寸管理、止损操作等，推荐使用逐 K 线模式。简单一句话，如果是指标交易，那么使用序列模式，算法交易，使用逐 K 线模式。

2.4. 交易系统函数

迅投极速策略交易系统编辑器提供完备的下单交易函数，可以完成下单、监控以及控制一系列的交易逻辑。

2.4.1. 下单函数

系统提供两种交易函数：TRADE 和 ORDER, 分别满足不同交易频率及精度的要求。

(1) TRADE函数

TRADE函数用于连接迅投的交易客户端，向交易端发送交易信号，具体的交易操作由交易客户端完成，

支持算法交易。其语法结构为：TRADE (TP, ADDRESS)，其中TP字段用于保存交易指令的各类参数，ADDRESS为系统指定发送交易指令的对应下单地址，不填时默认发送至'127.0.0.1:500000'，如下表为

TradeParam				
序号	参数名	简化名	类型	含义
1	strAcc	ac	string	账号组名
2	ePositionType	potype	EPositionType	分单模式
3	strMarket	market	string	市场
4	strCode	stcode	string	股票代码
5	eOperationType	optype	EOperationType	下单方向
6	eOrderType	ortype	EOrderType	算法交易类型
7	ePriceType	prtype	EPriceType	报价方式
8	dSuperPrice	suprice	double	单笔超价
9	eSuperPriceType	sptype	ENumberType	单笔超价类型
10	dFixPrice	fiprice	double	指定价格
11	nNum	vol	int	交易总量
12	eSingleVolumeType	svtype	EVolumeType	单笔基准量
13	dSingleVolumeRate	svrate	double	基准量比例
14	dPlaceOrderInterval	orinterval	double	下单间隔
15	dWithdrawOrderInterval	delinterval	double	撤单间隔
16	dLastVolumeMin	lvmin	double	尾单最小量
17	eLastVolumeMinType	lvmttype	ENumberType	尾单最小量类型，缺省比例
18	nSuperPriceEnable	spenable	int	超价启用笔数
19	dPriceRange	prange	double	波动区间



20	ePriceRangeType	prtype	ENumberType	波动区间价格类型，缺省比例
21	nSingleNumMax	svmax	int	单笔下单量最大值
22	nValidTime	vatime	int	有效时间，两种格式比如 164820，或者 10 分钟
23	nMaxOrderCount	mocount	int	最大委托次数
24	strUserOrderNum	usernum	string	用户自定义的下单编号

TRADE函数用法示例：

TRADEPARM TP;

TP.AC:= ' ACCOUNT1' ;TP.POTYPE:=0;TP.MARKET:= ' IF' ;TP.STCODE:= ' IF1604' ;TP.OTYPE:=0;

TP.ORTYPE:=1;TP.PRTYPE:=5;TP.SUPRICE:=100;TP.SPTYPE:=1;TP.FIPRICE:=2500;TP.VOL:=100;

TP.SVTYPE:=4;TP.SVRATE:=0.5;TP.ORINTERVAL:=3;TP.DELINTERVAL:=3;TP.LVMIN:-100;TP.SPENABLE:=10;

TP.PRANGE:=200;TP.PRTYPE:=1;TP.SVMAX:=100;TP.VATIME:=60;TP.MPCOUNT:=300;TP.USERNUM:= ' 001' ;

//以上为TP交易信息的设置，包括算法交易方式，下单标的、数量等等。

TRADE (TP, ' 127.0.0.1: 10000;127.0.0.1: 20000');//可以同时多个地址发送交易指令，以“;” 隔开。

(2) ORDER函数

ORDER函数为对应接口单功能的交易函数，用于向对应的接口单发送交易指令。此函数与TRADE函数区别在于，接口单功能将TRADE函数中关于算法交易等方面的设置界面化，以方便用户设置：

各类交易信息均可通过接口单功能设置保存实现。同时，通过ORDER函数可以对客户端发出篮子交易的指令，以执行预先保存在指定位置的一揽子证券。

ORDER函数的语法格式为ORDER(OP, CHANNEL, ADDRESS)。其中OP用于设置各类执行操作，CHANNEL以及ADDRESS为对应接口单的通道号以及网络地址。

ORDER函数用法示例：

//普通期货交易

```
order(0, 1, '127.0.0.1:1000');
```

//对指定地址接口单的1号通道执行开多操作，开多的标品种数量为接口单中保存

//篮子组合下单

```
order(25, 1, '127.0.0.1:1000', 'test 下单篮子');
```

```
order(37, 2, '127.0.0.1:1000', 'test 下单篮子');
```

//对预置股票篮子“test下单篮子”执行交易操作，篮子设置路径：

交易终端-->组合交易-->篮子管理

下单参数		操作参数说明
期货-六建风格	0	开多
	1	平昨多
	2	平今多

期货-四键风格	3	开空
	4	平昨空
	5	平今空
	6	平多, 优先平今
	7	平多, 优先平昨
	8	平空, 优先平今
	9	平空, 优先平昨
	10	卖出, 如有多仓, 优先平仓, 优先平今, 如有余量, 再开空
	11	卖出, 如有多仓, 优先平仓, 优先平昨, 如有余量, 再开空
期货-两键风格	12	买入, 如有空仓, 优先平仓, 优先平今, 如有余量, 再开多
	13	买入, 如有空仓, 优先平仓, 优先平昨, 如有余量, 再开多
	14	买入, 不优先平仓
	15	卖出, 不优先平仓
	19	看多, 优先平昨
期货-展期	20	看多, 优先平今
	21	看空, 优先平昨
	22	看空, 优先平今
股票	23	股票买入
	24	股票卖出
组合交易	25	组合买入
	26	组合卖出
	37	卖出投资组合

(3) PASSORDER函数

passorder(opType, orderType, accountID , orderCode, prType, price, volume)
 [, orderCode, prType, price, volume] 不是必填, 默认值分别是, "", -1, 0.0, 0

参数数据类型

opType: 数字型 (int)

orderType: 数字型 (int)

accountID: 字符串型 (string)

orderCode: 字符串型 (string)

prType: 数字型 (int)

price: 数字型 (double)

volume: 数字型 (double)

passorder(opType, orderType, accountID , orderCode, prType, price, volume)

opType (类型):

期货六键	期货四键	期货两键	股票买卖	组合交易
------	------	------	------	------

0 开多 1 平昨多 2 平今多 3 开空 4 平昨空 5 平今空	6 平多, 优先平今 7 平多, 优先平昨 8 平空, 优先平今 9 平空, 优先平昨	10 卖出, 如有多仓, 优先平仓, 优先平今, 如有余量, 再开空 11 卖出, 如有多仓, 优先平仓, 优先平昨, 如有余量, 再开空 12 买入, 如有空仓, 优先平仓, 优先平今, 如有余量, 再开多 13 买入, 如有空仓, 优先平仓, 优先平昨, 如有余量, 再开多 14 买入, 不优先平仓 15 卖出, 不优先平仓	23 股票买入 24 股票卖出 融资融券(部分版本不支持): 27 融资买入 28 融券卖出 29 买券还券 30 直面还券 31 卖券还款 32 直接还款 33 股票买入 34 股票卖出	股票组合: 25 组合买入 26 组合卖出 融资融券(部分版本不支持): 27 融资买入 28 融券卖出 29 买券还券 31 卖券还款 33 组合买入 34 组合卖出 期货组合: 40 开多 43 开空 46 平多, 优先平今 47 平多, 优先平昨 48 平空, 优先平今 49 平空, 优先平昨
--	--	--	--	--

passorder(opType, orderType, accountID , orderCode, prType, price, volume)

orderType 类型参数说明:

注: 期货不支持 1102 和 1202

注: 对所有账号组的操作 相当于 对账号组里的每个账号做一样的操作, 如

passorder(23, 1202, 'testS', '000001', 5, 0, 5000); 意思就是对账号组 testS 里的所有账号都以最新价开仓买入 5000 元市值的 000001 平安银行。

1101: 单股、单账号、普通、股/手 方式下单 (同 11)

1102: 单股、单账号、普通、金额(元)方式下单 (只支持股票)

1113: 单股、单账号、总资产、比例(0~1)方式下单

1123: 单股、单账号、可用、比例(0~1)方式下单

1201: 单股、账号组(无权重)、普通、股/手 方式下单 (同 12)

1202: 单股、账号组(无权重)、普通、金额(元)方式下单 (只支持股票)

1213: 单股、账号组(无权重)、总资产、比例(0~1)方式下单

1223: 单股、账号组(无权重)、可用、比例(0~1)方式下单



2101: 组合、单账号、普通、按组合股票 数量 (篮子中股票设定的数量) 方式下单 > 对应 volume 的单位为 篮子的 份 (同 21)

2102: 组合、单账号、普通、按组合股票 权重 (篮子中股票设定的权重) 方式下单 > 对应 volume 的单位为元 ok

2103: 组合、单账号、普通、按账号可用方式下单>(底层篮子股票怎么分配? 答: 按可用资金比例后按篮子中股票权重分配, 如用户没填权重则按相等权重分配) → 只对股票篮子支持

22: 组合、账号组 (无权重)、普通、默认方式下单, 兼容以前的 22 模式

2201: 组合、账号组 (无权重)、普通、按组合股票 数量 方式下单 (同 22)

2202: 组合、账号组 (无权重)、普通、按组合股票 权重 方式下单

2203: 组合、账号组 (无权重)、普通、按账号可用 方式下单 → 只对股票篮子支持

组合套利交易接口特殊设置

passorder(opType, orderType, accountID, orderCode, prType, hedgeRatio, volume)

accountID=' stockAccountID, futureAccountID'

orderCode =' basketName, futureName'

hedgeRatio: 套利比例 (0~2, 相当于%0 至 200%间套利)

volume: 份数\资金\比例

orderType:

2331: 组合、套利、合约价值自动套利、按组合股票 数量 方式下单

2332: 组合、套利、按合约价值自动套利、按组合股票 权重 方式下单

2333: 组合、套利、按合约价值自动套利、按账号可用方式下单

passorder(opType, orderType, accountID , orderCode, prType, price, volume)

accountID (账号 ID): 下单的账号 ID (可多个) 或 账号组名 或 套利组名 (一个篮子一个套利账号, 如 accountID=' 股票账户名, 期货账号')

passorder(opType, orderType, accountID , orderCode, prType, price, volume)

orderCode (下单代码):

有两种如果是单股或单期货、港股, 则该参数填合约代码; 如果是组合交易, 则该参数填篮子名称。如果是组合套利, 则填一个篮子名和一个期货合约名 (如 orderCode=' 篮子名, 期货合约名')。

passorder(opType, orderType, accountID , orderCode, prType, price, volume)

prType (下单选价类型): (特别的对于套利: 这个 prType 只对篮子起作用, 期货的采用默认的方式)

-1: 无效 (实际下单时, 模型需用界面交易函数那设定的选价类型)

4: 卖 1 价

5 : 最新价

- 6 :买 1 价
- 11 (指定价) 模型价 (只对单股情况支持, 对组合交易不支持)
- 12 :市价
- 13 :挂单价
- 14 :对手价

passorder(opType, orderType, accountID [, orderCode, prType, price, volume])

price(下单价格):

单股: prType 是模型价 时 price 有效; 其它情况无效

组合: 是组合套利时, price 作套利比例有效; 其它情况无效

passorder(opType, orderType, accountID [, orderCode, prType, price, volume])

根据 orderType 中 D 的值确定 volume 的单位:

D: (单股时) 1 股/手, 2 金额 (元), 3 比例 (%)

D: (组合时) 1 按组合股票数量(份), 2 按组合股票权重(元), 3 按账号可用 (%)

volume (下单数量 (股/手/元/%))

一个发出套利单的例子:

```
stockAccountID:='6000000255';
futureAccountID:='039427';
basketName:='testStock';//预先建立好的股票篮子
futureName:='IF1703';
operateType:=25;//股票篮子买入
orderType:=2333;//按账号可用方式下单并套利
priceType:=5;//以最新价
model_price:=1;//套利比例 100%, 这里对 modelprice 这个字段进行了重用作 套利比例用
volume:=0.05;//以账号可用资金的 5%买入篮子

AccountID:=stockAccountID+', '+futureAccountID;
orderCode:=basketName+', '+futureName;
doit:=0;
if c>o then //是阳线就套利
begin
//下套利单
orderType=2333
passorder(operateType, orderType, AccountID, orderCode, priceType, model_price, volume)
;
doit:=1;
end
```



single:doit;

passorder 对融资融券的支持:

passorder(opType, orderType, accountID [, orderCode, prType, price, volume])

opType:

27 融资买入

28 融券卖出

29 买券还券

30 直面还券

31 卖券还款

32 直接还款

33 股票买入

34 股票卖出

orderType:

11, 1101, 1102, 12, 1201, 1202

accountID: 信用账号

orderCode : 股票 ID 和 合约 (合同) 编号 ID 复用

当 opType 等于 29 或 31 时:

orderCode 可以携带上合约编号: orderCode = stkCode+"", " +compactID

当 opType 等于 32 时: orderCode = compactID 或 不填

其它情况 orderCode= stkCode 或不填

prType

price

volume (股、元)

例:

```
if o>c then begin
```

```
passorder(27, 1101, ' 6000000255' , ' 000001' , 5, 0, 1000); //融资买入 10 手 000001
```

```
end
```

(4) holding :返回持仓数量

调用: holding(AccountID , MarketID, StockID, Direction);

AccountID, MarketID, StockID 为字符串; Direction 为整型 (1 多, 2 空)

例:

```
ho:=holding( '037055' , ' IF' , ' IF06' , 2);
```

(5) holdings: 返回持仓信息

调用: holdings(Account);

例 1:

```
xxx:= holdings( '037055' );
```

例 2:

统计某个账号所有品种做多方向的持仓

```
xxx := holdings('037055');  
loh := 0;  
for x in xxx do begin  
if x.direction = 48 then  
loh:= loh + x.volume;  
end  
longhold:loh;
```

xxx 为一个 positiondetail 结构体，含有如下项：

exchangeid 证券市场, 交易所代码

exchangename 市场名字

productid 品种代码

productname 品种名称

instrumentid 证券代码, 合约代码

instrumentname 证券名称, 合约名称

hedgflag 投保

direction 买卖

opendate 成交日期

tradeid 最初开仓位的成交

volume 持仓量 当前拥股

openprice 开仓价

tradingday 交易日

margin 使用的保证金 历史的直接用 ctp 的，新的自己用成本价*存量*系数算 股票不需要

opencost 开仓成本 等于股票的成本价*第一次建仓的量，后续减持不影响，不算手续费 股票不需要

settlementprice /结算价 对于股票的当前价

closevolume 平仓量 等于股票已经卖掉的 股票不需要

closeamount 平仓额 等于股票每次卖出的量*卖出价*合约乘数（股票为 1）的累加 股票不需要

dloatprofit 浮动盈亏 当前量*（当前价-开仓价）*合约乘数（股票为 1）

closeprofit 平仓盈亏 平仓额 - 开仓价*平仓量*合约乘数（股票为 1） 股票不需要

marketvalue 市值 合约价值

positioncost 持仓成本 股票不需要

positionprofit 持仓盈亏 股票不需要

lastsettlementprice 最新结算价 股票不需要

instrumentvalue 合约价值 股票不需要

istoday 是否今仓

xttag 迅投标签

stockholder 股东账号



frozensvolume 期货不用这个字段, 冻结数量
canusevolume 期货不用这个字段, 股票的可用数量
onroadvolume 期货不用这个字段, 股票的在途数量
yesterdayvolume 期货不用这个字段, 股票的股份余额
lastprice 结算价 对于股票的当前价
profitrate 持仓盈亏比例
futuretradetype 成交类型
expiredate 到期日, 逆回购用
comtradeid 套利成交 Id
legid 组合 Id
totalcost 自定义累计成本 股票信用用到
singlecost 自定义单股成本 股票信用用到
coveredvolume 用于个股期权
sideflag 用于个股期权, 标记 '0' - 权利, '1' - 义务, '2' - '备兑'
referencerate 汇率, 目前用于港股通
structfundvol 分级基金可用 (可分拆或可合并)
redemptionvolume 分级基金可赎回量

