MuPADで 高校の数学を

－ 長野県高校視聴覚教育研究会において －

生越 茂樹（おごせ しげき）

2007年 10月 19日

第1章　平面のグラフ

1　y=f(x) のグラフ (plotfunc2d)

$y=f(x)$ のグラフ $(a \leqq x \leqq b)$	$\texttt{plotfunc2d(f(x), x = a ..b)}$
$y=f(x), y=g(x), \cdots$ のグラフ $(a \leqq x \leqq b)$	$\texttt{plotfunc2d(f(x), g(x),} \cdots \texttt{, x = a ..b)}$
$y=f(x)$ のアニメーション	$\texttt{plotfunc2d(f(x, t), x = a ..b, t = t}_1\texttt{..t}_2\texttt{)}$

$y=f(x)$ のグラフを描くには plotfunc2d を使うのがもっとも簡単です．(二つ以上のグラフを描くときは，コンマで区切ります.)
y の範囲も $c \leqq y \leqq d$ のように指定したいときは次のようにします.

$$\texttt{plotfunc2d(f(x), x = a ..b, ViewingBoxYRange = c ..d)}$$

注1) default では, MuPAD は 自動的に縦と横の倍率を変えます．これが邪魔な時は「Scaling=Constrained」をつけます．(default は 「Scaling=UnConstrained」 です.) 「ViewingBoxYRange」や「Scaling」のような option のことを, MuPAD では *Attribute* と呼びます.

アニメーションは パラメーターを一つ増やすだけです．MuPAD は 自動的に そのパラメータを **アニメーションパラメータ**と解釈します．アニメーションの *Attribute* は，Frames，TimeRange(動作時間, 単位は秒) などいろいろあります．default では, Frames= 50,TimeRange= 0..10 で 1 秒あたりのコマ数は，$\frac{50\ \text{コマ}}{10\ \text{秒}} = 5$ コマ/秒 となっています．また「TimeRange $=t_1..t_2$」を指定し「VisibleBeforeStart $= FALSE$, VisibleAfterEnd $= FALSE$」と組み合わせて使えば，ある図形を t_1 から t_2 秒の間だけ表示させることができます.

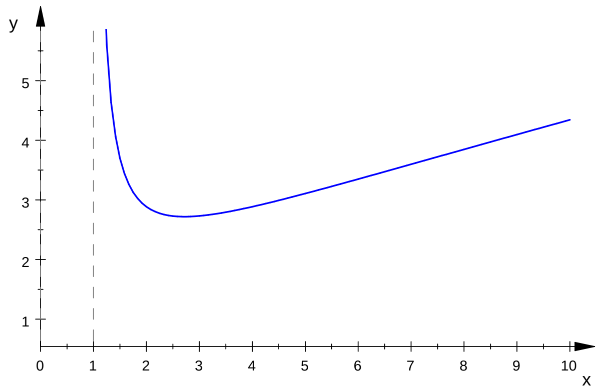
1.1　基本の例

Tutorial 1 (plotfunc2d)

$y = \frac{x}{\log x}(0 < x \leqq 10)$ のグラフの概略.

とりあえず, 普通に書いて見ます.

- plotfunc2d(x/ln(x), x = 0..10)



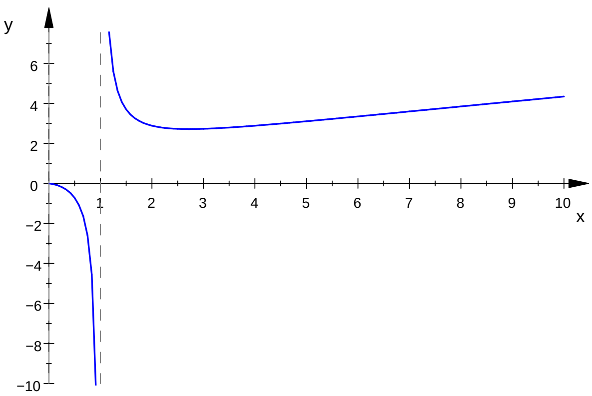
注1) これは ViewingBoxYMin,ViewingBoxYMax を使って　次のように書いても同じです.

$$\texttt{plotfunc2d(f(x), x = a ..b, ViewingBoxYMin = c, ViewingBoxYMax = d)}$$

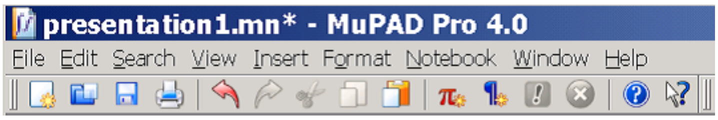
$0 < x < 1$ においては, グラフが表示されていません．$0 < x < 1$ においても $\frac{x}{\log x}$ は存在するはずなので，これは ViewingBox の問題と考えられます．そこで「ViewingBoxYMin=−10」を付け加えます．

- $\texttt{plotfunc2d}(\texttt{x}/\texttt{ln}(\texttt{x}), \texttt{ViewingBoxYMin} = -10, \texttt{x} = 0..10)$

すると, 次のようになります．

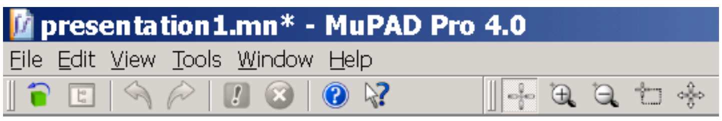


同じことを 「Object Browser」から interactive に変更することもできます．まず, **グラフの上を**ダブルクリックします．すると，メニューバーとツールバーが次のように変わるはずです．



