

Support for all TeXmacs mathematical symbols.

```
(1) -> a1:=["αβγδεεζη",
            "θθικλμνξ",
            "οππρρστ",
            "υφφχψω....."]
["αβγδεεζη", "θθικλμνξ", "οππρρστ", "υφφχψω....."]
Type: List(String)
```

```
(2) -> matrix(map(entries,a1))$OutputForm
( α β γ δ ε ε ζ η )
( θ θ ι κ λ μ ν ξ )
( ο π π ρ ρ σ ς τ )
( υ φ φ χ ψ ω ... .. )
Type: OutputForm
```

```
(3) -> a2:=["ΓΔΘΛΞΠΣΥ",
            "ΦΨΩ....."]
["ΓΔΘΛΞΠΣΥ", "ΦΨΩ....."]
Type: List(String)
```

```
(4) -> matrix(map(entries,a2))$OutputForm
( Γ Δ Θ Λ Ξ Π Σ Υ )
( Φ Ψ Ω ... .. )
Type: OutputForm
```

```
(5) -> b1:=["0123456789"]
["0123456789"]
Type: List(String)
```

```
(6) -> matrix(map(entries,b1))$OutputForm
( 0 1 2 3 4 5 6 7 8 9 )
Type: OutputForm
```

```
(7) -> b2:=["abcde fghijklm",
            "nopqrstuvwxyz",
            "ABCDEFGHIJKLM",
            "NOPQRSTUVWXYZ"]
["abcde fghijklm", "nopqrstuvwxyz", "ABCDEFGHIJKLM", "NOPQRSTUVWXYZ"]
Type: List(String)
```

```
(8) -> matrix(map(entries,b2))$OutputForm
( a b c d e f g h i j k l m )
( n o p q r s t u v w x y z )
( A B C D E F G H I J K L M )
( N O P Q R S T U V W X Y Z )
```

Type: OutputForm

(9) -> b3:=["abcdefghijklm",
"nopqrstuvwxyz",
"ABCDEFGHJKLM",
"NOPQRSTUVWXYZ"]

["abcdefghijklm", "nopqrstuvwxyz", "ABCDEFGHJKLM", "NOPQRSTUVWXYZ"]
Type: List(String)

(10) -> matrix(map(entries,b3))\$OutputForm

$$\begin{pmatrix} a & b & c & d & e & f & g & h & i & j & k & l & m \\ n & o & p & q & r & s & t & u & v & w & x & y & z \\ A & B & C & D & E & F & G & H & I & J & K & L & M \\ N & O & P & Q & R & S & T & U & V & W & X & Y & Z \end{pmatrix}$$

Type: OutputForm

(11) -> b4:=["αβγδεεζηθθικλμν",
"ξοπρρσςστυφφχψω",
"ΓΔΘΛΞΣΥΦΨΩ....."]

["αβγδεεζηθθικλμν", "ξοπρρσςστυφφχψω", "ΓΔΘΛΞΣΥΦΨΩ....."]
Type: List(String)

(12) -> matrix(map(entries,b4))\$OutputForm

$$\begin{pmatrix} \alpha & \beta & \gamma & \delta & \epsilon & \epsilon & \zeta & \eta & \theta & \theta & \iota & \kappa & \lambda & \mu & \nu \\ \xi & \omicron & \pi & \varpi & \rho & \rho & \sigma & \varsigma & \tau & \upsilon & \phi & \varphi & \chi & \psi & \omega \\ \Gamma & \Delta & \Theta & \Lambda & \Xi & \Sigma & \Upsilon & \Phi & \Psi & \Omega & \dots & \dots & \dots & \dots & \dots \end{pmatrix}$$

Type: OutputForm

(13) -> c1:=["ABCDEFGHIJKLM",
"NOPQRSTUVWXYZ"]

["ABCDEFGHIJKLM", "NOPQRSTUVWXYZ"]
Type: List(String)

(14) -> matrix(map(entries,c1))\$OutputForm

$$\begin{pmatrix} A & B & C & D & E & F & G & H & I & J & K & L & M \\ N & O & P & Q & R & S & T & U & V & W & X & Y & Z \end{pmatrix}$$

(14) -> f1:=["abcdefghijℓm",
"nopqrstuvwռդ",
"ᲐᲑᲒᲓᲔᲕᲖᲗᲘᲙᲚ",
"ᲛᲜᲝᲞᲟᲠᲡᲢᲣᲤᲥᲦᲧᲨᲩᲪᲫᲬᲭᲮᲯᲰᲱᲲᲳᲴᲵᲶᲷᲸᲹᲺ᲻᲼ᲽᲾᲿ"]

Type: OutputForm

(15) -> matrix(map(entries,f1))\$OutputForm

["abcdefghijℓm", "nopqrstuvwռդ", "ᲐᲑᲒᲓᲔᲕᲖᲗᲘᲙᲚ", "ᲛᲜᲝᲞᲟᲠᲡᲢᲣᲤᲥᲦᲧᲨᲩᲪᲫᲬᲭᲮᲯᲰᲱᲲᲳᲴᲵᲶᲷᲸᲹᲺ᲻᲼ᲽᲾᲿ"]
Type: List(String)