(6) ordering: 返回当前委托数量

调用: `ordering(AccountID, MarketID, StockID, Direction);`
AccountID, MarketID, StockID 为字符串; Direction 为整型 (1 buy, 2 sell)

例:

```
ord:=ordering('037055', 'IF', 'IF06', 2);
```

(7) orderings: 返回委托信息

调用: `orderings(AccountID);`

例:

统计某账号买入的所有的当天委托量

```
xxx := orderings('037055');  
loo := 0;  
for x in xxx do begin  
if x.direction = 48 then  
loo:= loo + x.volumetotaloriginal;  
end  
longorder:loo
```

xxx 为一个 orderdetail 结构体, 含有如下项:

exchangeid 证券市场, 交易所代码
exchangename 市场名字
productid 品种代码



productname 品种名称
instrumentid 证券代码, 合约代码
instrumentname 证券名称, 合约名称
sessionid
frontid 前端 id
orderref 下单引用 等于股票的内部委托号
orderpricetype 类型, 例如市价单 限价单
direction 期货多空 股票买卖
offsetflag 期货开平, 股票买卖其实就是开平
hedgeflag 投保
limitprice 限价单的限价, 就是报价
volumetotaloriginal 最初委托量
ordersubmitstatus 提交状态
ordersysid 委托号
orderstatus 委托状态
volumetraded 已成交量
volumetotal 当前总委托量 股票不需要总委托量
errorid
errormsg 状态信息
taskid
frozenmargin 冻结保证金
frozencommission 冻结手续费
insertdate 日期
inserttime 时间
xttag 迅投标签
tradeprice 成交均价
cancelamount 已撤数量
optname 展示委托属性的中文
tradeamount 成交额 期货=均价*量*合约乘数
entrusttype 委托类别
cancelinfo 废单原因
undercode 标的证券
covereflag 备兑标记 '0' - 非备兑, '1' - 备兑
orderpricermb 委托价格 人民币 用于港股通
tradeamounttrmb 成交金额 人民币用于港股通
referencerate 参考汇率 用于港股通

(8) deal: 返回某个时间内的成交数量

调用: deal(AccountID , MarketID, StockID, Direction);

或 deal(AccountID , MarketID, StockID, Direction, Seconds);

AccountID, MarketID, StockID 为字符串; Direction 为整型 (1 多, 2 空), Seconds 为整型, 表示多少秒内。



例 1:

de:=deal('037055' , ' IF' , ' IF06' , 2); //返回当天 IF06 的 sell 的成交数量

例 2:

de:=deal('037055' , ' IF' , ' IF06' , 2, 90); //返回最近 90 秒内 IF06 的 sell 的成交数量

(9) deals: 返回成交信息

调用: deals(AccountID);

例:

返回某账号当天 buy 的成交量

```
xxx := deals('037055');
```

```
dea:= 0;
```

```
for x in xxx do begin
```

```
if x.direction = 48 then
```

```
dea:= dea + x.volume;
```

```
end
```

```
longdeal:dea
```

xxx 为一个 dealdetail 结构体, 含有如下项:

exchangeid 证券市场, 交易所代码

exchangename 市场名字

productid 品种代码

productname 品种名称

instrumentid 证券代码, 合约代码

instrumentname 证券名称, 合约名称

tradeid 成交编号

orderref 下单引用 等于股票的内部委托号

ordersysid 委托号

direction 买卖 股票不需要

offsetflag 开平 股票的买卖

hedgeflag 投保 股票不需要

price 成交均价

volume 成交量 期货单位手 股票做到股

tradedate 成交日期

tradetime 成交时间

comssion 手续费

tradeamount 成交额 期货=均价*量*合约乘数

taskid

xttag 迅投标签

orderpricetype 类型, 例如市价单 限价单

optname 展示委托属性的中文

entrusttype 委托类别



futuretradetype 成交类型
realoffsetflag 实际开平, 主要是区分平今和平昨
coveredflag 备兑标记 '0' - 非备兑, '1' - 备兑
closetodayvolume 平今量, 不显示
orderpricermb 委托价格 人民币 用于港股通
pricermb 目前用于港股通
tradeamounttrmb 目前用于港股通
referencerate 汇率, 目前用于港股通
xttrade 是否是迅投交易

(8) cancel:撤单

cancel(委托号);

一个模型例子:

```
//VAB 不区分大小写
variable:TestHolding=0;
//.....下单参数定义.....

ORDERTYPE:=1101; //下单类型
ACCOUNTID:='6000000248'; //填写对应资金账号, 用户修改成自己的资金账号
ORDERCODE:=STKLABEL(); //下单代码, 当前主图品种
PRICETYPE:=5; //下单价格类型, 4 卖 1 价, 5 最新价, 6 买 1 价, 12 市价, 13 挂单价, 14 对手
价, 11(指定价)模型价
VOLUME:=100; //下单数量, 100 股

//.....下单.....

DIFF := EMA(CLOSE, 12) - EMA(CLOSE, 26);
DEA := EMA(DIFF, 9);
MACD1 := 2*(DIFF-DEA), COLORSTICK;

t:BARSLAST(TestHolding=0), nodraw;//////////持仓周期

bk:= macd1>0;//////////开仓条件, 可根据需要更改, 如 bk:= close>open;

bp:=(t>=3 and macd1<0);/////平仓条件

nn:=0;//////////往后推迟几个周期交易//

IF (ref(Bk, nn) and not(bp) and TestHolding=0 ) THEN BEGIN
TestHolding:=1;
BBD:=BARPOS;
```

```
DRAWTEXT(1 ,H+4,' 买入');
VERTLINE(1 ,h+10,l-10,coloryellow,1,VTDOT);
PASSORDER(23{股票买入},ORDERTYPE{单股下单类型},ACCOUNTID{账号},ORDERCODE{下单代码},PRICETYPE{选价类型},-1{下单价格选价类型非指定价填-1},VOLUME{下单量});
END

//卖出
IF (ref(bp,nn) AND TestHolding>0 ) THEN BEGIN
TestHolding:=0;
BBD:=0;
DRAWTEXT(1,H+1,' 卖出');
VERTLINE(1,h+10,l-10,colorwhite,1,VTDOT);
PASSORDER(24{股票买入},ORDERTYPE,ACCOUNTID,ORDERCODE,PRICETYPE,-1,VOLUME);
END

//.....撤单.....
orderss:= orderings(ACCOUNTID);//获取该资金账号所有的委托信息
nowtime:=CURRENTTIME();
for ord in orderss do begin
inserttime:=strtonum(ord.inserttime);
if isequalv(ord.orderstatus,56)=0 and nowtime - inserttime > 30 then begin //委托没
成功, 30s 内没成交或部分成交则撤单
cancel(ord.ordersysid);
end
end
```

2.4.2. 监控函数

迅投极速策略交易系统提供 TACCOUNT、MARKETVALUE、HOLDING 和 HOLDINGS 函数用以监控账户资金以及持仓状况

TACCOUNT 示例

获取指定账号的可用资金;

如: TACCOUNT(1,'37500001');

1 表示是期货账号, 2 为普通股票账号, 3 为信用账号, 37500001 是账号 ID, 返回可用资金;

HOLDING 示例

得到当前帐户持仓量, 多仓返回正数空仓返回负数

用法: HOLDING

HOLDINGS 示例

取某资金帐号对应的持仓

用法: HOLDINGS (AA) 表示获取帐号“AA”的持仓

MARKETVALUE 示例

取资金账号股票市值

用法: MARKETVALUE (2, “AA”) 表示返回帐号“AA”的股票市值, 1 表示是期货账号, 2 为普通股票账号, 3 为信用账号, 37500001 是账号 ID, 返回可用资金

2.4.3. 控制函数

控制函数主要有 SLEEP 和 CANCEL 函数分别用于控制下单的频率以及对于指定报单指令执行撤单。

CANCEL语法结构CANCEL (AA) AA为指定的委托编号, 通过系统获得

针对委托号进行撤单

用法: CANCEL (AA) 表示委托号为 “AA” 的委托

SLEEP函数作用于模型的最后一个周期, 在最新周期中延时设定的时间之后再执行之后的语句

用法: SLEEP (D), D 为延时的设置时间, 单位为毫秒 (1 秒钟等于 1000 毫秒)。

例如: SLEEP (1000) 表示等待 1 秒后再执行下行语句

2.4.4. 线型描述

(1) COLORBLACK

释义	设为黑色
示例	MA1:MA (CLOSE, 5), COLORBLACK

(2) COLORBLUE

释义	设为蓝色
----	------

示例	MA1:MA (CLOSE, 5), COLORBLUE
----	------------------------------

(3) COLORBROWN

释义	设为棕色
示例	MA1:MA (CLOSE, 5), COLORBROWN

(4) COLORCYAN

释义	设为青色
示例	MA1:MA (CLOSE, 5), COLORCYAN

(5) COLORGRAY

释义	设为灰色
示例	MA1:MA (CLOSE, 5), COLORGRAY

(6) COLORGREEN

释义	设为绿色
示例	MA1:MA (CLOSE, 5), COLORGREEN

(7) COLORMAGENTA

释义	设为晶红色
示例	MA1:MA (CLOSE, 5), COLORMAGENTA

(8) COLORRED

释义	设为红色
示例	MA1:MA (CLOSE, 5), COLORRED

(9) COLORWHITE

释义	设为白色
示例	MA1:MA (CLOSE, 5), COLORWHITE

(10) COLORYELLOW

释义	设为黄色
示例	MA1:MA (CLOSE, 5), COLORYELLOW

(11) COLORSTICK

释义	颜色柱状线：以零轴为中心画彩色棒状线，零轴下为阴线颜色，零轴上为阳线颜色
示例	CLOSE-OPEN, COLORSTICK

(12) CIRCLEDOT

释义	画小圆圈线
示例	CLOSE, CIRCLEDOT;

(13) NOAXIS

释义	无坐标：不影响坐标最高最低值
示例	C, NOAXIS; 输出收盘价, 但不影响坐标最高最低值, 用于叠加到其它指标上

(14) LINETHICK

释义	改变指标线粗细
示例	MA1:MA(CLOSE, 5), LINETHICK3

2.4.5. 逻辑函数

(1) ALL

释义	是否一直满足条件。
用法	ALL(X, N), 统计 N 周期中是否一直都满足 X 条件, 若 N=0 则从第一个有效值开始。
示例	ALL(CLOSE>OPEN, 20) 表示是否 20 周期内全部都收阳线。

(2) ANY

释义	是否存在
用法	ANY(X, N) 返回 N 周期内是否存在满足条件 X, N 可为常数或变量, 若 N=0 则从第一个有效值开始
示例	ANY(C>0, 10) 表示 10 个周期中存在阳线

(3) BETWEEN

释义	介于两个数之间。
用法	BETWEEN(A, B, C) 表示 A 处于 B 和 C 之间时返回 1, 否则返回 0
示例	BETWEEN(CLOSE, MA(CLOSE, 10), MA(CLOSE, 5)) 表示收盘价介于 5 日均线 and 10 日

	均线之间
--	------

(4) CROSS

释义	两条线交叉。
用法	CROSS (A, B) 表示当 A 从下方向上穿过 B 时返回 1, 否则返回 0
示例	CROSS (MA (CLOSE, 5), MA (CLOSE, 10)) 表示 5 日均线与 10 日均线交叉

(5) IF

释义	根据条件求不同的值。
用法	IF (X, A, B) 若 X 不为 0 则返回 A, 否则返回 B
示例	IF (CLOSE>OPEN, HIGH, LOW) 表示该周期收阳则返回最高值, 否则返回最低值

(6) IFN

释义	根据条件求不同的值。
用法	IFN (X, A, B) 若 X 不为 0 则返回 B, 否则返回 A
示例	IFN (CLOSE>OPEN, HIGH, LOW) 表示该周期收阴则返回最高值, 否则返回最低值

(7) ISDOWN

释义	该周期是否收阴
示例	ISDOWN (), 当收盘<开盘时, 返回值为 1, 否则为 0

(8) ISEQUAL

释义	该周期是否平盘。
示例	ISEQUAL (), 当收盘=开盘时, 返回值为 1, 否则为 0

(9) ISLASTBAR

释义	该周期是否为最后一个周期。
示例	ISLASTBAR, 最后一个周期返回值为 1, 其余为 0

(10) ISUP

释义	该周期是否收阳。
示例	ISUP (), 当收盘>开盘时, 返回值为 1, 否则为 0

(11) LAST

释义	持续存在
用法	LAST(X, A, B) 返回第前 A 周期到第前 B 周期是否一直满足条件 X 若 A 为 0，表示从第一天开始，B 为 0，表示到最后日止
示例	LAST(C>0, 10, 5) 表示从第前 10 个周期到第前 5 个周期内一直是阳线

(12) LONGCROSS

释义	两条线维持一定周期后交叉。
用法	LONGCROSS(A, B, N) 表示 A 在 N 周期内都小于 B，本周期从下方向上穿过 B 时返回 1，否则返回 0
示例	LONGCROSS(MA(CLOSE, 5), MA(CLOSE, 10), 5) 表示 5 日均线维持 5 周期后与 10 日均线交叉

(13) NOT

释义	求逻辑非。
用法	NOT(X) 返回非 X，即当 X=0 时返回 1，否则返回 0
示例	NOT(ISUP) 表示平盘或收阴

(14) RANGE

释义	介于某个范围之内。
用法	RANGE(A, B, C) 表示 A 大于 B 同时小于 C 时返回 1，否则返回 0
示例	RANGE(CLOSE, MA(CLOSE, 5), MA(CLOSE, 10)) 表示收盘价大于 5 日均线并且小于 10 日均线

(15) VALID

释义	判断指定值是否是有效数据
示例	VALID(X)，当 X 为有效数据时返回 1，否则返回 0

(16) VALUEWHEN

释义	条件跟随：当条件 COND 满足时，取当时的 DATA 的值，否则取得 VALUEWHEN 的前一个值。
用法	VALUEWHEN(COND, DATA)
用法	VALUEWHEN(HIGH>REF(HIGH, 5), HIGH) ;

表示当前最高价大于前五个周期最高价的最大值时返回当前最高价

2.4.6. 动态行情

序号	函数名称	释义
1	DYNAINFO (3)	取得最新动态行情：昨收
2	DYNAINFO (4)	取得最新动态行情：今开
3	DYNAINFO (5)	取得最新动态行情：最高
4	DYNAINFO (6)	取得最新动态行情：最低
5	DYNAINFO (7)	取得最新动态行情：最新
6	DYNAINFO (8)	取得最新动态行情：总手
7	DYNAINFO (9)	取得最新动态行情：现手
8	DYNAINFO (10)	取得最新动态行情：总额
9	DYNAINFO (12)	取得最新动态行情：涨跌
10	DYNAINFO (13)	取得最新动态行情：振幅
11	DYNAINFO (14)	取得最新动态行情：涨幅
12	DYNAINFO (15)	取得最新动态行情：委比
13	DYNAINFO (16)	取得最新动态行情：委差
14	DYNAINFO (18)	取得最新动态行情：委买
15	DYNAINFO (19)	取得最新动态行情：委卖
16	DYNAINFO (20)	取得最新动态行情：委买价
17	DYNAINFO (21)	取得最新动态行情：委卖价
18	DYNAINFO (25)	取得最新动态行情：买一量
19	DYNAINFO (26)	取得最新动态行情：买二量
20	DYNAINFO (27)	取得最新动态行情：买三量
21	DYNAINFO (28)	取得最新动态行情：买一价
22	DYNAINFO (29)	取得最新动态行情：买二价
23	DYNAINFO (30)	取得最新动态行情：买三价
24	DYNAINFO (31)	取得最新动态行情：卖一量
25	DYNAINFO (32)	取得最新动态行情：卖二量
26	DYNAINFO (33)	取得最新动态行情：卖三量
27	DYNAINFO (34)	取得最新动态行情：卖一价
28	DYNAINFO (35)	取得最新动态行情：卖二价
29	DYNAINFO (36)	取得最新动态行情：卖三价

2.4.7. 绘图函数

(1) BARSSET

释义	绘制标记.
----	-------

用法	<p>BARSSET (COND, PRICE, N, OFFSET),</p> <p>当 COND 条件满足时, 在 PRICE 位置绘制标记。N 为所影响到的周期数。OFFSET 为影响周期的偏移, 0 为以中心点前后影响, 负数为往前影响, 正数为向后影响。</p> <p>该函数通常用在逻辑公式中。</p>
----	---