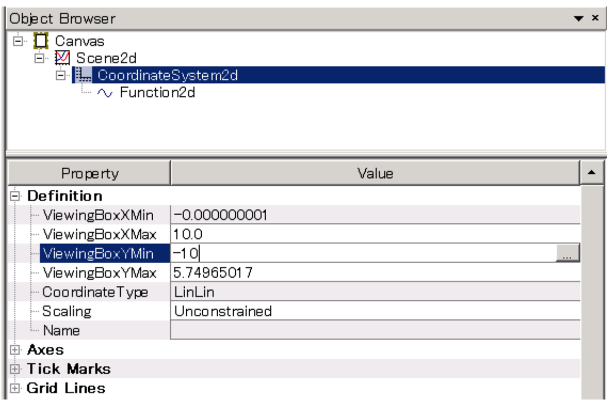
通常モード

⇓



画像モード

このメニューバーで「View⟶ObjectBrowser」をクリックするか，ツールバーの左から 2 番目のアイコンをクリックします． すると「Object Browser」が次のように開きます．



ViewingBox は「座標系」の問題なので， $\boxed{\text{CoordinateSystem2d}}$ をクリックします．次に， $\boxed{\text{Definition}}$ をクリックして,「ViewingBoxYMin=−10」と変えてみます．次にツールバーの $\boxed{!}$ をクリックするか, メニューバーから,「*View* ⟶ *Recalculate*」と辿り, `Recalculate`(再計算) させます．(ただ bug で,「`recalculate` が *active* にならない」時もあるようです.)

また グラフだけを独立に見ることもできます．通常モード になっていることを確認し，グラフの上で右クリックし $\boxed{\text{Open in Viewer}}$ を選びます．すると **Vcam** という viewer が独立に開きます．画像だけを保存するときにはこの viewer を使います．

Tutorial 2 (軌跡)

m を実数の定数とする．2 直線

$$l_1:\ mx - y = 0 \qquad l_2:\ x + my - m - 2 = 0$$

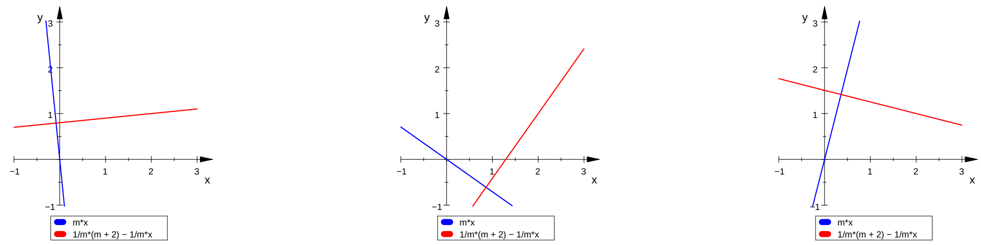
の交点の軌跡は，どのような図形になるか？

アニメーションで描いてみます． $m \fallingdotseq 0$ のとき，

$$x + my - m - 2 = 0 \iff y = -\frac{1}{m}x + \frac{m+2}{m}$$

よって次のようにしてみます.（軌跡は円になるので 「Scaling=Constrained」と指定します.）

$$\bullet \texttt{plotfunc2d(m}*\texttt{x},\ -1/\texttt{m}*\texttt{x} + (\texttt{m}+2)/\texttt{m},\ \texttt{x} = -1..3,\ \texttt{m} = -10..10$$
$$,\texttt{ViewingBoxYRange} = -1..3,\ \texttt{Scaling} = \texttt{Constrained})$$



Tutorial 3 (通過領域)

t が　全ての実数をとり変化するとき, 直線 $l: y = 2tx - t^2$ の通過領域を求めよ.

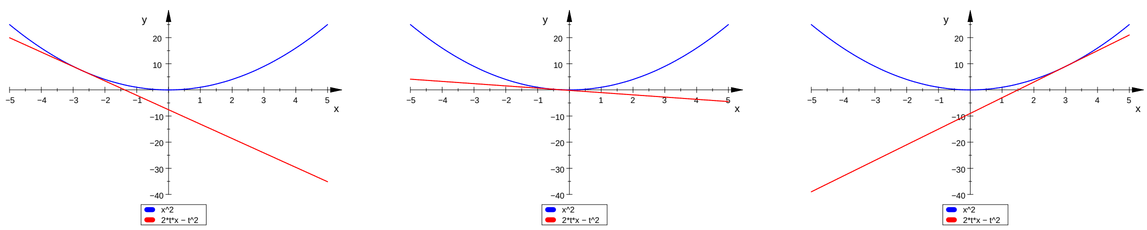
$$y = 2tx - t^2 \iff (t-x)^2 - x^2 + y = 0$$

ですから $C:\ y - x^2 = 0$ とおいて C と l の式を連立すると

$$(t-x)^2 = 0 \iff x = t\ (\text{重解})$$

よって C と l は $x = t$ の点で接する．(C は l の包絡線) MuPAD で C と l を一緒に表示します.

$$\bullet \texttt{plotfunc2d(x\^{}2},\ 2*\texttt{t}*\texttt{x} - \texttt{t\^{}2},\ \texttt{x} = -5..5,\ \texttt{t} = -3..3);$$



2 平面の幾何図形 (low-level primitives)

