

RedCrab

The Calculator

Neuheiten in der Bedienungsanleitung Version 4.10

copyright © by Redchillicrab, Singapore 2009, 2010, 2011,2012

New Functions / Operators

AddIn

The operator **AddIn** returns a data field with a linear series of numbers. The length of the field corresponds to the left operand. The first number is taken from the first element of the left operand. The following values are incremented by the value of the right operand.

Example: `[4..8] addin 0.5 = 4.0 4.5 5.0 5.5 6.0`

The left operand's first element defines the start value; the following values have no significance. They are place holders to determine the size of the field. The following example shows therefore the same result as above.

Example: `[4..0] addin 0.5 = 4.0 4.5 5.0 5.5 6.0`

A descending sequence of numbers generated by a negative operand.

Example: `[4..8] addin -0.5 = 4.0 3.5 3.0 2.5 2.0`

AddTo

The operator **AddTo** returns a data field that contains a linear series of numbers. The series begins with the first value of the left operand and is incremented by the value of the right operand to the next value of the left operand.

Example: `[4, 7] addto 0.5= 4.0 4.5 5.0 5.5 6.0 6.5 7.0`

`[4, 1] addto 0.5= 4.0 3.5 3.0 2.5 2.0 1.5 1.0`

`[4, 6, 5] addto 0.5= 4.0 4.5 5.0 5.5 6.0 5.5 5.0`

AddOn

The function **AddOn** adds any value of a continuous series of numbers to the next value.

Example: `addon([10, 30, 25, -10, 15]) = 10 40 65 55 70`

MulIn

The operator **MulIn** returns a data field with a logarithmic series of numbers. The number of values corresponds to the numbers of the left operand. The first number is taken from the first element of the left operand. The following values are each the product of multiplication by the right operand.

Example: `[2..8] mulin 2.0= 2 4 8 16 32 64 128`

The left operand's first element defines the start value; the following values have no significance. They are place holders to determine the size of the field. The following example shows therefore the same result as above.

Example: `[2..-4] mulin 2.0= 2 4 8 16 32 64 128`

Descending sequence of numbers generated by a value < 1 .

`[2..8] mulin 0.5= 2.0 1.0 0.50 0.25 0.13 0.06 0.03`

MulTo

The operator ***MulTo*** returns a data field with a series of logarithmic numbers. The series begins with the first value of the left operand and is gradually multiplied by the value of the right operand till the value of the next operand is reached.

Example: `[1,150] multo 2 = 1 2 4 8 16 32 64 128`

`[150,3] multo 0.5 = 150 75 37.5 18.8 9.38 4.69`

For ascending order, right operand must be $>$, for a descending order the right operand must be < 1 . Negative values and the values 0 and 1 are not allowed.

MulAd

The operator ***MulAd*** returns a series of numbers where each value of the left operand first multiplied by the right operand, and added to the next value.

Example:

`x = [1..5] fill 100 =100.00 100.00 100.00 100.00 100.00`
`y = x mulad 1.1 = 110.00 231.00 364.10 510.51 671.56`

The right operand can be a one-dimensional field.

Patt

The result of the operator ***Patt*** is a data field the size of the left operand. The field contains continuous values of the right field operand.

Example: $x = [1..10] \text{ patt } [1, 1, 2] = 1 \ 1 \ 2 \ 1 \ 1 \ 2 \ 1 \ 1 \ 2 \ 1$

New Discription

Fill

The result of the operator ***Fill*** is a data field the size of the left operand. The field contains values specified by the right operand.

Example: $x = [1..5] \text{ fill } 8 = 8 \ 8 \ 8 \ 8 \ 8$