(2) COLORRGB

释义	将红, 绿, 蓝三基色混和成一个颜色值。
用法	<p>COLORRGB (R, G, B)</p> <p>R, G, B 分别取值为 0—255。</p> <p>该函数仅用在 DRAWTEXT 等画线函数中 COLOR 参数指定颜色使用。</p>
示例	<p>DRAWTEXT (CLOSE/OPEN>1.08, LOW, ' 大阳线', COLORRGB (255, 0, 0)); 将显示红色的大阳线文字。</p> <p>若用户需指定指标线颜色, 请使用 COLOR00FFFF 等这种语法。</p>

(3) DRAWTEXT

释义	在图形上显示文字。
用法	<p>DRAWTEXT (COND, PRICE, TEXT[, COLOR, ALIGN]),</p> <p>当 COND 条件满足时, 在 PRICE 位置书写文字 TEXT。COLOR (可选参数) 文字颜色, ALIGN (可选参数) 对齐方式 0 中对齐; 1 左对齐; 2 右对齐; 3 图中; 4 图上; 5 图下。</p>
示例	<p>DRAWTEXT (CLOSE/OPEN>1.08, LOW, ' 大阳线')</p> <p>表示当日涨幅大于 8% 时在最低价位置显示“大阳线”字样。</p>

(4) KLINE

释义	在图形上绘制 K 线。
用法	<p>KLINE (O, H, L, C, T)</p> <p>O, H, L, C 分别为开高低收</p> <p>T 为绘制类型, 0 表示与主图 K 线画法相同, 1 表示不影响坐标高低值, 可用于叠加在其它指标上</p>
示例	<p>例 1: KLINE (O, H, L, C, 0)</p> <p>例 2: 在 KDJ 公式中叠加 KLINE (O, H, L, C, 1)</p>

(5) VERTLINE

释义	在图形上绘制垂直线。
用法	<p>VERTLINE (COND, [PRICE1, PRICE2, COLOR, WIDTH, TYPE])</p> <p>当 COND 条件满足时，在 PRICE1 和 PRICE2 之间画线。</p> <p>PRICE1 和 PRICE2 均省略时表示在窗格高低之间画垂直线</p> <p>COLOR(可选参数)为线颜色</p> <p>WIDTH(可选参数)为线的宽度</p> <p>TYPE(可选参数)为线的风格分别为：</p> <p>VTSOLID 普通线</p> <p>VTDASH 虚线</p> <p>VTDOT 点线</p> <p>VTDASHDOT 虚线和点交替</p> <p>VTDASHDOTDOT 虚线和两点交替。</p> <p>除了 VTSOLID 风格以外，其他风格必须保证 WIDTH 为 1 时才有效。</p>

(6) DRAWICON

释义	在图形上绘制小图标。
用法	<p>DRAWICON (COND, PRICE, TYPE[, ALIGN]), 当 COND 条件满足时, 在 PRICE 位置画 TYPE 号图标。</p> <p>ALIGN(可选参数)对齐方式 , 0 图标中对齐; 1 图标上缘; 2 图标中缘; 3 图中; 4 图上; 5 图下</p> <p>例如: DRAWICON(CLOSE>OPEN, LOW, 1) 表示当收阳时在最低价位置画 1 号图标。</p> <p>绘制图标按照“符号”工具栏对应排列，如果要自定义绘制的图标，请在“自定义”工具栏操作里将对应的图标修改即可。</p>

(7) DRAWBMP

释义	在图形上绘制位图。
用法	<p>DRAWBMP (COND, PRICE, BMPFILE[, ALIGN]), 当 COND 条件满足时,</p> <p>在 PRICE 位置画 BMPFILE 文件名指定的 BMP 位图</p> <p>(初始路径与\DOCUMENT 目录的文档 *.STK 文件放在一起)。</p>
示例	DRAWBMP (CLOSE>OPEN, LOW, 'SUN')

	<p>表示当收阳时在最低价位置画 SUN.BMP 位图。</p> <p>ALIGN(可选参数) 对齐方式 , 0 图标中对齐;1 图标上缘;2 图标中缘;3 图中;4 图上;5 图下</p>
--	---

(8) DRAWNUMBER

释义	在图形上显示数字。
用法	<p>DRAWNUMBER(COND, PRICE, NUMBER, PRECISION[, COLOR, ALIGN])</p> <p>当 COND 条件满足时, 在 PRICE 位置书写数字 NUMBER,</p> <p>PRECISION 为小数显示位数(取值范围 0-7), 其中 0-6 表示位数, 7 表示自动显示位数。</p> <p>COLOR(可选参数) 为数字颜色。</p> <p>ALIGN(可选参数) 对齐方式 0 中对齐;1 左对齐;2 右对齐;3 图中;4 图上;5 图下。</p>
示例	<p>DRAWNUMBER(CLOSE/OPEN>1.08, HIGH, (CLOSE-REF(C, 1))/REF(C, 1)*100, 2)</p> <p>表示当日涨幅大于 8%时在最高价位置显示涨幅(相对开盘价的百分比)。</p>

2.4.8. 字符串函数

(1) MARKETNAME

释义	取得当前品种的市场名称。
用法	MARKETNAME(), 将返回当前品种的市场名称
示例	<p>沪市返回“上证所”;</p> <p>深市返回“深交所”</p>

(2) INBLOCK

释义	判断本股票是否板块成员
用法	INBLOCK(S), 若本股票是板块 S 的成员将返回 1, 否则返回 0.
示例	INBLOCK('工业板块'), 若本股票属于工业板块则返回 1

(3) LOWERSTR

释义	将字符串转换为小写。
用法	LOWERSTR(STR), 将返回 STR 对应的小写字符串
示例	LOWERSTR('EFGH') 将返回“efgh”

(4) LTRIM

释义	除去字符串开始空格
用法	LTRIM(STR)
示例	LTRIM(' SH') 返回 SH

(5) MARKETLABEL

释义	取得当前品种的市场代码
用法	MARKETLABEL(), 将返回当前品种的市场代码
示例	沪市返回"SH", 深市返回"SZ"

(6) MARKETLABEL1

释义	取得当前品种的市场代码。 较 MARKETLABEL, MARKETLABEL1 返回值更加精确
用法	MARKETLABEL1(), 将返回当前品种的市场代码, 例如: 上证 A 股返回"SHZB", 深证主板返回"SZZB", 深证中小板返回"SZZX", 深圳创业板返回"SZCY", 其他板块的返回值与 MARKETLABEL 函数相同。

(7) NUMTOSTR

释义	将数字转化为字符串, 用户可以设定精度
用法	NUMTOSTR(N, M), 将 N 转化为字符串返回, 精确到小数点后 M 位
示例	NUMTOSTR(CLOSE, 5) 将返回收盘价对应的字符串, 例如"15.78000"

(8) RTRIM

释义	除去字符串尾部空格
用法	RTRIM(STR)
用法	RTRIM(' SH ') 返回 SH

(9) STKLABEL

释义	取得品种代码
用法	STKLABEL(), 将返回当前品种的代码, 例如深发展 A 将返回"000001"

(10) STKNAME

释义	取得品种名称
----	--------

用法	STKNAME(), 将返回当前品种的名称, 例如深发展 A 将返回“深发展 A”
----	---

(11) STRCAT

释义	把一个字符串添加到另一个字符串中
用法	STRCAT(DES, STR), 将 STR 字符串添加到 DES 字符串末尾
示例	STRCAT('ABC', 'DEF') 将返回“ABCDEF”

(12) STRCMP

释义	字符串比较
用法	STRCMP(STR1, STR2), 若 STR1>STR2 则返回 1, STR1<STR2 返回-1, 相等则返回 0
示例	STRCMP('ABCDEF', 'ABC') 返回 1 表示当前最高价大于前五个周期最高价的最大值时返回当前最高价

(13) STRFIND

释义	在字符串中查找另一个字符串
用法	STRFIND(STR, S1, N), 从字符串 STR 的第 N 个字符开始查找字符串 S1, 返回找到的位置, 若没有找到就返回 0
示例	STRFIND('ABCDEFGH', 'CDE', 1) 返回 3

(14) STRICMP

释义	忽略大小写比较字符串
用法	STRICMP(STR1, STR2), 若 STR1>STR2 则返回 1, STR1<STR2 返回-1, 相等则返回 0
示例	STRCMP('ABCDEF', 'ABC') 返回 1

(15) STRINGTOFILE

释义	输出指定的字符串到一个指定的文件中 用户可以在公式中通过输出指定的字符串到文件来实现调试或者其他的目的. 借此可以借助这个功能来完成监控公式运行的各种细节参数. 该函数用法与 DEBUGFILE2 基本相同, 唯一区别是该函数在写文件时会自动清空之前文件中写入的数据
----	---

用法	STRINGTOFILE (PATH, STR), PATH 为用户的本地计算机路径, STR 为用户指定输出的一个行文字
示例	<p>STRINGTOFILE ('D:TEST.TXT', '当前资产为 1000'), 将在公式的监控部分输出到 D:TEST.TXT 文件.</p> <p>如果字符串输出中涉及到数字变量, 可以使用 NUMTOSTR 函数转化成字符串后, 然后再进行相加</p>

(16) STRINSERT

释义	从指定位置插入一个子字符串
用法	STRINSERT (STR, INDEX, STR1), 在 STR 字符串的第 INDEX 地方开始插入字符串 STR1
示例	STRINSERT ('ABCDEF', 2, 'ZZZ') 函数将返回 "ABZZZCDEF"

(17) STRLEFT

释义	取得字符串的左边部分
用法	STRLEFT (STR, N), 返回字符串 STR 的左边 N 个字符
示例	STRLEFT ('ABCDEF', 3) 得到 "ABC"

(18) STRLEN

释义	求字符串的长度
用法	STRLEN (STR), 将返回 STR 字符串的长度, 由于系统采用 ANSI 字符编码, 一个汉字等于 2 个字节

(19) STRMID

释义	取得字符串的中间部分
用法	STRMID (STR, N, M), 返回字符串 STR 的第 N 个字符开始的长度为 M 个字符的字符串
示例	STRMID ('ABCDEF', 3, 3) 得到 "CDE"

(20) STRNCMP

释义	指定长度比较字符串
用法	STRNCMP (STR1, STR2, LEN)

示例	VAR1:=STRNCMP(STKNAME,'ST',2)=0 输出 1 表示该股为 ST 股
----	---

(21) STRREMOVE

释义	从指定位置开始的地方删除一个或多个字符
用法	STRREMOVE (STR, INDEX, COUND), 在 STR 字符串的第 INDEX 地方开始删除 COUND 个字符
示例	STRREMOVE ('ABCDEF', 2, 2) 函数将返回"ABEF"

(22) STRREPLACE

释义	用一个字符替换另一个字符
用法	STRREPLACE (STR, STROLD, STRNEW), 将 STR 字符串中的 STROLD 替换为 STRNEW
示例	STRREPLACE ('ABCDEFG', 'BCD', 'ZZZ') 函数将返回"AZZZEFG"。 在替换之后, 该字符串有可能增长或缩短; 那是因为 STRNEW 和 STROLD 的长度不需要是相等的。

(23) STRRIGHT

释义	取得字符串的右边部分
用法	STRRIGHT (STR, N), 返回字符串 STR 的右边 N 个字符
示例	STRRIGHT ('ABCDEF', 3) 得到"DEF"

(24) STRTONUM

释义	将字符串转化为数字
用法	STRTONUM (STR), 将 STR 转化为数字返回
示例	STRTONUM ('12.5') 将返回数值 12.5

(25) STRTRIMLEFT

释义	整理字符串左边
用法	STRTRIMLEFT (STR, STR1), 将一群特定的字符 STR1 从字符串 STR 的开始处删除
示例	STRTRIMLEFT (' ABC', ' ') 函数将返回"ABC"

(26) STRTRIMRIGHT

释义	整理字符串右边
用法	STRTRIMRIGHT (STR, STR1), 将一群特定的字符 STR1 从字符串 STR 的末尾处删除

示例	STRTRIMRIGHT('ABC ', ' ') 函数将返回"ABC"
----	--------------------------------------

(27) UPPERSTR

释义	将字符串转换为大写
用法	UPPERSTR(STR), 将返回 STR 对应的大写字符串
示例	UPPERSTR('abcd') 将返回"ABCD"

2.4.9. 引用函数

(1) BARSCOUNT

释义	求有效周期数
用法	BARSCOUNT(X) 第一个有效数据到当前的天数
示例	BARSCOUNT(CLOSE) 取得上市以来总交易日数

(2) BACKSET

释义	将当前位置到若干周期前的数据设为 1
用法	BACKSET(X, N), 若 X 非 0, 则将当前位置到 N 周期前的数值设为 1
示例	BACKSET(CLOSE>OPEN, 2) 若收阳则将该周期及前一周期数值设为 1, 否则为 0

(3) BARSLAST

释义	上一次条件成立到当前的周期数
用法	BARSLAST(X): 上一次 X 不为 0 到现在的天数
示例	BARSLAST(CLOSE/REF(CLOSE, 1) >= 1.1) 表示上一个涨停板到当前的周期数 如果没有符合条件的周期, 函数将返回零

(4) BARSSINCE

释义	第一个条件成立到当前的周期数
用法	BARSSINCE(X): 第一次 X 不为 0 到现在的天数
示例	BARSSINCE(HIGH>10) 表示股价超过 10 元时到当前的周期数 如果没有符合条件的周期, 函数将返回零

(5) BARSSINCEN

释义	N 个周期内第一个条件成立到当前的周期数
用法	BARSSINCEN(X, N):N 周期内第一次 X 不为 0 到现在的周期数, N 大于或等于 2
示例	BARSSINCEN(HIGH>10, N) 表示 N 个周期内的股价超过 10 元时到当前的周期数 如果没有符合条件的周期, 函数将返回零

(6) COUNT

释义	统计满足条件的周期数
用法	COUNT(X, N), 统计 N 周期中满足 X 条件的周期数, 若 N=0 则从第一个有效值开始
示例	COUNT(CLOSE>OPEN, 20) 表示统计 20 周期内收阳的周期数

(7) CALLSTOCK

释义	引用同期的其他证券数据
用法	<p>CALLSTOCK(CODE, TYPE[, CYC, N]), 引用指定品种代码为 CODE, 周期为 CYC(可选) 若不填或者为-1 表示使用当前周期, 类型为 TYPE 的数据</p> <p>N 为左右偏移周期个数(可选) 0 表示引用当前数据, <0 为引用之前数据, >0 为引用之后数据。</p> <p>其中 TYPE 的值可为 VTOPEN(开盘) VTHIGH(最高) VTLOW(最低) VTCLOSE(收盘) VTVOL(成交量) VTAMOUNT(成交额) vtOPENINT(持仓量) VTADVANCE(涨数, 大盘有效) VTDECLINE(跌数, 大盘有效) 以及外部数据和万德数据</p> <p>如果找不到同期数据, 那么将返回最近的一个。</p> <p>CYC 范围为 0-19, 分别表示</p> <p>0:分笔成交、1:1 分钟、2:5 分钟、3:15 分钟、4:30 分钟、5:60 分钟</p> <p>6:日、7:周、8:月、9:年、10:多日、11:多分钟、12:多秒</p> <p>13:多小时、14:季度线、15:半年线、16:节气线、17:3 分钟、18:10 分钟、19:多笔线</p>
示例	<p>CALLSTOCK('1A0001', VTCLOSE, 6, -1) 表示引用昨日品种 1A0001 的日线收盘价</p> <p>CALLSTOCK('SH600000', VTOPEN) 表示引用 SH 市场的 600000, 使用当前周期</p> <p>引用数据时, 需要实现确认被引用品种周期数据齐全, 再首次使用或者在不定时, 请手工进行数据补充工作</p>