いくつかの幾何図形をすぐ描くことができます．これを”low-level primitive” と呼びます．2 次元 (平面) の low-level primitive の, 代表的なものは 以下のとおりです．ここで例えば「plot :: Circle」と言うのは 「plot library」の 「Circle」と言う名前のコマンドと言う意味です.

plot :: Arc2d	円弧 (円の一部)	$\mathtt{Arc2d(r, [C_x, C_y]}, \alpha..\beta)$
plot :: Arrow2d	矢印 (ベクトル)	$\mathtt{Arrow2d([x_1, y_1], [x_2, y_2])}$
plot :: Circle2d	円弧	$\mathtt{Circle2d(r, [C_x, C_y])}$
plot :: Line2d	線分	$\mathtt{Line2d([x_1, y_1], [x_2, y_2])}$
plot :: Point2d	点	$\mathtt{Point2d(x, y)}$
plot :: Text2d	文	$\mathtt{Text2d(}" \cdots ", \mathtt{[C_x, C_y])}$

これらは, 全て,

- plot(plot :: Circle2d(3, [3, 0])) · · · 中心 (3, 0) で, 半径 3 の円を描け

のように plot() コマンドと一緒に使います．しかし，次のように分けることもできます.

- mycircle := plot :: Circle2d(3, [3, 0]) · · · 中心(3,0)で, 半径 3 の円 (mycircle) を作る
- plot(mycircle) · · · mycircle を実際に描く

ここで「 mycircle := · · · 」というのは「mycircle を以下のように定義する」と言う意味です．注2)

3 様々なグラフ (high-level primitives)

plotfunc2d のように, 式を指定してグラフを描かせることもできます．こちらは，”high-level primitive” と呼びます．2 次元の high-level primitive の一例です.

plot :: Function2d	$y = f(x)$ のグラフ (陽関数表示)	$\mathtt{Function2d(f(x), x = x_1..x_2)}$
plot :: Implicit2d	$f(x, y) = 0$ のグラフ (陰関数表示)	$\mathtt{Implicit2d(f(x, y), x = x_1..x_2, y = y_1..y_2)}$
plot :: Curve2d	曲線のパラメータ表示	$\mathtt{Curve2d([x(t), y(t)], t = t_1..t_2)}$

3.1 $y = f(x)$ のグラフ (plot::Function2d)

Function2d は，$y = f(x)$ のグラフを描く時に使います．例えば，$y = x^2 (-2 \leqq x \leqq 2)$ のグラフは，次のようにします.

- f := plot :: Function2d(xˆ2, x = −2..2)
- plot(f)

これは，plotfunc2d を用いて,「plotfunc2d(xˆ2, x = −2..2)」としたものと全く同じです．Function2d は, 他の primitive （ plot :: Curve2d，plot :: Circle2d など ）と組み合わせる場合に用います.

注2) これは, 最近流行の 「オブジェクト指向」 の流れで，plot :: CIrcle(3, [0, 3]) で 「半径 3 で, 中心が (3, 0) の円」という**物 (object) を作り**，plot(object) で, その **物 (object) を描く**という感じです.

3.2　パラメータ表示された曲線:$x = f(t), y = g(t)$ のグラフ (`plot::Curve2d`)

$x = f(t), y = g(t)$ $(a \leqq t \leqq b)$ とパラメータ表示されたグラフの object は

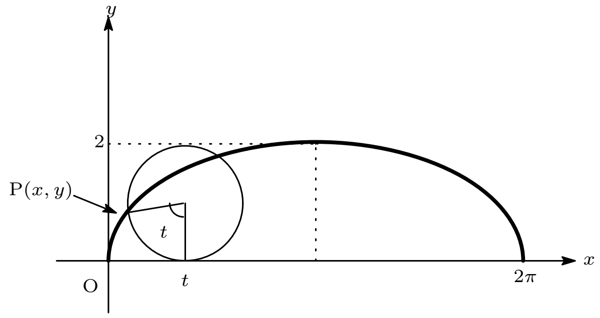
$$\texttt{plot::Curve2d([f(t), g(t)], t = a..b)}$$

Tutorial 3. サイクロイド

サイクロイド:

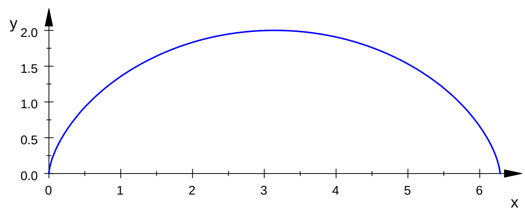
$$\begin{cases} x = t - \sin t \\ y = 1 - \cos t \end{cases}$$

を描け.

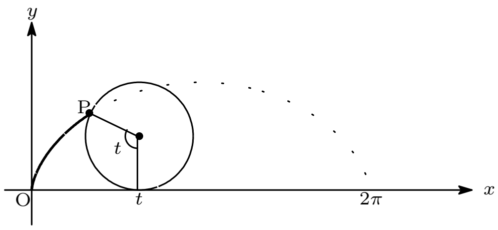


縦と横の縮尺を変えたくないので「Scaling=Constrained」を指定します.