(16) -> bb1:=["abcdefghijklm",
"nopqrstuvwxyz",
"ABCDEFGHJKLM",
"NOPQRSTUVWXYZ"]

$$\begin{pmatrix} a & b & c & d & e & f & g & h & i & j & k & l & m \\ n & o & p & q & r & s & t & u & v & w & x & y & z \\ \mathfrak{A} & \mathfrak{B} & \mathfrak{C} & \mathfrak{D} & \mathfrak{E} & \mathfrak{F} & \mathfrak{G} & \mathfrak{H} & \mathfrak{I} & \mathfrak{J} & \mathfrak{K} & \mathfrak{L} & \mathfrak{M} \\ \mathfrak{N} & \mathfrak{O} & \mathfrak{P} & \mathfrak{Q} & \mathfrak{R} & \mathfrak{S} & \mathfrak{T} & \mathfrak{U} & \mathfrak{V} & \mathfrak{W} & \mathfrak{X} & \mathfrak{Y} & \mathfrak{Z} \end{pmatrix}$$

Type: OutputForm

["abcdefghijklm", "nopqrstuvwxyz", "ABCDEFGHIJKLM", "NOPQRSTUVWXYZ"]

Type: List(String)

(18) -> `matrix(map(entries,bb1))$OutputForm`

$$\begin{pmatrix} a & b & c & d & e & f & g & h & i & j & k & l & m \\ n & o & p & q & r & s & t & u & v & w & x & y & z \\ A & B & C & D & E & F & G & H & I & J & K & L & M \\ N & O & P & Q & R & S & T & U & V & W & X & Y & Z \end{pmatrix}$$

Type: OutputForm

All of the above except **0123456789** are also variable names (symbols):

(19) -> $\alpha\beta\gamma\delta\epsilon\zeta\eta$

$\alpha\beta\gamma\delta\epsilon\zeta\eta$

Type:

Variable(<alpha><beta><gamma><delta><varepsilon><epsilon><zeta><eta>)

(20) -> $\theta\iota\kappa\lambda\mu\nu\xi$

$\theta\iota\kappa\lambda\mu\nu\xi$

Type:

Variable(<theta><varthetaeta><iota><kappa><lambda><mu><nu><xi>)

(21) -> $\omicron\pi\rho\rho\sigma\tau$

$\omicron\pi\rho\rho\sigma\tau$

Type:

Variable(<omicron><pi><varpi><rho><varrho><sigma><varsigma><tau>)

(22) -> $\upsilon\phi\chi\psi\omega$

$\upsilon\phi\chi\psi\omega$

Type:

Variable(<upsilon><phi><varphi><chi><psi><omega>)

(23) -> $\Gamma\Delta\Theta\Lambda\Xi\Pi\Sigma\Upsilon$

$\Gamma\Delta\Theta\Lambda\Xi\Pi\Sigma\Upsilon$

Type:

Variable(<Gamma><Delta><Theta><Lambda><Lambda><Xi><Pi><Sigma><Upsilon>)

(24) -> $\Phi\Psi\Omega$

$\Phi\Psi\Omega$

Type:

Variable(<Phi><Psi><Omega>)

(25) -> $abcdefghjklm$

abcdefghijklm

Type: Variable(<b-a><b-b><b-c><b-

d><b-e><b-f><b-g><b-h><b-j><b-k><b-l><b-m>)

(26) -> *abcdefghijklm*

abcdefghijklm

Type: Variable(<b-up-a><b-up-b><b-up-

c><b-up-d><b-up-e><b-up-f><b-up-g><b-up-h><b-up-i><b-up-j><b-up-k><b-up-l><b-up-m>)

(27) -> $\alpha\beta\gamma$

$\alpha\beta\gamma$

Type: Variable(<b-alpha><b-

beta><b-gamma>)

(28) -> *ABCDE*

ABCDE

Type: Variable(<cal-A><cal-

B><cal-C><cal-D><cal-E>)

(29) -> *abℳ*

abℳ

Type: Variable(<frak-a><frak-

b><frak-A><frak-B>)

(30) -> *ab*

ab

Type: Variable(<bbb-a><bbb-

b>)

We can make operators look prettier:

(31) -> *sin(x)*

sin(x)

Type: Expression(Integer)

(32) -> *sin(x)*

sin(x)

Type: Expression(Integer)

These should really be infix operators, not symbols:

(33) -> \oplus

oplus

Type: Variable(oplus)

(34) -> \otimes

otimes

Type: Variable(otimes)

Should π be converted to FriCAS %pi?

(35) -> $C := 2\pi r$

$2r\pi$

Type: Polynomial(Integer)

(36) -> eval(C, [<pi>=%pi,r=1])

2π

Type: Expression(Integer)

(37) -> %::Float

Cannot convert from type Expression(Integer) to Float for value
 2π

(37) -> $2*\%pi+x$

$x+2\pi$

Type: Expression(Integer)

(38) -> eval(%,x=1)

$2\pi+1$

Type: Expression(Integer)

(39) -> %::Pi

Cannot convert from type Expression(Integer) to Pi for value
 $2\pi+1$

(39) -> $2*\%pi+1$

$2\pi+1$

Type: Pi

(40) -> %::Float

7.2831853071_795864769

Type: Float

(41) -> $\sum_{i=1..10} i$

55

Type: Fraction(Polynomial(Integer))

Every upper and lower case Latin letter has 5 variants

(42) -> i, i, i, i, i

```
[i, i, i, i]
<frac-i>))
(44) ->
```

```
Type: Tuple(OrderedVariableList([i, <b-i>, <b-up-i>, <frac-i>]))
```