(8) CALLPYTHON

释义	引用 python 指标的计算结果
用法	<p>CALLPYTHON(OUTPUT, CYC, MSCODE), 引用指定市场及品种组合代码为 MSCODE, 周期为 CYC 的数据 OUTPUT</p> <p>CYC 范围为 0-19, 分别表示</p> <p>0:分笔成交、1:1 分钟、2:5 分钟、3:15 分钟、4:30 分钟、5:60 分钟</p> <p>6:日、7:周、8:月、9:年、10:多日、11:多分钟、12:多秒</p> <p>13:多小时、14:季度线、15:半年线、16:节气线、17:3 分钟、18:10 分钟、19:多笔线</p>
示例	<p>CALLPYTHON('TEST.A', 6, '') 表示引用当前图对应品种的 TEST python 类型指标中的在日线上的输出变量 A</p> <p>CALLPYTHON('TEST.A', 6, 'SH600000') 表示引用 SH 市场的 600000 对应 python 指标的输出 A 的数据</p> <p>引用数据时, 需要实现确认被引用品种周期数据齐全, 再首次使用或者在不定时, 请手工进行数据补充工作</p>

(9) CURRBARSCOUNT

释义	求到最后交易日的周期数
用法	CURRBARSCOUNT 求到最后交易日的周期数

(10) DMA

释义	求动态移动平均
用法	<p>DMA(X, A), 求 X 的动态移动平均。</p> <p>算法: 若 $Y = \text{DMA}(X, A)$</p> <p>则 $Y = A * X + (1 - A) * Y'$, 其中 Y' 表示上一周期 Y 值, A 必须小于 1。</p>
示例	DMA(CLOSE, VOL/CAPITAL()) 表示求以换手率作平滑因子的平均价

(11) DRAWNULL

释义	取得一个无效数字
示例	IF(CLOSE > REF(CLOSE, 1), CLOSE, DRAWNULL) 表示下跌时分析图上不画线

(12) EMA

释义	求指数平滑移动平均
用法	EMA(X, N), 求 X 的 N 日指数平滑移动平均。算法: 若 $Y=EMA(X, N)$ 则 $Y=[2*X+(N-1)*Y']/(N+1)$, 其中 Y' 表示上一周期 Y 值
示例	EMA(CLOSE, 30) 表示求 30 日指数平滑均价

(13) FILTER

释义	过滤连续出现的信号
用法	FILTER(X, N): X 满足条件后, 删除其后 N 周期内的数据置为 0
示例	FILTER(CLOSE>OPEN, 5) 查找阳线, 5 天内再次出现的阳线不被记录在内

(14) HHV

释义	求最高值
用法	HHV(X, N), 求 N 周期内 X 最高值, N=0 则从第一个有效值开始
示例	HHV(HIGH, 30) 表示求 30 日最高价

(15) HHVBARS

释义	求上一高点到当前的周期数
用法	HHVBARS(X, N): 求 N 周期内 X 最高值到当前周期数, N=0 表示从第一个有效值开始统计
示例	HHVBARS(HIGH, 0) 求得历史新高到到当前的周期数

(16) HOD

释义	求高值名次
用法	HOD(X, N): 求当前 X 数据是 N 周期内的第几个高值, N=0 则从第一个有效值开始
示例	HOD(HIGH, 20) 返回是 20 日的第几个高价

(17) IMA

释义	求指数权重移动平均
用法	IMA(X, N, S) 求 X 的 N 日 S 系数权重的指数移动平均, S 如果小于 100 表示远期权重大于近期权重, 大于 100 表示近期的权重大于远期权重, 等于 100 就相当于 MA
示例	IMA(CLOSE, 10, 120) 表示求收盘价 10 日的 120%指数权重移动平均

(18) LLV

释义	求最低值
用法	LLV(X, N), 求 N 周期内 X 最低值, N=0 则从第一个有效值开始
示例	LLV(LOW, 0) 表示求历史最低价

(19) LLVBARS

释义	求上一低点到当前的周期数
用法	LLVBARS(X, N): 求 N 周期内 X 最低值到当前周期数, N=0 表示从第一个有效值开始统计
示例	LLVBARS(HIGH, 20) 求得 20 日最低点到当前的周期数

(20) LOD

释义	求低值名次
用法	LOD(X, N): 求当前 X 数据是 N 周期内的第几个低值, N=0 则从第一个有效值开始
示例	LOD(LOW, 20) 返回是 20 日的第几个低价

(21) MA

释义	求简单移动平均
用法	MA(X, N), 求 X 的 N 日移动平均值。算法: $(X1+X2+X3+\dots+XN)/N$
示例	MA(CLOSE, 10) 表示求 10 日均价

(22) VOLSTICK

释义	将数据画成柱状线
示例	VOL, VOLSTICK; 画成交量柱状线

(23) STACKVOLSTICK

释义	将数据画成叠加柱状线
示例	VOL, stackvolstick; 画成交量柱状线

(24) MEDIAN

释义	取若干指定周期的中位数据
----	--------------

示例	MEDIAN(X, N), 取 N 周期 X 的中数, 如果 N 是奇数, 取排完序的 X 中间一个元素; 如果偶数, 取中间两个的平均值返回
----	---

(25) NEWHBARS

释义	在历史上所有比当前数值高的数值序列中, 离当前第 N 个近的数字到当前的周期数
用法	NEWHBARS (X, N): 求高于当日 X 的第 N 个 x 的距离
示例	NEWHBARS (HIGH, 1) 求高于当日 h 的上一个 h 距离当前的周期数, 即, 今天的 h, 创了多少天以来的新高

(26) NEWLBARS

释义	在历史上所有比当前数值低的数值序列中, 离当前第 N 个近的数字到当前的周期数
用法	NEWLBAR (X, N): 求低于当日 X 的第 N 个 x 的距离
示例	NEWLBAR (LOW, 1) 求低于当日 1 的上一个 1 距离当前的周期数, 即, 今天的 1, 创了多少天以来的新低

(27) REF

释义	引用若干周期前的数据
用法	REF (X, A), 引用 A 周期前的 X 值
示例	REF (CLOSE, 1) 表示上一周期的收盘价, 在日线上就是昨收

(28) REFDATE

释义	引用自 1900 年以来指定日期的数据
用法	REFDATE (X, DATE[, TIME]), 引用 DATE 日期 TIME (可省略) 的 X 值
示例	REFDATE (CLOSE, 1011208) 表示 2001 年 12 月 08 日的收盘价; REFDATE (CLOSE, 1011208, 133030) 表示 2001 年 12 月 08 日 13:30:30 的收盘价 TIME 参数可省略使用, 省略时间一般用在日线及其以上周期使用, 对于日线以下周期则一般需要带时间参数。 注意: 对于逐 K 线运行模式, X 值不可以引用到未来数据, 但是序列模式则无此限

(29) REFV

释义	引用若干周期后的数据
用法	REFX (X, A), 引用 A 周期后的 X 值
示例	REFX (CLOSE, 1) 表示后一周期的收盘价, 在日线上就是明收

(30) RET

释义	按时间引用若干周期前的数据
用法	RET (X, A), 引用 A 周期时间前的 X 值
示例	RET (CLOSE, 10) 在日线上表示引用 10 天前的收盘价 (注意不是 10 周期前的)

(31) SFILTER

释义	过滤连续出现的信号
用法	SFILTER (X, COND): X 满足条件后, 将其后所有周期内的数据置为 0, 直到 COND 条件满足为止
示例	SFILTER (CLOSE>OPEN, CLOSE<OPEN) 查找阳线, 再次出现的阳线不被记录在内, 直到出现阴线为止

(32) SMA

释义	求移动平均
用法	SMA (X, N, M), 求 X 的 N 日移动平均, M 为权重。 算法: 若 $Y = SMA(X, N, M)$ 则 $Y = [M * X + (N - M) * Y'] / N$, 其中 Y' 表示上一周期 Y 值, N 必须大于 M
示例	SMA (CLOSE, 30, 1) 表示求 30 日移动平均价

(33) STKINDI

释义	引用任意品种任意周期的任意指标输出
用法	STKINDI (STKLABEL, INDINAME, CO, PERIOD[, m], n) STKLABEL 指定品种代码, 如为空表示当前品种 INDINAME 为指标公式调用 CO 为坐标轴类型 0 交易日坐标 1 自然日 2 交易交易时间 PERIOD 为周期类型, 有效值范围为 (0-19), 依次表示: 0: 分笔成交、1: 1 分钟、2: 5 分钟、3: 15 分钟、4: 30 分钟、5: 60 分钟、

	<p>6:日、7:周、8:月、9:年、10:多日、11:多分钟、12:多秒、</p> <p>13:多小时、14:季度线、15:半年线、16:节气线、17:3 分钟、18:10 分钟、19:多笔线</p> <p>m 为左右偏移周期个数（可选），0 表示引用当前数据，小于 0 为引用之前数据，大于 0 为引用之后数据</p> <p>n 为 0, 1, 2, 3, 4 表示复权数据类型, 0 为不复权, 1 为前复权, 2 为后复权, 3 为等比前复权, 4 为等比后复权</p>
示例	<p>STKINDI('1A0001', 'MA.MA1', 0, DATAPERIOD);</p> <p>计算 1A0001 的当前周期 MA 指标的 MA1 指标线</p> <p>STKINDI('', 'RSI.RSI1', 0, 6);</p> <p>计算当前品种的日线周期 RSI 指标的 RST1 指标线</p> <p>STKINDI('SH600000', 'RSI', 0, 6, -1);</p> <p>引用昨日 SH 市场 600000 品种的日线周期 RSI 指标最后一行输出并且使用公式的默认参数</p> <p>若参数为数字变量，那么需要有个字符串转换的过程</p> <p>s:=5;</p> <p>m:=NUMTOSTR(s, 0); //NUMTOSTR 函数将数字转换到字符串，再带入变量中</p> <p>vola:stkindi('if10', 'ATR.ATR', 0, 6, -1); //计算 IF10 合约的日线周期指标 ATR 的 ATR 指标线，传递参数 m 值为 5。</p> <p>//第 2 个参数''里的是文本，由三部分组成：'ATR.ATR('和 m 和')'，这三部分由两个连接符号&连接起来，实现对数值参数的文本传输。引用数据时，需要实现确认被引用品种周期数据齐全，再首次使用或者在不确定时，请手工进行数据补充工作</p>

(34) SUM

释义	求总和
用法	SUM(X, N), 统计 N 周期中 X 的总和, N=0 则从第一个有效值开始
示例	SUM(VOL, 0) 表示统计从上市第一天以来的成交量总和

(35) SUMBARS

释义	向前累加到指定值到现在的周期数
用法	SUMBARS (X, A) : 将 X 向前累加直到大于等于 A, 返回这个区间的周期数
示例	SUMBARS (VOL, CAPITAL) 求完全换手到现在的周期数

(36) TMA

释义	求递归移动平均
用法	TMA (X, N, M), 求 X 的递归移动平均, N、M 为权重
算法	若 $Y = TMA(X, N, M)$ 则 $Y = (N * Y' + M * X)$, 其中 Y' 表示上一周期 Y 值。初值为 $M * X$
示例	TMA (CLOSE, 0.9, 0.1) 表示求 X 的递归移动平均

(37) TODAYBAR

释义	求当日数据周期的数量
用法	TODAYBAR, 得到当日从开盘以来到现在的周期数量

(38) TR

释义	求真实波幅
用法	TR, 求真实波幅
示例	ATR:=MA (TR, 10) 表示求真实波幅的 10 周期均值

(39) TRMA

释义	三角移动平均
用法	TRMA (X, N) 求 X 在 N 周期内的三角移动平均 三角移动平均计算方法: 第 1 种可能先算 (奇数+1)/2 周期移动平均, 得出值再算这个值的 (奇数+1)/2 周期的移动平均。 第 2 种可能先算偶数/2 周期移动平均, 得出值再算这个值的 (偶数/2+1) 周期的移动平均, 也就是先判断一下 N 是奇数还是偶数, 然后再选对应的计算式。
示例	TRMA (CLOSE, 10) 求收盘价的 10 周期三角移动平均

(40) WMA

释义	求加权移动平均
用法	WMA(X, N), 求 X 的加权移动平均
算法	<p>若 $Y=WMA(X, N)$ 则</p> $Y = (N * X_0 + (N-1) * X_1 + (N-2) * X_2 + \dots + 1 * X_N) / (N + (N-1) + (N-2) + \dots + 1)$ <p>X_0 表示本周期值, X_1 表示上一周期值...</p>
示例	<p>WMA(CLOSE, 20)</p> <p>表示求 20 日加权均价</p>

2.4.10. 控制函数

(1) AND

释义	逻辑与运算
用法	<p>A AND B 或 A && B</p> <p>表示条件 A 与条件 B 同时成立</p>

(2) BEGIN

释义	把多条语句看作一条语句
用法	语法: BEGIN... END

(3) BREAK

释义	跳出循环
用法	语法: BREAK;

(4) DO

释义	执行语句
用法	<p>FOR VAR=N1 TO N2 DO EXPR;</p> <p>WHILE COND DO EXPR</p>

(5) EXIT

释义	终止公式执行. 注意: 逐 K 线模式下运行时, EXIT 的使用不当会导致比如 HHV, MA, 等统计性质的函数计算出现失误, 建议这些函数都要放在 EXIT 退出语句的前面保证每个周期都能执行到.
用法	语法: EXIT;

(6) FOR

释义	循环语句
用法	<p>FOR VAR=N1 TO N2 DO EXPR;</p> <p>从 VAR=N1 开始到 VAR=N2 开始循环执行 EXPR 语句, 每执行一次 VAR 加 1</p> <p>FOR VAR=N1 DOWNT0 N2 DO EXPR2;</p> <p>从 VAR=N1 开始到 VAR=N2 开始循环执行 EXPR 语句, 每执行一次 VAR 减 1</p> <p>注意: 逐 K 线计算模式时, 请尽量不要在 FOR 循环控制语句中使用 MA, HHV, LLV 等带有序列变量的统计类函数,</p> <p>虽然能通过编译, 但是会运行不正常, 应该把他们放在语句的外面计算</p>

(7) GLOBALVARIABLE

释义	申明并初始化超全局变量
用法	<p>GLOBALVARIABLE 除了不支持数组外其他使用均与 VARIABLE 普通全局变量相同,</p> <p>唯一不同之处在与 GLOBALVARIABLE 超全局变量在不会每次从头刷新指标数据时重新被刷新计算, 而是始终会记住最后一次被赋予的值。</p> <p>超全局变量一般会在客户停止后台程式化交易或者关闭框架图表后销毁重置。</p>

(8) IF

释义	根据条件求不同的值
用法	<p>IF (X, A, B) 若 X 不为 0 则返回 A, 否则返回 B</p> <p>例如: IF (CLOSE>OPEN, HIGH, LOW) 表示该周期收阳则返回最高值, 否则返回最低值</p>

(9) INPUT

释义	申明并设置参数
用法	<p>INPUT:PNAME1 (DFT, MIN, MAX, STEP), PNAME2 (DFT, MIN, MAX, STEP)...</p> <p>PNAME 表示参数名, DFT 表示缺省值</p> <p>MIN 表示最小值, MAX 表示最大值</p> <p>STEP 表示优化步长, 除 DEFAULT 外都可省略</p>
示例	INPUT:N(5), M(10, 1, 100, 2);

	表示定义参数 N, 缺省值为 5
	定义参数 M, 缺省值为 10, 最小值为 1, 最大值为 100, 优化步长为 2

(10) OR

释义	逻辑或运算
用法	<p>A OR B 或 A B</p> <p>表示条件 A 与条件 B 只要有一个成立即可</p>

(11) REPEAT

释义	循环语句
用法	<p>语法: REPEAT EXPR UNTIL COND</p> <p>循环执行语句 EXPR 直到满足 COND 条件的时候才中止</p> <p>注意: 逐 K 线计算模式时, 请尽量不要在 REPEAT 条件控制语句中使用 MA, HHV, LLV 等带有序列变量的统计类函数, 虽然能通过编译, 但是会运行不正常, 应该把他们放在语句的外面计算</p>