- `mycycloid := plot::Curve2d([t - sin(t), 1 - cos(t)], t = 0..2*PI):`
- `plot(mycycloid, Scaling = Constrained)`



3.2.1　[サイクロイドのアニメーション]



半径 1 の円上の点 P が, はじめに原点にあったとし，その円が毎秒 1 の速度で，x 軸正方向に動いているとします．時刻 t において，円の中心も $(t,1)$ になるので，

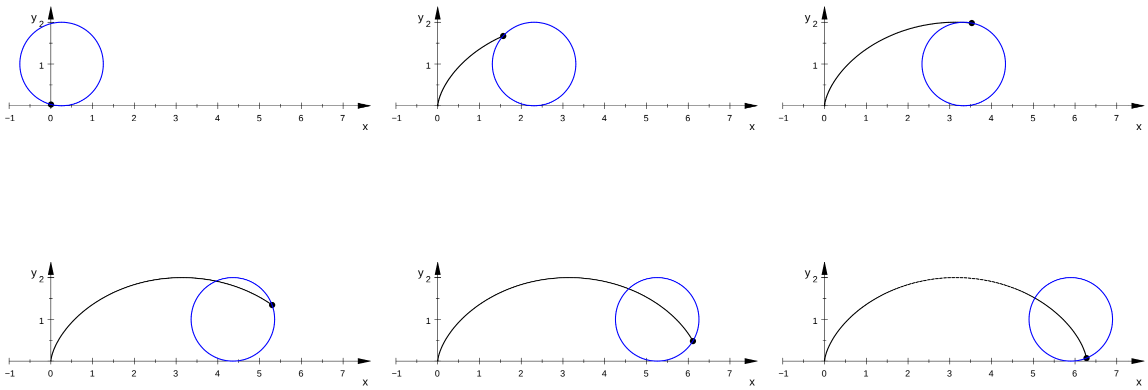
$$\begin{cases} \text{中心 } (t,1), \text{半径 1 の円 :} & (x-t)^2+(y-1)^2=1 \\ \text{サイクロイドの}\textbf{一部}\text{(O から P まで) :} & x=u-\sin u,\ y=1-\cos u\ (0 \leqq u \leqq t) \\ \text{点 P :} & x=t-\sin t,\ y=1-\cos t \end{cases}$$

となります．よってアニメーション $(0 \leqq t \leqq 2\pi)$ は 次のようになります.

- $\texttt{mycircle} := \texttt{plot} :: \texttt{Circle2d}(1,[\texttt{t},1],\ \texttt{t}=0..2*\texttt{PI},\ \texttt{Color}=\texttt{RGB} :: \texttt{Blue}) :$
  $\cdots$ 半径が 1, 中心が $(t,1)$ の青色の円
- $\texttt{mycycloid} := \texttt{plot} :: \texttt{Curve2d}([\texttt{u}-\texttt{sin}(\texttt{u}),1-\texttt{cos}(\texttt{u})],\ \texttt{u}=0..\texttt{t},\ \texttt{t}=0..2*\texttt{PI})$
- $\texttt{mypoint} := \texttt{plot} :: \texttt{Point2d}([\texttt{t}-\texttt{sin}(\texttt{t}),1-\texttt{cos}(\texttt{t})],\ \texttt{t}=0..2*\texttt{PI})$

これを, まとめて plot します.

- $\texttt{plot}(\texttt{mycircle},\texttt{mycycloid},\texttt{mypoint},\texttt{Color}=\texttt{RGB} :: \texttt{Black})$



注3)

[3]「mycircle」の定義で,「Color = RGB :: Blue」というのは,「青色」の mycircle と言う意味です．また最後の行の「Color = RGB :: Black」は，mycircle 以外の曲線を黒色にします．一般に plot( ) **の括弧内で指定した *Attribute* は，括弧内の全ての object に影響しますが，個別に指定された *Attribute* は変更しません．**従って

- $\texttt{mycircle} := \texttt{plot} :: \texttt{Circle2d}(1,[\texttt{t},1],\ \texttt{Color}=\texttt{RGB} :: \texttt{Blue}) :$
- $\texttt{mycycloid} := \texttt{plot} :: \texttt{Curve2d}([\texttt{u}-\texttt{sin}(\texttt{u}),1-\texttt{cos}(\texttt{u})],\ \texttt{u}=0..\texttt{t})$
- $\texttt{plot}(\texttt{mycircle},\texttt{mycycloid},\texttt{mypoint},\texttt{t}=0..2*\texttt{PI},\texttt{Color}=\texttt{RGB} :: \texttt{Black})$

の様に, 各オブジェクト内で個別に指定せずに，plot( ) 内に 一つだけ**アニメーションパラメータ**を指定することも可能です.

第2章　河合塾・東京 での視聴覚授業

我々のグループ (*Cabri* 研究会）[注4)] は, 東京都の主な校舎で それぞれ年に 5 回ほどのコンピュータを使った授業を行っています．(今年は 極限, 三角関数，ベクトル, 一次変換，平面図形 でした.) これは希望者のみの受講 (2 千円 / 1 回) で，多いときは 70 名ぐらい集まります．その場合，私の特に注意しているのは次の 2 点です.

(1) できる限り interactive (対話的) にすること.
(2) できる限り animation (動画) を入れること.

(1) とはいっても 残念ながら河合塾には コンピューター室は無いので,「授業前に簡単な問題を解いてもらい，その解答をコンピューターを使いながら考える」のが関の山です．しかし私や他の先生の webpage には，ソフトやプログラムを置いてあるので，インターネットを通じて，興味を持った生徒は自分の PC で遊ぶことができるように工夫してあります.
(2) 単にグラフを見せるのではなく，できるだけアニメーションにしています．一次変換でも 原像から最終図形の間を 適当な一次変換で補って, [注5)] アニメーションにできるようにしています.

