/MITS = *maxIterations*	Keeps returned values small by specifying a small number for *maxIterations*. *maxIterations* must be a value between 1 and 32767 (default is 100).
/V[=*v*]	Prints output variables to history. *v*=1: Prints variables (same as /V). *v*=0: Nothing printed (same as no /V).

Details

The ratio is computed by continued fraction expansion and recurrence relations for the convergents and checking num - (V_numerator/V_denominator) against *maxError*.

Setting *maxError* = 0 computes a maximally accurate ratio. The returned values can be surprisingly large:

```
RatioFromNumber/V/MERR=0 (1/1666)
  V_numerator= 4398046511104; V_denominator= 7.3271454874993e+15;
  ratio= 0.00060024009603842; V_difference= 0;
```

Using the default /MERR returns the expected 1 and 1666. The difference is attributable to floating-point roundoff errors.

The ratio is computed by continued fraction expansion and recurrence relations for the convergents and checking *num* - (V_numerator/V_denominator) against /MERR.

Output Variables

RatioFromNumber sets the following output variables:

`V_difference`	V_numerator/V_denominator - *num* (positive if the approximation is too big).
`V_flag`	0: V_difference less than or equal to /MERR. 1: V_difference greater than /MERR.
`V_numerator,V_denominator`	Values for the numerator and denominator. The ratio of V_numerator/V_denominator approximates *num*.
`V_iterations`	The number of iterations actually used.

RatioFromNumber prints the output variables if you specify /V or /V=1 but only when running in the main thread.

Examples

```
RatioFromNumber/V pi
  V_numerator= 355; V_denominator= 113; ratio= 3.141592920354;
  V_difference= 2.667641894049 7e-07; V_iterations= 3;

RatioFromNumber/V/MITS=2 pi
  V_numerator= 22; V_denominator= 7; ratio= 3.1428571428571;
  V_difference= 0.0012644892673497; V_iterations= 1;
```

See Also

gcd, **trunc**, **PrimeFactors**

Rect

The Rect structure is used as a substructure usually to store the coordinates of a window or control.

```
Structure Rect
   Int16 top
   Int16 left
   Int16 bottom
   Int16 right
EndStructure
```

RectF

The RectF structure is the same as Rect but with floating point fields.