(12) VARIABLE

释义	<p>申明并初始化变量; variable 为全局变量申明语句, 一般用在逐 K 线计算模式下声明一个全局变量或者在序列模式和逐 K 线模式下声明一个内部数组。序列模式下 variable 申明的序列变量会被强制转换为常数, 故与普通的常数变量是没有区别的, 因此序列模式下的序列变量不要使用 variable 进行申明。</p>
语法	VARIABLE: VARNAME1=INITVALUE1, VARNAME2=INITVALUE2...;
示例	<p>VARIABLE: X=1, Y=CLOSE, ARR[10]=0, SARR[5]='STR';</p> <p>表示定义常数变量 X 并初始化为 1,</p> <p>申明序列变量 Y 并初始化为收盘价,</p> <p>申明含 10 个浮点数的数组并全部初始化为 0</p> <p>申明含 5 个字符串的数组并都初始化为 'STR'</p>

(13) WHILE

释义	循环语句
语法	<p>WHILE COND DO EXPR</p> <p>当满足 COND 条件的时候, 循环执行语句 EXPR</p>

注意:逐K线计算模式时,请尽量不要在 WHILE 条件控制语句中使用 MA, HHV, LLV
 等带有序列变量的统计类函数,
 虽然能通过编译,但是会运行不正常,应该把他们放在语句的外面计算

2.4.11. 指标函数

(1) SAR

释义	抛物转向
用法	SAR(N, S, M), N 为计算周期, S 为步长, M 为步长的极限值
示例	SAR(10, 2, 20) 表示计算 10 日抛物转向, 步长为 2%, 极限值为 20%

(2) SARTURN

释义	抛物转向点
用法	SARTURN(N, S, M), N 为计算周期, S 为步长, M 为极值, 若发生向上转向则返回 1, 若发生向下转向则返回-1, 否则为 0 其用法与 SAR 函数相同

2.4.12. 统计函数

(1) BETA2

释义	指定序列的贝塔系数
用法	BETA2(A1, A2, N); 求 A1, A2, N 周期的贝塔系数, 该系数表明 A1 每变动 1%, 则 A2 将变动 V%
示例	BETA2(INDEXC, CLOSE, 5) 表示大盘收盘价与该品种的 5 周期贝塔系数

(2) COVAR

释义	求 2 个序列的协方差
用法	COVAR(X1, X2, N) 为 X1 与 X2 的 N 周期的协方差
示例	COVAR(CLOSE, INDEXC, 8) 表示收盘价与大盘指数之间的 8 周期的协方差

(3) MODE

释义	数据集中出现最多的值
用法	返回在某一数组或数据区域中出现频率最多的数值。同 MEDIAN 一样, MODE 也是一个位置测量函数。

	<p>MODE(array, N); 求数据集 array 中 N 周期内出现频率最多的数值。</p> <p>如果数据集合中不含有重复的数据, 则 MODE 数返回错误值 N/A。</p>
--	--

(4) DEVSQ

释义	数据偏差平方和
用法	DEVSQ(X, N)

(5) PEARSON

释义	返回 Pearson (皮尔生) 乘积矩相关系数 r, 这是一个范围在 -1.0 到 1.0 之间 (包括 -1.0 和 1.0 在内) 的无量纲指数, 反映了两个数据集合之间的线性相关程度。
用法	PEARSON(A, B, N), 计算 A, B 序列的 N 周期乘积矩相关系数
示例	PEARSON(L, H, 10), 表示最低价与最高价的 10 周期乘积矩相关系数

(6) INTERCEPT

释义	利用现有的 x 值与 y 值计算直线与 y 轴的截距。截距为穿过已知的 known_x's 和 known_y's 数据点的线性回归线与 y 轴的交点。当自变量为 0 (零) 时, 使用 INTERCEPT 函数可以决定因变量的值。例如, 当所有的数据点都是在室温或更高的温度下取得的, 可以用 INTERCEPT 函数预测在 0° C 时金属的电阻
用法	INTERCEPT(Y, X, N), 求序列 Y, X 的线性回归线截距, Y 为因变的观察值或数据集合, X 为自变的观察值或数据集合
示例	INTERCEPT(L, H, 5), 表示计算最低价和最高价的 5 周期线性回归线截距

(7) KURT

释义	返回数据集的峰值。峰值反映与正态分布相比某一分布的尖锐度或平坦度。正峰值表示相对尖锐的分布。负峰值表示相对平坦的分布。
用法	KURT(X, N), 计算数据集 X 的 N 周期峰值
示例	KURT(C, 10), 表示收盘价的 10 周期峰值

(8) BINOMDIST

释义	一元二项式分布的概率值
用法	返回一元二项式分布的概率值。函数 BINOMDIST 适用于固定次数的独立试验,

	<p>当试验的结果只包含成功或失败二种情况，且当成功的概率在实验期间固定不变。例如，函数 BINOMDIST 可以计算三个婴儿中两个是男孩的概率。</p> <p>BINOMDIST (A, B, P, F), A 为试验成功的次数, B 为独立试验的次数, P 为每次试验中成功的概率, 为一逻辑值，用于确定函数的形式。如果 F 为 TRUE，函数 BINOMDIST 返回累积分布函数，即至多 A 次成功的概率；如果为 FALSE，返回概率密度函数，即 A 次成功的概率</p>
示例	BINOMDIST (6, 10, 0.5, FALSE), 10 次试验成功 6 次的概率为(0.205078)

(9) EXPONDIST

释义	<p>返回指数分布。使用函数 EXPONDIST 可以建立事件之间的时间间隔模型，例如，在计算银行自动提款机支付一次现金所花费的时间时，</p> <p>可通过函数 EXPONDIST 来确定这一过程最长持续一分钟的发生概率</p>
用法	<p>EXPONDIST (A, B, F), A 为函数的数值, B 为参数值, F 为一逻辑值，指定指数函数的形式。如果 F 为 TRUE，函数 EXPONDIST 返回累积分布函数；如果 F 为 FALSE，返回概率密度函数</p>

(10) FISHER

释义	Fisher 变换
用法	<p>返回点 x 的 Fisher 变换。该变换生成一个正态分布而非偏斜的函数。使用此函数可以完成相关系数的假设检验。</p> <p>FISHER(x)</p> <p>X 为一个数字，在该点进行变换。</p> <p>说明：</p> <p>如果 x 为非数值型，函数 FISHER 返回错误值 #VALUE!。</p> <p>如果 $x \leq -1$ 或 $x \geq 1$，函数 FISHER 返回错误值 #NUM!。</p>
示例	FISHER(0.75);在点 0.75 进行 Fisher 变换的函数值(0.972955)

(11) FISHERINV

释义	反 Fisher 变换
用法	<p>返回 Fisher 变换的反函数值。使用此变换可以分析数据区域或数组之间的相关性。如果 $y = \text{FISHER}(x)$，则 $\text{FISHERINV}(y) = x$。</p>

	<p>FISHERINV (y)</p> <p>Y 为一个数值，在该点进行反变换。</p> <p>说明：</p> <p>如果 y 为非数值型，函数 FISHERINV 返回错误值 #VALUE!。</p>
示例	FISHERINV(0.972955);在点 0.972955 进行 Fisher 变换的反函数值(0.75)

(12) HYPGEOMDIST

释义	<p>返回超几何分布。给定样本容量、样本总体容量和样本总体中成功的次数，函数 HYPGEOMDIST 返回样本取得给定成功次数的概率。</p> <p>使用函数 HYPGEOMDIST 可以解决有限总体的问题，其中每个观察值或者为成功或者为失败，且给定样本容量的每一个子集有相等的发生概率</p>
用法	HYPGEOMDIST (A, B, K, D), A 样本中成功的次数, B 样本容量, K 样本总体中成功的次数, D 样本总体的容量

(13) FTEST

释义	F 检验的结果
用法	<p>返回 F 检验的结果。F 检验返回的是当数组 1 和数组 2 的方差无明显差异时的单尾概率。可以使用此函数来判断两个样本的方差是否不同。例如，给定公立和私立学校的测试成绩，可以检验各学校间测试成绩的差别程度。</p>
语法	<p>FTEST(array1, array2, N)</p> <p>Array1 第一个数组或数据区域。</p> <p>Array2 第二个数组或数据区域。</p> <p>N 数组数据周期数量</p>
示例	<p>Array1={6, 7, 9, 15, 21} Array2={20, 28, 31, 38, 40} 返回上述数据的 F 检验结果(0.648318)</p> <p>FTEST(C, H, 10);返回收盘价和最高价 10 周期内的检验结果</p>

(14) LARGE

释义	数据集中第 k 个最大值
用法	<p>返回数据集中第 K 个最大值。使用此函数可以根据相对标准来选择数值。例如，可以使用函数 LARGE 得到第一名、第二名或第三名的得分</p>

语法	<p>LARGE (ARRAY, N, K)</p> <p>ARRAY 为需要从中选择第 K 个最大值的数组或数据区域。</p> <p>N 为数组的计算数据周期数量</p> <p>K 为返回值在数组或数据单元格区域中的位置（从大到小排）。</p> <p>说明</p> <p>如果数组为空，函数 LARGE 返回错误值 #NUM!。</p> <p>如果 $K \leq 0$ 或 K 大于数据点的个数，函数 LARGE 返回错误值 #NUM!。</p> <p>如果区域中数据点的个数为 N，则函数 LARGE (ARRAY, 1) 返回最大值，函数 LARGE (ARRAY, N) 返回最小值。</p>
示例	LARGE (C, 10, 3); 求收盘价的 10 周期内的第三个最大值

(15) FORCAST

释义	线性回归预测值
用法	FORCAST (X, N) 为 X 的 N 周期线性回归预测值
示例	FORCAST (CLOSE, 10) 表示求 10 周期线性回归预测本周期收盘价

(16) DRL

释义	计算回归偏离度
用法	DRL (X, N), 得到 X 的 N 周期回归偏离度
示例	DRL (C, 10), 表示求收盘价的 10 周期回归偏离度

(17) FORCAST2

释义	曲线回归预测值
用法	FORCAST2 (X, N) 为 X 的 N 周期曲线(方程: $y=a*x*x+b*x+c$) 回归预测值
示例	FORCAST2 (CLOSE, 10) 表示求 10 周期收盘价曲线回归预测本周期的值

(18) DRL2

释义	交易家数取指定市场分类中、有效交易家数
用法	NOLOT (MARKET, D) MARKET 为市场名称, D 为分类序号 (1 指数; 2A 股; 3B 股; 4 基金; 5 债券; 6 权证)
示例	DRL2 (C, 10), 表示求收盘价的 10 周期曲线回归偏离度的值 (%)

(19) NOLOT

释义	曲线回归偏离度
用法	DRL2(X, N), 得到 X 的 N 周期曲线回归偏离度
示例	NOLOT('SH', 2) 表示取上海 A 股的总交易家数 使用该函数前, 如果你是在开盘以后才接入, 那么请补充沪深股市分笔成交以后, 刷新扩展统计数据

(20) PERCENTILE

释义	返回区域中数值的第 K 个百分点的值。可以使用此函数来建立接受阈值。例如, 可以确定得分排名在第 90 个百分点之上的检测候选人。
用法	PERCENTILE (ARRAY, N, K) ARRAY 为定义相对位置的数组或数据区域。N 数组的数据周期数据量 K 0 到 1 之间的百分点值, 包含 0 和 1。 说明 如果 ARRAY 为空或其数据点超过 8,191 个, 函数 PERCENTILE 返回错误值 #NUM!。 如果 K 为非数字型, 函数 PERCENTILE 返回错误值 #VALUE!。 如果 K < 0 或 K > 1, 函数 PERCENTILE 返回错误值 #NUM!。 如果 K 不是 1/(N-1) 的倍数, 函数 PERCENTILE 使用插值法来确定第 K 个百分点的值。
示例	若 ARRAY={1, 3, 2, 4}, N=4, K=0.3, 上面列表中的数据在第 30 个百分点的值 (1.9)

(21) PERCENTRANK

释义	返回特定数值在一个数据集中的百分比排位。此函数可用于查看特定数据在数据集中所处的位置。例如, 可以使用函数 PERCENTRANK 计算某个特定的能力测试得分在所有的能力测试得分中的位置。
用法	PERCENTRANK (ARRAY, N, X, SIGNIFICANCE) ARRAY 为定义相对位置的数组或数字区域。N 为数组的周期数量大小; X 为数组中需要得到其排位的值, SIGNIFICANCE 表示返回的百分数值的有效位数。说明 : 如果数组为空, 函数 PERCENTRANK 返回错误值 #NUM!。 如果 SIGNIFICANE <

	1, 函数 PERCENTRANK 返回错误值 #NUM!。如果数组里没有与 X 相匹配的值, 函数 PERCENTRANK 将进行插值以返回正确的百分比排位。
示例	ARRAY={13, 12, 11, 8, 4, 3, 2, 1, 1, 1}, N=10, X=2, SIGNIFICANCE=3, 2 在上面数据列表中的百分比排位 (0.333), 因为该数据集中小于 2 的值有 3 个, 而大于 2 的值有 6 个, 因此为 $3/(3+6)=0.333$)

(22) PERMUT

释义	返回从给定数目的对象集合中选取的若干对象的排列数。排列为有内部顺序的对象或事件的任意集合或子集。排列与组合不同, 组合的内部顺序无意义。此函数可用于彩票抽奖的概率计算
用法	PERMUT (A, B), A 表示对象个数的整数, B 表示每个排列中对象个数的整数
示例	PERMUT (100, 3) 在上述条件下所有可能的排列数量 (970200)

(23) POISSON

释义	返回泊松分布。泊松分布通常用于预测一段时间内事件发生的次数, 比如一分钟内通过收费站的轿车的数量。
用法	POISSON (X, B, F), X 事件数, B 期望值, F 为一逻辑值, 确定所返回的概率分布形式。如果 F 为 TRUE, 函数 POISSON 返回泊松累积分布概率, 即, 随机事件发生的次数在 0 到 x 之间 (包含 0 和 1); 如果为 FALSE, 则返回泊松概率密度函数, 即, 随机事件发生的次数恰好为 x。

(24) QUARTILE

释义	返回数据集的四分位数。四分位数通常用于在销售额和测量数据中对总体进行分组。例如, 可以使用函数 QUARTILE 求得总体中前 25% 的收入值。
用法	<p>QUARTILE (ARRAY, N, QUART)</p> <p>ARRAY 为需要求得四分位数值的数据或数字型单元格区域。</p> <p>N 为数组数据周期数量</p> <p>QUART 决定返回哪一个四分位值。</p> <p>说明</p> <p>如果数组为空, 函数 QUARTILE 返回错误值 #NUM!。</p> <p>如果 QUART 不为整数, 将被截尾取整。</p>

	<p>如果 $QUART \leq 0$ 或 $QUART > 4$, 函数 QUARTILE 返回错误值 #NUM!。</p> <p>当 QUART 分别等于 0、2 和 4 时, 函数 MIN、MEDIAN 和 MAX 返回的值与函数 QUARTILE 返回的值相同。</p>
示例	若 $ARRAY = \{1, 2, 4, 7, 8, 9, 10, 12\}$, $N=8$, $QUART=1$, 上述数据的第一个四分位数(第 25 个百分点值) (3.5)

(25) RSQ

释义	返回根据 known_y's 和 known_x's 中数据点计算得出的 Pearson 乘积矩相关系数的平方。有关详细信息, 请参阅函数 REARSON。R 平方值可以解释为 y 方差与 x 方差的比例
用法	<p>$RSQ(A, B, N)$, 计算 A, B 序列的 N 周期乘积矩相关系数的平方。</p> <p>$RSQ(L, H, 10)$, 表示最低价与最高价的 10 周期乘积矩相关系数的平方</p>

(26) SKEW

释义	返回分布的偏斜度。偏斜度反映以平均值为中心的分布的不对称程度。正偏斜度表示不对称部分的分布更趋向正值。负偏斜度表示不对称部分的分布更趋向负值。
用法	$SKEW(C, 10)$, 表示求收盘价 10 周期的分布的偏斜度

(27) SLOPE

释义	线性回归斜率
用法	$SLOPE(X, N)$ 为 X 的 N 周期线性回归线的斜率
示例	$SLOPE(CLOSE, 10)$ 表示求 10 周期线性回归线的斜率