ここでは,「極限」の授業から 基本公式:

$$\lim_{\theta\to 0}\frac{\sin\theta}{\theta}=1 \iff \sin\theta \quad \theta, \qquad \lim_{\theta\to 0}\frac{1-\cos\theta}{\theta^2}=\frac{1}{2} \iff \cos\theta \quad 1-\frac{1}{2}\theta^2.$$

$$\lim_{x\to 0}\frac{e^x-1}{x}=1 \iff e^x \quad 1+x, \qquad \lim_{x\to 0}\frac{\log(1+x)}{x}=1 \iff \log(1+x) \quad x.$$

の CG を，さらに 発展として $\sin\theta$ の 3 次近似式:

$$\lim_{\theta\to 0}\frac{\theta-\sin\theta}{\theta^3}=\frac{1}{6} \iff \sin\theta=\theta+\frac{1}{6}\theta^3+o(\theta^3)$$

のアニメーションをお見せします．また私のサイトから「積分」に関し 次の `plot` をお見せします.

- $x^2+z^2 \leqq 1,\ x^2+y^2 \leqq 1,\ y^2+z^2 \geqq 1$ の体積を求めよ. (東大 2005 年)
- $x^2+y^2 \leqq z^2,\ z^2 \leqq x,\ 0 \leqq z \leqq 1$ で表される領域の体積を求めよ. (東大 1994 年)
- A$(1,1,0)$,B$(-1,1,0)$,C$(-1,-1,0)$,D$(1,-1,0)$,P$(0,0,3)$ をとる．四角錐 PABCD の中に含まれ円柱 $x^2+y^2=1$ の外側にある領域の体積を求めよ. (東大 1998 年)
- A : $y^2+z^2=1$, B : $x^2-\sqrt{3}xz+z^2=\frac{1}{4}$ の交線の A 上での展開図を描け．(東大 1992 年)
- 円柱の一部: $x^2+y^2=1,\ z \leqq y+\frac{1}{2}$ の 平面 $y=k$ による切り口のアニメーション．(有名問題)

注4) *Cabri* というのは *Cinderella* と同じような幾何ソフトで，*Cabri* II*plus*(2D グラフィック) と *Cabri3D*(3D グラフィック) の二つがあります．*Cabri* II*plus* は *Cinderella* に比べて「痒い所に手の届くような」ソフトです．しかし *Cinderella* のように射影幾何がベースになっていないので，射影幾何や双曲幾何 (ポアンカレ円盤) の扱いでは一歩譲ります．*Cabri3D* は 空間の幾何図形 (球や平面など) をマウスで描くこともできる (私の知る限り唯一の) ソフトです．どちらも 1 万円強で買えます.

注5) 例えば「θ の回転」の場合は, その間に「$\frac{\theta}{n}$ の回転」 を $(n-1)$ 個補間する．行列 A が実数の固有値 $\alpha,\ \beta$ を持つときは, $P^{-1}A_0P=\left(\begin{smallmatrix}1&0\\0&1\end{smallmatrix}\right)$, $P^{-1}A_nP=\left(\begin{smallmatrix}\alpha&0\\0&\beta\end{smallmatrix}\right)$ になるような行列の列 $\{A_k\}$ $(k=0\cdots n)$ を作る.

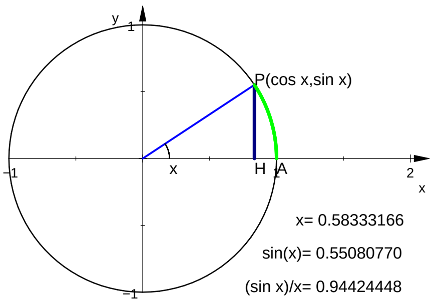
プレゼンテーション (極限) の説明

「$\lim_{\theta\to 0}\frac{\sin\theta}{\theta}=1$」のアニメーション

P は単位円上の点で P から x 軸に下ろした垂線の足を H とする．P が B$(\cos 1, \sin 1)$ の点からスタートし，点 A$(1,0)$ に向けて限りなく近づく時,

$$\frac{\text{線分 PH の長さ}}{\text{弧 PA の長さ}}=\frac{\overline{\mathrm{PH}}}{\angle\mathrm{AOP}}$$

の極限のアニメーション.



- `tmin` := 0.001 : ···**アニメーションパラメータ** t の最小値
- `Circle := plot :: Circle2d(1, [0, 0], LineStyle = Dashed, Color = RGB :: Black)` :
- `lineOP := plot :: Line2d([0, 0], [cos(t), sin(t)], t = 1..tmin)` : ···$(0,0)$ と $(\cos t, \sin t)$ を結ぶ線分
- `linePH := plot :: Line2d([cos(t), 0], [cos(t), sin(t)], Color = RGB :: NavyBlue, t = 1..tmin)` :
- `arcAP := plot :: Arc2d(1, [0, 0], 0..t, Color = RGB :: Green, t = 1..tmin)` :
  ···半径 1, 中心 $(0,0)$, $0 \leqq \theta \leqq t$ の円弧
- `arcX := plot :: Arc2d(0.2, [0, 0], 0..t, t = 1..tmin, Color = RGB :: Black)` :
- `pointX := plot :: Text2d("x", [0.2, 0])` : ···「x」という文字を $(0,2,0)$ におく
- `pointP := plot :: Text2d("P(cosx, sinx)", [cos(t), sin(t)], t = 1..tmin)` :
- `pointH := plot :: Text2d("H", [cos(t), 0], t = 1..tmin)` : ···**text の位置**のアニメーション
- `pointA := plot :: Text2d("A", [1, 0])` :
- `SinX := plot :: Text2d(t− > "sin(x) = ".stringlib :: formatf(sin(t), 8), [2, −0.75], t = 1..tmin)` :
- `X := plot :: Text2d(t− > "x = ".stringlib :: formatf(t, 8), [2, −0.5], t = 1..tmin)` :
- `SinXoverX := plot :: Text2d(t− > "(sinx)/x = ".stringlib :: formatf(sin(t)/t, 8), [2, −1], t = 1..tmin)` :
- `plot(Circle, lineOP, linePH, arcAP, arcX, pointX, pointP, pointH, pointA, SinX, X, SinXoverX`
  `, TicksNumber = Low, Frames = 100);`

「`plot :: Text2d(" ··· ", [x, y]`」で,「··· 」のテキストを (x,y) の位置に置きます．このとき位置のアニメーションは，今までと同様にできますが，**テキストの内容**のアニメーションは **関数定義コマンド**「− >」を使います．[注6] また,「小数 x を 有効数字 n 桁のテキストに直す」には 「`stringlib :: formatf(x, n)`」を使います．普通はこの 2 つは一緒に使います．また二つのテキストをくっつけるには「.」を使います．例えば 「`SinXoverX`」の定義において「`t− > "(sinx)/x = ".stringlib :: formatf(sin(t)/t, 8)`」の部分は 「$\frac{\sin t}{t}$ の値を 有効数字 8 桁のテキストに変えて，"`(sinx)/x =` " というテキストとくっつける」という意味です．[注7]

注6) 「− >」は，関数を定義するときに使います．例えば「$f(x)=x^2$」の定義は

• `f := x− > x^2`	$>> x \longrightarrow x^2$
• `f(3), f(4)`	>> 9, 16

注7) 数式 (*expression*) とテキスト (*text*) は，データ型が違います．*expression* を *text* に直すには `expr2text` を使うか，単に 「" "」で囲みます．従って「`t− > "(sinx)/x = ".expr2text(sin(t)/t)`」でも大丈夫ですが，桁数が一定でなくなります．

第3章　空間のグラフ

1　$y = f(x, y)$ のグラフ (plotfunc3d)

$z = f(x, y)$ のグラフ $(a \leqq x \leqq b,\ c \leqq y \leqq d)$	$\mathtt{plotfunc3d(\ f(x, y), x = a\ ..b,\ y = c..d\ )}$
$z = f(x, y)$ のアニメーション	$\mathtt{plotfunc2d(\ f(x, y, t), x = a\ ..b,\ y = c..d\ , t = t_1..t_2)}$

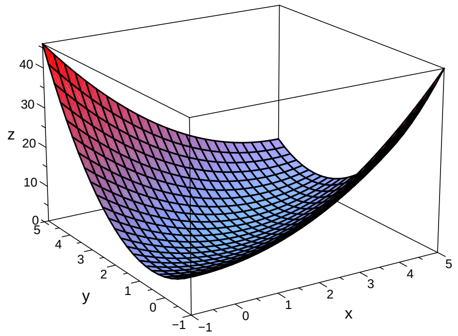
$z = f(x, y)$ のグラフを描くのは plotfunc3d を使うのが, 最も簡単です. z の範囲は,「ViewingBoxZRange $= \mathtt{z_1..z_2}$」で指定します.

例 (plotfunc3d)

$$f(x, y) = x^2 - 2xy + 2y^2 - 4y + 5\ (-1 \leqq x \leqq 5,\ -1 \leqq y \leqq 5) \qquad \cdots(*)$$

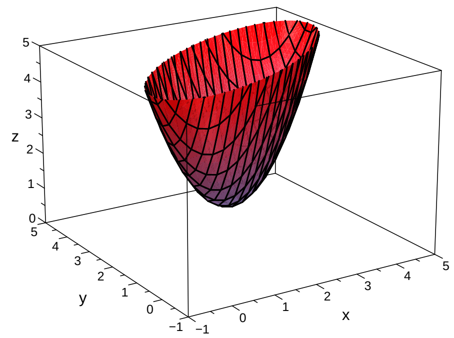
のグラフを描いて 最小値を求めよ.

- $\mathtt{plotfunc3d(x\hat{}2 - 2*x*y + 2*y\hat{}2 - 4*y + 5, x = -1..5, y = -1..5)}$



「$z = (x - y)^2 + (y - 2)^2 + 1$」ですから, $x = y = 2$ の前後で z 成分が最小となるはずですが, グラフからは良くわかりません. そこで, ViewingBoxZRange の範囲を変えて見ます.

- $\mathtt{plotfunc3d(x\hat{}2 - 2*x*y + 2*y\hat{}2 - 4*y + 5, x = -1..5, y = -1..5, ViewingBoxZRange = 0..5)}$



注8)

注8) 2 次元と同様, $\mathtt{plot(plot :: Function3d(f, x = x_1..x_2, y = y_1..y_2))}$ の省略が, $\mathtt{plotfunc3d(f, x = x_1..x_2, y = y_1..y_2)}$ です.

2 空間の幾何図形 (3d low-level primitive)

3 次元の low-level primitive の一例です.