(28) SMALL

释义	返回数据集中第 K 个最小值。使用此函数可以返回数据集中特定位置上的数值。
用法	<p>$SMALL(ARRAY, N, K)$</p> <p>ARRAY 为需要找到第 K 个最小值的数组或数字型数据区域。</p> <p>N 为数组周期数量。</p> <p>K 为返回的数据在数组或数据区域里的位置(从小到大)。</p> <p>说明</p> <p>如果 ARRAY 为空, 函数 SMALL 返回错误值 #NUM!。</p> <p>如果 $K \leq 0$ 或 K 超过了数据点个数, 函数 SMALL 返回错误值 #NUM!。</p>

	如果 N 为数组中的数据点个数，则 SMALL (ARRAY, 1) 等于最小值， SMALL (ARRAY, N) 等于最大值。
示例	SMALL (C, 10, 2) ;返回收盘价的 10 周期内第 2 个最小值

(29) STANDARDIZE

释义	返回正态化数值
用法	STANDARDIZE (A, B, S), A 为需要进行正态化的数值, B 分布的算术平均值, S 为分布的标准偏差, 返回以 B 为平均值, 以 S 为标准偏差的分布的正态化数值。
示例	STANDARDIZE (42, 40, 1.5), 符合上述条件的 42 的正态化数值 (1.333333)

(30) STD

释义	估算标准差
用法	STD (X, N) 为 X 的 N 日估算标准差

(31) STDP

释义	总体标准差
用法	STDP (X, N) 为 X 的 N 日总体标准差

(32) STEYX

释义	返回通过线性回归法计算每个 x 的 y 预测值时所产生的标准误差。标准误差用来度量根据单个 x 变量计算出的 y 预测值的误差量。
用法	STEYX (Y, X, N), 计算 Y, X 序列变量的线性回归法预测标准误差。
示例	STEYX (L, H, 10), 表示最低价与最高价的 10 周期线性回归法预测标准误差

(33) TRIMMEAN

释义	返回数据集的内部平均值。函数 TRIMMEAN 先从数据集的头部和尾部除去一定百分比的数据点，然后再求平均值。当希望在分析中剔除一部分数据的计算时，可以使用此函数。
用法	TRIMMEAN (ARRAY, N, PERCENT) ARRAY 为需要进行整理并求平均值的数组或数值区域。 N 为数组数据周期数量 PERCENT 为计算时所要除去的数据点的比例，例如，如果 PERCENT = 0.2,

	<p>在 20 个数据点的集合中，就要除去 4 个数据点(20 X 0.2)：头部除去 2 个，尾部除去 2 个。</p> <p>说明</p> <p>如果 PERCENT < 0 或 PERCENT > 1，函数 TRIMMEAN 返回错误值 #NUM!。</p> <p>函数 TRIMMEAN 将除去的数据点数目向下舍入为最接近的 2 的倍数。如果 PERCENT = 0.1，30 个数据点的 10% 等于 3 个数据点。函数 TRIMMEAN 将对称地在数据集的头部和尾部各除去一个数据。 .</p>
示例	TRIMMEAN(C, 20, 0.2); 从计算中除去 20% 的 20 周期收盘价内部平均值。

(34) VAR

释义	估算样本方差
用法	VAR(X, N) 为 X 的 N 日估算样本方差.

(35) VARP

释义	总体样本方差
用法	VARP(X, N) 为 X 的 N 日总体样本方差

(36) WEIBULL

释义	返回韦伯分布。使用此函数可以进行可靠性分析，比如计算设备的平均故障时间
用法	WEIBULL(A, B, D, F), A 为参数值, B 为分布参数, D 为分布参数, F 为指明函数的形式, TRUE 为韦伯累积分布, FALSE 为韦伯概率密度

(37) AVEDEV

释义	平均绝对偏差
用法	AVEDEV(X, N)

(38) DELIVERYINTERVAL

释义	取当前品种在当前 K 线上，距离最近交割日的交易日数
用法	DELIVERYINTERVAL()

2.4.13. 时间函数

(1) BARSTATUS

释义	函数返回数据位置状态。
----	-------------

	1 表示第 1 根 K 线，2 表示最后 1 根 K 线，0 表示中间的 K 线。
用法	BARSTATUS

(2) BARPOS

释义	函数返回当前是第几根 K 线。对于日线数据就表示从上市到现在总共有多少交易日
用法	BARPOS

(3) CURRENTDATE

释义	计算时的当前日期
用法	CURRENTDATE 函数返回计算时的日期, 有效值范围为(101-1991231), 表示 1900/01/01-2099/12/31 注意: 该函数返回常数

(4) CURRENTTIME

释义	计算当前计算机时间
用法	CURRENTTIME 函数返回计算时的时间(时分秒), 有效值范围为(000000-235959) 注意: 该函数返回常数

(5) D1970TODATE

释义	取得 1970 日期 X 的日期值
用法	D1970TODATE (X)

(6) DATE

释义	取得该周期年月日的数值
用法	DATE () 函数返回有效值范围为(19000101-20991231), 表示 19000101-20991231

(7) DATEDIFF

释义	取得日期之间的时间间隔天数
用法	DATEDIFF (DATE1, DATE2) DATE1, DATE2 为序列变量或常数, 格式与 DATE 同, 有效值范围为

	(101-1991231), 表示 19000101-20991231 返回 DATE1、DATE2 两个日期之间的相差的天数， 如果 DATE1 晚于 DATE2，则 DATEDIFF 函数返回负数
示例	DATEDIFF(LSOLARTERMDATE(1), DATE) 表示求当年小寒到当前周期相差的天数

(8) DATEPOS

释义	取指定日期的数据序号
用法	DATEPOS (DATE), DATE 日期格式为一字符串格式，比如是标准日期时间格式， 并只能是常数
示例	DATEPOS ('2012-04-17 10:38:00'); 求 2012-04-17 10:38:00 日期时间的数据 对应图表上数据的序号

(9) DATETOD1970

释义	取得日期 X 距离 1970 年 1 月 1 日以来的天数
用法	DATETOD1970 (X)
示例	DATETOD1970 (DATE) 返回当天距离 1970 年 1 月 1 日的天数。 若日期小于 1970 年则返回对应负数。

(10) DAY

释义	取得该周期的日期
用法	DAY () 函数返回有效值范围为 (1-31)

(11) DAYOFWEEK

释义	取得指定日期的星期数
用法	DAYOFWEEK (D), D 为指定的日期 函数返回有效值范围为 (0-6)

(12) DAYS1970

释义	取得该周期从 1970 以来的天数。 返回自从 1970 年 1 月 1 日以来的天数，例如在 1971 年 1 月 1 日返回 365。
----	--

用法	DAYS1970
----	----------

(13) HOUR

释义	取得该周期的小时数。
用法	HOUR() 函数返回有效值范围为(0-23)，对于日线及更长周期此函数无效

(14) MINUTE

释义	取得该周期的分钟数
用法	MINUTE() 函数返回有效值范围为(0-59)，对于日线及更长周期此函数无效

(15) MONTH

释义	取得该周期的月份
用法	MONTH() 函数返回有效值范围为(1-12)

(16) OPENMINUTES

释义	取得开盘分钟数
用法	OPENMINUTES (TIME) 返回已开盘分钟数(0-总开盘分钟数), 开盘前的都为 1, 收盘后都为总开盘分钟数
示例	OPENMINUTES (CURRENTTIME), 在日线上可得到现在已开盘分钟数 OPENMINUTES (TIME), 在分钟线上可得到当时已开盘分钟数

(17) TOTOTIME

释义	取得从 0 点开始 X 秒后的时间值
用法	TOTOTIME (X)

(18) TIME

释义	取得该周期的时分秒
用法	TIME() 函数返回有效值范围为(000000-235959)，对于日线及更长周期此函数无效.

该函数返回序列变量

(19) TIME0

释义	取得该周期从当日 0 点以来的秒数。 返回自从当日 0 点以来的秒数，对于日线以上的分析周期返回 0
用法	TIME0

(20) TIMETOT0

释义	取得时间 X 距离当日 0 点的秒数
用法	TIMETOT0(X)

(21) WEEKDAY

释义	取得该周期的星期数
用法	WEEKDAY() 函数返回有效值范围为(0-6)

(22) YEAR

释义	取得该周期的年份
用法	YEAR() 函数返回有效值范围为(1900-2099)

2.4.14. 数学函数

(1) ABS

释义	求绝对值
用法	ABS(X) 返回 X 的绝对值
示例	ABS(-34) 返回 34

(2) ACOS

释义	反余弦值
用法	ACOS(X) 返回 X 的反余弦值

(3) ASIN

释义	反正弦值
----	------

用法	ASIN(X) 返回 X 的反正弦值
----	--------------------

(4) ATAN

释义	反正切值
用法	ATAN(X) 返回 X 的反正切值

(5) CEILING

释义	向数值增大方向舍入
用法	CEILING(A) 返回沿 A 数值增大方向最接近的整数
示例	CEILING(12.3) 求得 13, CEILING(-3.5) 求得-3

(6) COMBIN

释义	计算从给定数目的对象集合中提取若干对象的组合数。利用函数 COMBIN 可以确定一组对象所有可能的组合数。
用法	COMBIN(A, B), A 为对象的总数量, B 为每一组合中对象的数量
示例	COMBIN(8, 2), 从八个候选人中提取两个候选人的组合数(28)

(7) COS

释义	余弦值
用法	COS(X) 返回 X 的余弦值

(8) EXP

释义	指数
用法	EXP(X) 为 E 的 X 次幂
示例	EXP(CLOSE) 返回 E 的 CLOSE 次幂

(9) FLOOR

释义	向数值减小方向舍入
用法	FLOOR(A) 返回沿 A 数值减小方向最接近的整数
示例	FLOOR(12.3) 求得 12, FLOOR(-3.5) 求得-4

(10) FRACPART

释义	取得数据的小数部分
----	-----------

用法	FRACPART(X) 返回数值的小数部分
示例	FRACPART(12.3) 求得 0.3, FRACPART(-3.5) 求得-0.5

(11) INTPART

释义	绝对值减小取整，即取得数据的整数部分
用法	INTPART(A) 返回沿 A 绝对值减小方向最接近的整数
示例	INTPART(12.3) 求得 12, INTPART(-3.5) 求得-3

(12) LN

释义	求自然对数
用法	LN(X) 以 E 为底的对数
示例	LN(CLOSE) 求收盘价的对数

(13) LOG

释义	求以 10 为底的对数
用法	LOG(X) 取得 X 的对数
示例	LOG(100) 等于 2

(14) MAX

释义	求最大值
用法	MAX(A, B) 返回 A 和 B 中的较大值
示例	MAX(CLOSE-OPEN, 0) 表示若收盘价大于开盘价返回它们的差值，否则返回 0

(15) MIN

释义	求最小值
用法	MIN(A, B) 返回 A 和 B 中的较小值
示例	MIN(CLOSE, OPEN) 返回开盘价和收盘价中的较小值

(16) MOD

释义	求模运算
用法	MOD(A, B) 返回 A 对 B 求模
示例	MOD(26, 10) 返回 6

(17) POW

释义	乘幂
用法	POW(A, B) 返回 A 的 B 次幂
示例	POW(CLOSE, 3) 求得收盘价的 3 次方

(18) RAND

释义	随机整数
用法	RAND(N) 返回一个范围在 1-N 的随机整数
示例	CLOSE*(RAND(10)/10+0.4) 输出收盘价乘以[0.5-1.4]的随机系数

(19) REVERSE

释义	求相反数
用法	REVERSE(X) 返回-X
示例	REVERSE(CLOSE) 返回-CLOSE

(20) ROUND

释义	四舍五入为整数, 显示时不带小数
用法	ROUND(X) 将 X 四舍五入为整数
示例	ROUND(3.3) 求得 3, ROUND(3.5) 求得 4, ROUND(-3.5) 求得-4

(21) ROUNDS

释义	四舍五入整理小数到指定位数
用法	ROUNDS(A, B) 表示整理数字 A 的小数点位数到 B。该函数可用以做浮点数的精确相等判断
示例	ROUNDS(12.345, 2) 将返回 12.35

(22) SGN

释义	求符号值
用法	SGN(X), 当 X>0, X=0, X<0 分别返回 1, 0, -1

(23) SIN

释义	正弦值
用法	SIN(X) 返回 X 的正弦值

(24) SQRT

释义	开平方
用法	SQRT(X) 为 X 的平方根
示例	SQRT(CLOSE) 收盘价的平方根

(25) TAN

释义	正切值
用法	TAN(X) 返回 X 的正切值

2.4.15. 行情函数

(1) CLOSE

释义	取得该周期收盘价
用法	CLOSE

(2) C

释义	取得该周期收盘价，与 CLOSE 等价
用法	c

(3) HIGH

释义	取得该周期最高价
用法	HIGH

(4) H

释义	取得该周期最高价，与 HIGH 等价
用法	H

(5) LOW

释义	取得该周期最低价
用法	LOW

(6) L

释义	取得该周期最低价
用法	LOW

(7) AMOUNT

释义	取得该周期原始成交额
用法	AMOUNT

(8) OAMOUNT

释义	取得该周期原始成交额(未复权)
用法	OAMOUNT

(9) OCLOSE

释义	取得该周期原始收盘价(未复权)
用法	OCLOSE

(10) OHIGH

释义	取得该周期原始最高价(未复权)
用法	OHIGH

(11) OLOW

释义	取得该周期原始最低价(未复权)
用法	OLOW

(12) OOPEN

释义	取得该周期原始开盘价(未复权)
用法	OOPEN

(13) OPEN

释义	取得该周期开盘价
用法	OPEN

(14) 0

释义	取得该周期开盘价，与开盘价等价
用法	0

(15) OVOL

释义	取得该周期原始成交量(未复权)
用法	OVOL

(16) VOL

释义	取得该周期成交量
用法	VOL

(17) V

释义	取得该周期成交量，与 VOL 等价。
用法	V

(18) OPENA

释义	取得该周期开盘成交额。
用法	OPENA

(19) OPENV

释义	取得该周期开盘成交量。
用法	OPENV

(20) OPENINT

释义	取得该周期持仓量。
用法	OPENINT

(21) ASKPRICE

释义	取得该周期的委卖价。
用法	ASKPRICE

(22) BIDPRICE

释义	取得该周期的委买价。
用法	BIDPRICE

(23) ASKVOL

释义	取得该周期的委卖量。
----	------------

用法	ASKVOL
----	--------

(24) BIDVOL

释义	取得该周期的委买量。
用法	BIDVOL

(25) INDEXA

释义	取得同期大盘的成交额。
用法	INDEXA

(26) INDEXC

释义	取得同期大盘的收盘价。
用法	INDEXC

(27) INDEXH

释义	取得同期大盘的最高价。
用法	INDEXH

(28) INDEXL

释义	取得同期大盘的最低价。
用法	INDEXL

(29) INDEXO

释义	取得同期大盘的开盘价。
用法	INDEXO

(30) INDEXV

释义	取得同期大盘的成交量。
用法	INDEXV

(31) CAPITAL

释义	流通盘大小
用法	CAPITAL(), 返回流通盘大小, 单位为手。指数为 0

(32) SUSPEND

释义	是否停牌
用法	SUSPEND(), 无参数时默认返回后一个交易日是否停牌, 参数为-1 返回前一个交易日是否停牌, 参数为日期返回指定日期是否停牌 返回值为 1 时, 为停牌, 为 0 时为非停牌

2.4.16. 扩展数据

(1) EXTDATA

释义	获取指定周期的扩展数据
用法	EXTDATA (NAME, STOCK, TIME) 其中 NAME 是扩展数据的名字, STOCK 表示品种, TIME 表示相对当前周期的偏移值, 0 表示当前周期
示例	扩展数据名为 ZZZ, X := EXTDATA('ZZZ', 'SZ600000', 0), 则 X 表示 600000 在当前周期对应的 ZZZ 的值, 股票市场须用大写字母

(2) EXTDATARANGE

释义	获取扩展数据中满足指定条件的股票列表
用法	EXTDATARANGE (NAME, TIME, CONDITION) NAME 表示扩展数据的名字, TIME 表示相对当前周期的偏移值, 0 表示当前周期, CONDITION 为条件字符串
示例	扩展数据名为 ZZZ, LIST := EXTDATARANGE('ZZZ', 0, 'ZZZ == 10'), 则 LIST 为当前周期, ZZZ 的值等于 10 的股票代码的集合