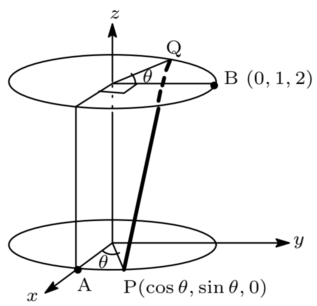
plot :: Arrow3d	矢印 (ベクトル)	$\texttt{Arrow3d}([x_1, y_1, z_1], [x_2, y_2, z_2])$
plot :: Box	直方体	$\texttt{Box}(x_1..x_2, y_1..y_2, z_1..z_2)$
plot :: Circle3d	円	$\texttt{Circle3d}(r, [C_x, C_y, C_z], [n_x, n_y, n_z])$
plot :: Cone	円錐	$\texttt{Cone}(r_{base}, [C_x, C_y, C_z], [D_x, D_y, D_z])$
plot :: Cylinder	円柱	$\texttt{Cylinder}(r, [C_x, C_y, C_z], [D_x, D_y, D_z])$
plot :: Line3d	線分	$\texttt{Line3d}([x_1, y_1, z_1], [x_2, y_2, z_2])$
plot :: Point3d	点	$\texttt{Point3d}(x, y, z)$
plot :: Sphere	球	$\texttt{Sphere}(r, [C_x, C_y, C_z])$
plot :: Text3d	文	$\texttt{Text3d}(" \cdots ", [C_x, C_y, C_z])$

2.1 進んだアニメーション

2.1.1 線分のアニメーション

Tutorial 1

2 点 A$(1,0,0)$, B$(0,1,2)$ を結ぶ線分 AB を, z 軸の周りに回転して得られる曲面はどのような曲面か?



点 A,B を z 軸の周りに θ 回転した点を, それぞれ P,Q とすると,

$$\mathrm{P}(\cos\theta, \sin\theta, 0),\ Q\left(\cos\left(\frac{\pi}{2}+\theta\right), \sin\left(\frac{\pi}{2}+\theta\right), 2\right)$$

よって次のようにすれば, 線分 PQ の回転をアニメーションにできます.

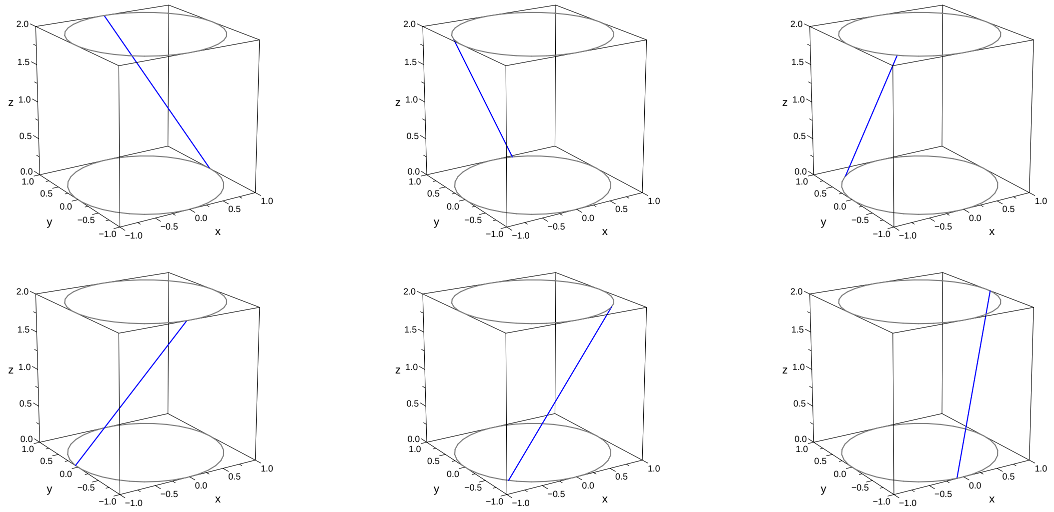
•segment := plot :: Line3d([cos(t), sin(t), 0], [cos(t + PI/2), sin(t + PI/2), 2], t = 0..2 ∗ PI) :

これに 2 つの円をつけて plot します.

•upperCircle := plot :: Circle3d(1, [0, 0, 2], Color = RGB :: Gray50) :

•lowerCircle := plot :: Circle3d(1, [0, 0, 0], Color = RGB :: Gray50) :

•plot(segment, upperCircle, lowerCircle)



2.1.2　図形の列 (列生成子「$」の利用)

どんな曲面なのか見るために，パラメータを「$\frac{\pi}{20}$」刻みで変えた線分 40 本を一度に描いてみます．即ち，

$\mathtt{plot} :: \mathtt{Line3d}([\mathtt{cos}(0), \mathtt{sin}(0), 0], [\mathtt{cos}(\mathtt{PI}/2), \mathtt{sin}(\mathtt{PI}/2), 2])$,

$\mathtt{plot} :: \mathtt{Line3d}([\mathtt{cos}(\mathtt{PI}/20), \mathtt{sin}(\mathtt{PI}/20), 0], [\mathtt{cos}(\mathtt{PI}/20 + \mathtt{PI}/2), \mathtt{sin}(\mathtt{PI}/20 + \mathtt{PI}/2), 2])$,

$\mathtt{plot} :: \mathtt{Line3d}([\mathtt{cos}(2 * \mathtt{PI}/20), \mathtt{sin}(2 * \mathtt{PI}/20), 0], [\mathtt{cos}(2 * \mathtt{PI}/20 + \mathtt{PI}/2), \mathtt{sin}(2 * \mathtt{PI}/20 + \mathtt{PI}/2), 2]) \cdots$