(3) EXTDATAMATCH

释义	判断该周期当前品种的扩展数据是否满足指定条件
用法	EXTDATAMATCH (NAME, TIME, CONDITION) 表示当前品种在名为 NAME 的扩展数据中的 TIME 周期的值是否满足 CONDITION 条件
示例	扩展数据名为 ZZZ, X := EXTDATARANGE('ZZZ', 0, 'ZZZ == 10'), 则 X 表示当前品种当前周期 ZZZ 的值是否等于 10, 若等于 10, 则 X = 1, 否则 X = 0

(4) EXTDATARANK

释义	判断当前品种的扩展数据在所有品种中的排名，按扩展数据值从大到小排
用法	EXTDATARANK (NAME, TIME) 表示当前品种在名为 NAME 的扩展数据中的 TIME 周期的值在所有品种中的排名
示例	扩展数据名为 ZZZ，X:EXTDATARANK('ZZZ', 0, 'SZ000001') 表示当前品种 ('SZ000001') 当前周期 ZZZ 的值在 ZZZ 对应的股票集合中的排名

(5) EXTDATABIGGER

释义	返回扩展数据中比某个值大的股票数
用法	EXTDATABIGGER (NAME, TIME, VALUE) 表示当前品种在名为 NAME 的扩展数据中的 TIME 周期的值在所有品种中的排名
示例	扩展数据名为 ZZZ，X := EXTDATABIGGER('ZZZ', 0, 10)，表示扩展数据当前周期 ZZZ 的值比 10 大的数的个数

(6) EXTRANKTOVALUE

释义	返回扩展数据中排名为某一位置对应的数值
用法	EXTRANKTOVALUE (NAME, TIME, RANK) 表示当前品种在名为 NAME 的扩展数据中的 TIME 周期排名为 RANK 的对应数值
示例	扩展数据名为 ZZZ，X := EXTRANKTOVALUE('ZZZ', 0, 10)，表示扩展数据 ZZZ 当前周期排名为 10 的股票对应的值

2.4.17. 组合模型

(1) SETGROUPMAXHOLDING

释义	设置组合模型的最大持仓
用法	SETGROUPMAXHOLDING (50) 表示最多同时持有 50 支股票的仓位，默认值为 3

(2) SETGROUPMODE

释义	组合模型运行模式
用法	SETGROUPMODE (MODE) MODE 的取值为 0 或 1，0 表示纯脚本运行的模型，1 表示在 C++中控制买卖逻辑的运行方式，默认为 0

(3) GETSTOCKINFO

释义	获取个股的买卖点信息
用法	<p>STOCKINFO := GETSTOCKINFO(STOCK, 1); 表示取股票代码为 STOCK 的股票下一周期的买卖点信息, STOCK 为股票名称, 1 为周期偏移量 (取当前为 0, 向前取为负数, 向后取为正数)</p> <p>返回值是一个结构体, 该结构体有三个属性: BUY, SELL, PRICE;</p> <p>STOCKINFO.BUY > 0 表示该周期出现了买点</p> <p>STOCKINFO.SELL > 0 表示该周期出现了卖点</p> <p>STOCKINFO.PRICE 表示该周期该个股的收盘价, 可以用于计算买入价和卖出价</p> <p>STOCKINFO.HOLDING 表示该周期该个股的持仓, 可用于判断买卖点</p> <p>STOCKINFO.SUSPEND 表示该周期该个股是否停盘, STOCKINFO.SUSPEND=1 表示停盘, STOCKINFO.SUSPEND=0 表示未停盘</p>

(4) GETSTOCKINFOBYINDEX

释义	获取个股的买卖点信息
用法	用法和 GETSTOCKINFO 一样
示例	GETSTOCKINFOBYINDEX (STOCKID, 0) 表示取股票代码为 STOCKID 的股票的当前周期的买卖点信息

(5) ISSTOCKINHOLDING

释义	查询当前股票是否有持仓
示例	<p>ISSTOCKINHOLDING (STOCK) 表示股票代码为 STOCK 的股票是否在当前的持仓组合中。TRUE 表示在, FALSE 表示不在。</p> <p>ISSTOCKINHOLDING ('SZ000001') 表示当前持仓中是否有深圳平安</p>

(6) ISSTOCKINHOLDINGBYINDEX

释义	查询当前股票是否有持仓
用法	用法和 ISSTOCKINHOLDING 一样
示例	ISSTOCKINHOLDING (STOCKID) 表示股票代码为 STOCKID 的股票是否在当前的持仓组合中。TRUE 表示在, FALSE 表示不在。

(7) GETHOLDINGINFO

释义	获取个股的持仓信息
----	-----------

示例	<p> <code>HOLDING = GETHOLDINGINFO(STOCK)</code>; 表示获取当前持仓中, STOCK 所对应的股票的持仓信息, 返回值是一个结构体, 该结构体有五个属性: <code>HOLDING</code>, <code>BUYPRICE</code>, <code>BUYDATE</code>, <code>PROFIT</code>, <code>PRICE</code>, <code>HOLDINGPERIODS</code>; </p> <p> <code>HOLDING.HOLDING</code> 表示持仓手数 <code>HOLDING.BUYPRICE</code> 表示买入价格 <code>HOLDING.BUYDATE</code> 表示买入的时间点 <code>HOLDING.PROFIT</code> 表示从买入到当前的利润 <code>HOLDING.PRICE</code> 表示当前周期的收盘价 <code>HOLDING.HOLDINGPERIODS</code> 表示从买入到当前周期的周期数 </p>
----	---

(8) GETHOLDINGINFOBYINDEX

释义	获取个股的持仓信息
示例	<p>用法和 <code>GETHOLDINGINFO</code> 一样</p> <p> <code>HOLDING = GETHOLDINGINFOBYINDEX(STOCKID)</code>; 表示获取当前持仓中, STOCK 所对应的股票的持仓信息。 </p>

(9) GROUPBUY

释义	组合模型买入
示例	<p> <code>GROUPBUY(STOCK)</code> 表示在组合模型中买入股票 STOCK, 该操作会将该股票记入组合模型的持仓中, 并将股票加入该周期的买入面板中。 </p>

(10) GROUPBUYBYINDEX

释义	组合模型买入
用法	用法和 <code>GROUPBUY</code> 一样
示例	<p> <code>GROUPBUYINDEX(STOCKID)</code> 表示在组合模型中买入股票 STOCKID。 </p>

(11) GROUPELL

释义	组合模型卖出
用法	<p> <code>GROUPELL(STOCK)</code> 表示在组合模型中卖出股票 STOCK, 该操作会从组合模型的持仓中删除该股票, 并将股票加入该周期的卖出面板中。 </p> <p> <code>GROUPELL</code> 返回一个 <code>BOOL</code> 变量, 表示卖出是否成功, 比如在股票停牌时, 是无法卖出的 </p>

(12) GROUPELLBYINDEX

释义	组合模型卖出
用法	用法和 GROUPELL 一样
示例	GROUPELLBYINDEX (STOCKID) 表示在组合模型中卖出股票 STOCKID。

(13) GROUPOSSIBLEBUY

释义	组合模型计算买入备选
用法	GROUPOSSIBLEBUY (STOCK) 表示将 STOCK 加入组合模型该周期中的买入备选面板中，每周期买入的股票是买入备选股票集的一个子集。

(14) GROUPOSSIBLEBUYBYINDEX

释义	组合模型计算买入备选
用法	用法和 GROUPOSSIBLEBUY 一样 GROUPOSSIBLEBUYBYINDEX (STOCKID) 表示将 STOCKID 加入组合模型该周期中的买入备选面板中。

(15) GROUPOSSIBLESELL

释义	组合模型计算卖出备选
用法	GROUPOSSIBLESELL (STOCK) 表示将 STOCK 加入组合模型该周期中的卖出备选面板中，每周期卖出的股票是卖出备选股票集的一个子集。

(16) GROUPOSSIBLESELLBYINDEX

释义	组合模型计算卖出备选
用法	用法和 GROUPOSSIBLESELL 一样 GROUPOSSIBLESELLBYINDEX (STOCKID) 表示将 STOCK 加入组合模型该周期中的卖出备选面板中。

(17) INDYNAMICBASKET

释义	是否在动态股票篮子中
用法	INDYNAMICBASKET (STOCKID) 表示将 STOCKID 是否在动态股票篮子中，TRUE 表示在，FALSE 表示不在。

(18) SETGROUPINDEX

释义	当组合模型的运行模式为 C++ 模式时，设置这个参数，可以在买入备选中的股票数大于可能的最大持仓时，将买入备选中的股票根据扩展数据的排名排序，并买入排名靠前的股票。
用法	<p>组合模型中有下面的代码时</p> <pre>SETGROUPMAXHOLDING(50); SETGROUPMODE(1); SETGROUPINDEX('RISE');</pre> <p>则当某个周期持仓中已经有了 40 支股票，并且还有 30 支股票出现了买点时，组合模型会根据 RISE 中的排名信息，将买入备选中的 30 支股票进行排序，并买入排名靠前的 10 支股票</p>

(19) GETHOLDINGPROFIT

释义	获取组合模型当前周期持仓的浮动盈亏
用法	<code>X := GETHOLDINGPROFIT();</code> 则 X 表示当前周期组合模型持仓的浮动盈亏

2.4.18. 交易函数

(1) ORDER

释义	通过网络端口向指定的接口单发送交易信号
用法	<p>ORDER 的语法格式为 <code>ORDER(OPERATION, CHANNEL, ADDR);</code></p> <p>OPERATION 表示交易的类型，CHANNEL 为指定的接口单通道号，ADDR 为可选参数，不填表示使用系统默认的地址连接迅投交易终端，可以指定多个迅投交易终端中的接口单地址，多个地址间用逗号隔开</p>
示例	<code>ORDER(3, 1, '192.168.1.128:5000, 192.168.1.129:5000');</code> 表示向 192.168.1.128, 192.168.1.129 这两台机器上的 1 号接口单发送开空信号

(2) PASSORDER

释义	通过函数交易发送交易信号
用法	<p>PASSORDER 的语法格式为 <code>PASSORDER(OPTYPE, ORDERTYPE, ACCOUNTID, ORDERCODE, PRTYPE, PRICE, VOLUME[, strategyName])</code></p> <p>数据类型：OPTYPE, ORDERTYPE, PRTYPE, PRICE, VOLUME 是数字类型；ACCOUNTID, ORDERCODE 是字符串类型</p>



- (1) 最少填七个参数
- (2) 其中[strategyName] 自定义策略名不是必填
- (3) OPTYPE(操作类型):

OPTYPE 参数内容说明

期货六键:

- 0|开多
- 1|平昨多
- 2|平今多
- 3|开空
- 4|平昨空
- 5|平今空

期货四键:

- 6|平多, 优先平今
- 7|平多, 优先平昨
- 8|平空, 优先平今
- 9|平空, 优先平昨

期货两键:

- 10|卖出, 如有多仓, 优先平仓, 优先平今, 如有余量, 再开空
- 11|卖出, 如有多仓, 优先平仓, 优先平昨, 如有余量, 再开空
- 12|买入, 如有空仓, 优先平仓, 优先平今, 如有余量, 再开多
- 13|买入, 如有空仓, 优先平仓, 优先平昨, 如有余量, 再开多
- 14|买入, 不优先平仓
- 15|卖出, 不优先平仓

股票买卖:

- 23|股票买入
- 24|股票卖出

融资融券:

- 27|融资买入
- 28|融券卖出

29 | 买券还券

30 | 直接还券

31 | 卖券还款

32 | 直接还款

组合交易：

25 | 组合买入

26 | 组合卖出

37 | 卖出投资组合

40 | 期货组合开多

43 | 期货组合开空

46 | 期货组合平多, 优先平今

47 | 期货组合平多, 优先平昨

48 | 期货组合平空, 优先平今

49 | 期货组合平空, 优先平昨

(4) ORDERTYPE(下单类型)：

参数详细说明

11: 单股、单账号、普通、默认 方式下单, 兼容以前的 11 模式

1101: 单股、单账号、普通、股/手 方式下单(同 11)

1102: 单股、单账号、普通、金额(元) 方式下单(该方式只支持股票下单)

1113: 单股、单账号、总资产、比例(0~1) 方式下单

1123: 单股、单账号、可用、比例(0~1) 方式下单

12: 单股、账号组(无权重)、普通、默认 方式下单, 兼容以前的 12 模式

1201: 单股、账号组(无权重)、普通、股/手 方式下单(同 12)

1202: 单股、账号组(无权重)、普通、金额(元) 方式下单(该方式只支持股票下单)

1213: 单股、账号组(无权重)、总资产、比例(0~1) 方式下单

1223: 单股、账号组(无权重)、可用、比例(0~1) 方式下单



21:组合、单账号、普通、默认方式下单, 兼容以前的 21 模式

2101:组合、单账号、普通、按组合股票数量 方式下单, 对应 VOLUME 填篮子份数(同 21)

2102:组合、单账号、普通、按组合股票权重 方式下单, 对应 VOLUME 填金额(元)

2103:组合、单账号、普通、按账号可用方式下单, 对应 VOLUME 填比例(0~1) (该方式只支持股票组合)

22:组合、账号组(无权重)、普通、默认方式下单, 兼容以前的 22 模式

2201:组合、账号组(无权重)、普通、按组合股票数量 方式下单, 对应 VOLUME 填篮子份数(同 22)

2202:组合、账号组(无权重)、普通、按组合股票权重 方式下单, 对应 VOLUME 填金额(元)

2203:组合、账号组(无权重)、普通、按账号可用方式下单, 对应 VOLUME 填比例(0~1) (该方式只支持股票组合)

2331:组合、套利、合约价值自动套利、按组合股票数量 方式下单, 对应 VOLUME 填篮子份数

2332:组合、套利、按合约价值自动套利、按组合股票权重 方式下单, 对应 VOLUME 填金额(元)

2333:组合、套利、按合约价值自动套利、按账号可用方式下单, 对应 VOLUME 填比例(0~1)

组合套利对 ACCOUNTID 参数的约定:accountID :=
'stockAccountID,futureAccountID'

组合套利对 ORDERCODE 参数的约定:orderCode :='basketName,futureName'

对 PRICE 参数的约定为套利比例(0~1)

	<p>融资融券只支持 ORDERTYPE:11, 1101, 1102, 12, 1201, 1202</p> <p>(5) ACCOUNTID(账号 ID): 下单的账号 ID 或 账号组名; 组合套利时用逗号隔 开股票账号和期货账号</p> <p>(6) ORDERCODE(下单代码):</p> <p>两种情况: 单股或单期货、港股, 则该参数填合约代码; 组合交易, 则该参 数填篮子名称, 组合套利时用逗号隔开篮子名称和期货合约名</p> <p>(7) PRTYPE(下单选价类型):</p> <p>4 卖 1 价, 5 最新价, 6 买 1 价, 12 市价, 13 挂单价, 14 对手价, 11(指 定价) 模型价</p> <p>(8) PRICE(下单价格): 当 PRTYPE 是模型价(指定价) 11 时 PRICE 有效, 其它情 况下 PRICE 无效(注意: 组合套利时该参数作套利比例)</p> <p>(9) VOLUME: 下单数量(股 OR 手)、组合份数、资金比例 (0~1)</p>
示例	<p>PASSORDER(23, 1101, '6000000201', 'SH600000', 5, -1, 100); 表示发送账号 6000000201 以最新价买入 100 股 SH600000 的信号;</p> <p>PASSORDER(25, 2333, '6000000201, 037429', 'stockbasket, IF1703', 5, 1, 0.5); 表示以最新价买入账号 6000000201 的 50% 的可用资金的一篮子股票 stockbasket, 并以 100% 的套利比例在期货账号 037429 下开空单 IF1703</p>

(3) TACCOUNT

释义	获取指定账号的可用资金
用法	<p>TACCOUNT(1, '37500001');</p> <p>1 表示是期货账号, 2 为普通股票账号, 3 为信用账号, 37500001 是账号 ID, 返回可用资金;</p>

(4) HOLDING

释义	得到当前帐户持仓量, 多仓返回正数空仓返回负数
用法	<p>HOLDING(AccountID, MarketID, StockID, Direction);</p> <p>AccountID, MarketID, StockID 为字符串; Direction 为整型 (1 多, 2 空)</p>
示例	<p>例 1: ho:=holding('037055', 'IF', 'IF06', 2); //IF06 做空持仓</p> <p>例 2: ho:=holding('6000000255', 'SH', '600000', 1) //股票 600000 持仓</p>

(5) HOLDINGS

释义	取某资金帐号对应的持仓
用法	holdings(Account);表示获取帐号 Account 的持仓
示例	<p>统计某个账号所有品种做多方向的持仓</p> <pre>xxx := holdings('037055');</pre> <pre>loh := 0;</pre> <pre>for x in xxx do begin</pre> <pre> if x.direction = 48 then</pre> <pre> loh:= loh + x.volume;</pre> <pre>end</pre> <p>longhold:loh;</p> <p>xxx 为一个 positiondetail 结构体, 含有如下项:</p> <p>exchangeid 证券市场, 交易所代码</p> <p>exchangename 市场名字</p> <p>productid 品种代码</p> <p>productname 品种名称</p> <p>instrumentid 证券代码, 合约代码</p> <p>instrumentname 证券名称, 合约名称</p> <p>hedgeflag 投保</p> <p>direction 买卖</p> <p>opendate 成交日期</p> <p>tradeid 最初开仓位的成交</p> <p>volume 持仓量 当前持股</p> <p>openprice 开仓价</p> <p>tradingday 交易日</p> <p>margin 使用的保证金 历史的直接用 ctp 的, 新的自己用成本价*存量*系数 数算 股票不需要</p> <p>opencost 开仓成本 等于股票的成本价*第一次建仓的量, 后续减持不影响, 不算手续费 股票不需要</p> <p>settlementprice /结算价 对于股票的当前价</p>



closevolume 平仓量 等于股票已经卖掉的 股票不需要

closeamount 平仓额 等于股票每次卖出的量*卖出价*合约乘数(股票为1)的累加 股票不需要

dloatprofit 浮动盈亏 当前量*(当前价-开仓价)*合约乘数(股票为1)

closeprofit 平仓盈亏 平仓额 - 开仓价*平仓量*合约乘数(股票为1) 股票不需要

marketvalue 市值 合约价值

positioncost 持仓成本 股票不需要

positionprofit 持仓盈亏 股票不需要

lastsettlementprice 最新结算价 股票不需要

instrumentvalue 合约价值 股票不需要

istoday 是否今仓

xttag 迅投标签

stockholder 股东账号

frozensvolume 期货不用这个字段, 冻结数量

canusevolume 期货不用这个字段, 股票的可用数量

onroadvolume 期货不用这个字段, 股票的在途数量

yesterdayvolume 期货不用这个字段, 股票的股份余额

lastprice 结算价 对于股票的当前价

profitrate 持仓盈亏比例

futuretradetype 成交类型

expiredate 到期日, 逆回购用

comtradeid 套利成交 Id

legid 组合 Id

totalcost 自定义累计成本 股票信用用到

singlecost 自定义单股成本 股票信用用到

coveredvolume 用于个股期权

sideflag 用于个股期权, 标记 '0' - 权利, '1' - 义务, '2' - '备兑'

referencerate 汇率, 目前用于港股通

	structfundvol 分级基金可用（可分拆或可合并） redemptionvolume 分级基金可赎回量
--	--

(6) ORDERING (不生效)

释义	返回当前委托数量
用法	ordering(AccountID , MarketID, StockID, Direction, [strategyName]); AccountID, MarketID, StockID 为字符串; Direction 为整型(1 buy, 2 sell), strategyName 自定义策略名, 为字符串
示例	例 1: ho:=ordering('037055','IF','IF06',2); 例 2: ho:=ordering('6000000255','SZ','000001',1) 例 3: ho:=ordering('6000000255','SZ','000001',1, '策略1')

(7) ORDERINGS

释义	取某资金账号对应的委托信息
用法	orderings(AccountID, [strategyName]);
示例	统计某账号买入的所有的当天委托量 xxx := orderings('037055'); loo := 0; for x in xxx do begin if x.direction = 48 then loo:= loo + x.volumetotaloriginal; end longorder:loo xxx 为一个 orderdetail 结构体, 含有如下项: exchangeid 证券市场, 交易所代码 exchangenname 市场名字 productid 品种代码 productname 品种名称 instrumentid 证券代码, 合约代码 instrumentname 证券名称, 合约名称



sessionid
frontid 前端 id
orderref 下单引用 等于股票的内部委托号
orderpricetype 类型，例如市价单 限价单
direction 期货多空 股票买卖
offsetflag 期货开平，股票买卖其实就是开平
hedgeflag 投保
limitprice 限价单的限价，就是报价
volumetotaloriginal 最初委托量
ordersubmitstatus 提交状态
ordersysid 委托号
orderstatus 委托状态
volumetraded 已成交量
volumetotal 当前总委托量 股票不需要总委托量
errorid
errmsg 状态信息
taskid
frozenmargin 冻结保证金
frozencommission 冻结手续费
insertdate 日期
inserttime 时间
xttag 迅投标签
tradeprice 成交均价
cancelamount 已撤数量
optname 展示委托属性的中文
tradeamount 成交额 期货=均价*量*合约乘数
entrusttype 委托类别
cancelinfo 废单原因
undercode 标的证券

	<p>covereflag 备兑标记 '0' - 非备兑, '1' - 备兑</p> <p>orderpricermb 委托价格 人民币 用于港股通</p> <p>tradeamountrm 成交金额 人民币用于港股通</p> <p>referencerate 参考汇率 用于港股通</p>
--	---

(8) DEAL

释义	返回某个时间内的成交数量
用法	<pre>deal(AccountID , MarketID, StockID, Direction, [strategyName]);</pre> <p>或 <pre>deal(AccountID , MarketID, StockID, Direction, Senconds, [strategyName]);</pre></p> <p>AccountID, MarketID, StockID 为字符串; Direction 为整型 (1 多, 2 空), Senconds 为整型, 表示多少秒内; strategyName 自定义策略名, 为字符串</p>
示例	<p>例 1: <code>de:=deal('037055','IF','IF06',2);</code> // 返回当天 IF06 的 sell 的成交数量</p> <p>例 2: <code>de:=deal('037055','IF','IF06',2,90);</code> // 返回最近 90 秒内 IF06 的 sell 的成交数量</p> <p>例 3: <code>de:=deal('6000000255','SH','600004',2,90);</code> // 返回最近 90 秒内 600004 的 sell 的成交数量</p> <p>例 4: <code>de:=deal('6000000255','SH','600004',2,90,'策略1');</code> // 返回策略 1 最近 90 秒内 600004 的 sell 的成交数量</p>

(9) DEALS

释义	返回资金账号的成交信息
用法	<pre>deals(AccountID, [strategyName]);</pre>
示例	<p>返回某账号当天 buy 的成交量</p> <pre>xxx := deals('037055');</pre> <pre>dea:= 0;</pre> <pre>for x in xxx do begin</pre> <pre> if x.direction = 48 then</pre> <pre> dea:= dea + x.volume;</pre>



```
end  
longdeal:dea  
xxx 为一个 dealdetail 结构体, 含有如下项:  
    exchangeid 证券市场, 交易所代码  
    exchangename 市场名字  
    productid 品种代码  
    productname 品种名称  
    instrumentid 证券代码, 合约代码  
    instrumentname 证券名称, 合约名称  
    tradeid 成交编号  
    orderref 下单引用 等于股票的内部委托号  
    ordersysid 委托号  
    direction 买卖 股票不需要  
    offsetflag 开平 股票的买卖  
    hedgeflag 投保 股票不需要  
    price 成交均价  
    volume 成交量 期货单位手 股票做到股  
    tradedate 成交日期  
    tradetime 成交时间  
    comssion 手续费  
    tradeamount 成交额 期货=均价*量*合约乘数  
    taskid  
    xtag 迅投标签  
    orderpricetype 类型, 例如市价单 限价单  
    optname 展示委托属性的中文  
    entrusttype 委托类别  
    futuretradetype 成交类型  
    realoffsetflag 实际开平, 主要是区分平今和平昨  
    coveredflag 备兑标记 '0' - 非备兑, '1' - 备兑
```

	<p>closetodayvolume 平今量, 不显示</p> <p>orderpricermb 委托价格 人民币 用于港股通</p> <p>pricermb 目前用于港股通</p> <p>tradeamountrmb 目前用于港股通</p> <p>referencerate 汇率, 目前用于港股通</p> <p>xttrade 是否是迅投交易</p>
--	--

(10) LOADBASKET (返回结果和定义不符)

释义	读取导入指定定路径下的指定的交易篮子
用法	<p>loadbasket(dir,filename,discover) {filename 格式支持: xls、csv}</p> <p>discover:遇到同名篮子操作: 0 取消导入, 1 覆盖, 2 合并</p>
示例	<p>retv:=loadbasket('c:/test/', 'stockbasket.csv', 1) //该路径下存在对应文件返回 0, 否则返回-1</p> <p>或</p> <p>retv:=loadbasket('c:\\test\\', 'stockbasket.csv', 1) //该路径下存在对应文件返回 0, 否则返回-1</p>

(11) CANCEL

释义	针对委托号进行撤单
用法	CANCEL(AA)表示委托号为“AA”的委托

(12) STOPPRICE

释义	取个股, 期货, 期权涨停价, 跌停价
用法	<p>stopprice(select)或 stopprice(select , market, stockCode);</p> <p>其中 select 为整型 (1 跌停价, 2 涨停价), market, stockCode 为字符串.</p>
示例	<p>例 1: dd:=stopprice(1); //返回当前股票的跌停价</p> <p>例 2: dd:=stopprice(1, 'IF', 'IF06',); //指定 IF06, 返回 IF06 的跌停价</p>

(13) Contractmultiplier

释义	函数返回期货合约的乘数
用法	<p>contractmultiplier('') //取当前图合约的乘数</p> <p>contractmultiplier(contract)</p>

	contract 为合约代码，支持变量赋值
示例	contract:='cu06'; A:=contractmultiplier(contract); 将'cu06'这份合约的乘数赋值给 A

(14) MARKETVALUE

释义	取资金账号股票市值
用法	MARKETVALUE (2, 'AA') 表示返回帐号'AA'的股票市值，1 表示是期货账号，2 为普通股票账号，3 为信用账号，37500001 是账号 ID，返回可用资金

2.4.19. 系统函数

(1) PRINTOUT

释义	把变量输出到文件
用法	如:模型名是“新建模型 1” X : CLOSE; Y : HIGH; PRINTOUT (X,Y) 在安装目录的 BIN 子目录下生成名为: 新建模型 1.OUT

(2) ISEQUALV

释义	判断两个数是否相等
用法	相等返回 1，否则返回 0
示例	ISEQUALV (2, 2) 返回 1.

(3) ISGREATER

释义	判断是否大于
用法	ISGREATER (A,B)，如果 A 大于 B 返回 1，否则返回 0
示例	ISGREATER (5, 4)，返回 1

(4) ISGREATEREQUAL

释义	判断是否大于等于
用法	ISGREATEREQUAL (A,B)，如果 A 大于等于 B 返回 1，否则返回 0
示例	ISGREATEREQUAL (1, 1) 返回 1。

(5) ISLESS

释义	判断是否小于
用法	ISLESS (A, B), 如果 A 小于 B 返回 1, 否则返回 0
示例	ISLESS (1, 2) 返回 1.

(6) ISLESSEQUAL

释义	判断是否小于等于
用法	ISLESSEQUAL (A, B), 如果 A 小于等于 B 返回 1, 否则返回 0
示例	ISLESSEQUAL (2, 1), 返回 0.

(7) ISVALID

释义	判断是否有效
用法	ISVALID (X)
示例	ISVALID (2/0) 返回 0

(8) NOSORTED

释义	对集合中的元素不进行排序, 在不计算排名的情况下使用
用法	NOSORTED (\$ZZZ) 表示对股票篮子 ZZZ 不排序

(9) EXIST

释义	判断是否存在下标中
用法	EXIST (SELLMMM[], SELLRANK), 如果 SELLMMM 下标存在 SELLRANK 返回 1, 否则返回 0

(10) HOLDINGORNOT

释义	判断是否存在集合或数组中
用法	HOLDINGORNOT (A[], B), 如果 A 中存在 B 返回 1, 或者返回 0

(11) TOHOLD

释义	把股票加入到一个集合中
用法	TOHOLD (POSITION, STOCKID) 表示把股票 STOCKID 加入到 POSITION 中

(12) TOABANDON

释义	把股票从集合中删除
用法	TOABANDON (POSITION, STOCKID) 表示把股票 STOCKID 从 POSITION 中删除

(13) SETDATAALIGNMODE

释义	当模型引用了多个品种的数据时，该函数设置数据对齐的方式，该函数有 1 个参数，取值说明如下： 0：当前 K 线所有品种的数据到齐后才开始计算 1：当前 K 线只要主图品种的数据到了后就开始计算
用法	SETDATAALIGNMODE (1)

(14) SENDMAIL

释义	发送邮件，该函数有 7 个参数，取值说明如下： SERVERNAME smtp 或 pop 服务器地址 USERNAME 发送邮箱地址 PASSWORD 发送邮箱登陆密码 RCVNAME 接收者姓名 RCVADDR 接收者邮箱地址 SUBJECT 标题 CONTENT 内容
用法	SENDMAIL('smtp.thinktrader.net', 'zhangsan@thinktrader.net', '123456', 'lisi', 'lisi@thinktrader.net', 'say hello', 'Hello, lisi')

(15) TURN

释义	该函数只能支持以天为单位的日 K 线
用法	TURN('引用参数/引用字段', '移动周期') TURN('', 0) 返回界面显示股票的换手率 TURN('STOCKCODE', 'T') 返回股票代码为 STOCKCODE 在 time 个周期之后的换手率，T 为正，返回 T 周期之后的换手率。 T 为负数，返回 T 个周期之前的换手率。
示例	TURN('SH600000', -1) ; 返回浦发银行，一个周期之前的换手率。

(16) gettreasury10y

释义	获取指定日期的十年期国债到期收益率基准数值
用法	gettreasury10y(date) 第一个有效数据到当前的天数
示例	gettreasury10y('20170306'); 取得 2017 年 3 月 5 日的无风险收益率

2.4.20. 附加函数

(1) OPENMINUTES

释义	输出开盘时间
用法	OPENMINUTES()

(2) CONST

释义	取常数值
用法	CONST(A) 返回 A 的常数值

(3) ISBUYORDER

释义	是否主动性买单
用法	ISBUYORDER() 仅用于分笔图

(4) ISSELLORDER

释义	是否主动性卖单
用法	ISSELLORDER() 仅用于分笔图

(5) BUYVOL

释义	主动性买单量
用法	BUYVOL() 仅用于分笔图, 非主动性买单返回 0

(6) SELLVOL

释义	主动性卖单量
用法	SELLVOL() 仅用于分笔图, 非主动性卖单返回 0

(7) UPNDAY

释义	返回是否连涨周期数
用法	UPNDAY(CLOSE, 2) 判断是否连续两周期收盘价连涨

(8) DOWNNDAY

释义	返回是否连跌周期数
用法	DOWNNDAY(CLOSE, 2) 判断是否连续两周期收盘价连跌

(9) NDAY

释义	判断是否多周期一直有某式成立
用法	NDAY(CLOSE>OPEN, 2) 判断是否连续两周期收盘价大于开盘价

(10) TICKVOLDISTRIBUTION

释义	返回固定时间范围内成交量分布中超过某个占比对应的最优 VOL
用法	TICKVOLDISTRIBUTION(seconds, ratio, direction), seconds:秒数, ratio:总笔数占比(0~1], direction:0 all, 1 buy, 2 sell
示例	TICKVOLDISTRIBUTION(120, 0.4, 1) //主动性买盘 120 秒内占总笔数 40%的最优 vol 值

(11) CONTRACT

释义	获取本地合约代码对应的交易所合约代码
用法	TICKVOLDISTRIBUTION(seconds, ratio, direction), seconds:秒数, ratio:总笔数占比(0~1], direction:0 all, 1 buy, 2 sell
示例	TICKVOLDISTRIBUTION(120, 0.4, 1) //主动性买盘 120 秒内占总笔数 40%的最优 vol 值