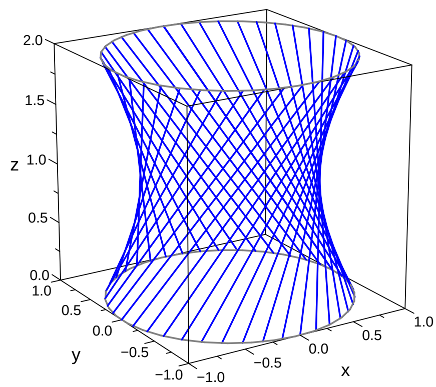
$\mathtt{plot} :: \mathtt{Line3d}([\mathtt{cos}(39 * \mathtt{PI}/20), \mathtt{sin}(39 * \mathtt{PI}/20), 0], [\mathtt{cos}(39 * \mathtt{PI}/20 + \mathtt{PI}/2), \mathtt{sin}(39 * \mathtt{PI}/20 + \mathtt{PI}/2), 2])$

上の式を実際に入力する必要はありません．**列生成子**「$」を使うと 次の一つの式で定義されます．[注9)]

- $\mathtt{segments} := \mathtt{plot} :: \mathtt{Line3d}([\mathtt{cos}(\mathtt{PI}/20 * \mathtt{k}), \mathtt{sin}(\mathtt{PI}/20 * \mathtt{k}), 0],$
  $[\mathtt{cos}(\mathtt{PI}/20 * \mathtt{k} + \mathtt{PI}/2), \mathtt{sin}(\mathtt{PI}/20 * \mathtt{k} + \mathtt{PI}/2), 2])\ \$\mathtt{k} = 0..39 :$　$\cdots$ ①

まとめて，plot します．

- $\mathtt{plot}(\mathtt{segments}, \mathtt{upperCircle}, \mathtt{lowerCircle})$



注9)「$」は，列を生成します．例えば

●$\mathtt{k\hat{}2}\ \$\mathtt{k} = 1..3$　　>> 1, 4, 9

●$\mathtt{sin}(\mathtt{k} * \mathtt{PI}/6)\ \$\mathtt{k} = 0..3$　　>> $0,\ \frac{1}{2},\ \frac{\sqrt{3}}{2},\ 1$

このとき「$」の前に　コンマ「,」を打たないようにしてください．

2.1.3 「TimeRange」の指定

ある図形をある時間 $(a \leqq t \leqq b)$ **の間だけ表示させるには「**`TimeRange` $=$ a..b**」と指定し，「**`VisibleBeforeStart` $= FALSE$**」 或いは「**`VisibleAfterEnd` $= FALSE$**」とセットで使います．**[注10)]

これを使い，描く線分の数が次第に増えていくような，アニメーションにします．

●`Segments := plot :: Line3d([cos(PI/20 * k), sin(PI/20 * k), 0],`
`[cos(PI/20 * k + PI/2), sin(PI/20 * k + PI/2), 2], TimeRange = 0.25 * k..10 ) $k = 0..39 :` ⋯②

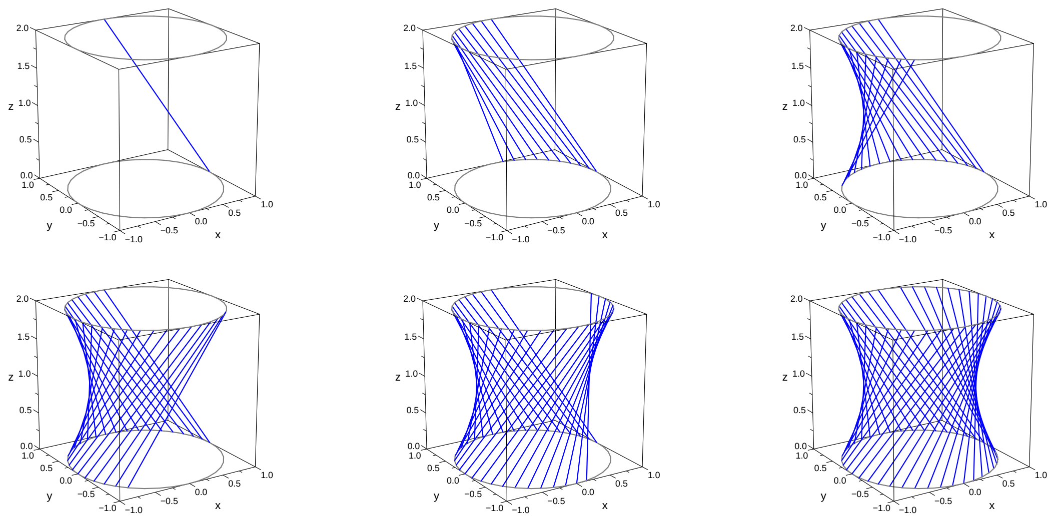
`TimeRange = 0.25 * k..10` だけを，① へ追加しました．これによって

```
plot :: Line3d([cos(0), sin(0), 0], [cos(PI/2), sin(PI/2), 2], TimeRange = 0..10),
plot :: Line3d([cos(PI/20), sin(PI/20), 0], [cos(PI/20 + PI/2), sin(PI/20 + PI/2), 2], TimeRange = 0.25..10) ···
plot :: Line3d([cos(39 * PI/20), sin(39 * PI/20), 0], [cos(39 * PI/20 + PI/2), sin(39 * PI/20 + PI/2), 2]
                                                                    , TimeRange = 9.75..10)
```

がまとめて生成されたことになります．これを，`VisibleBeforeBegin` $= FALSE$ のオプションをつけて，`plot` すれば， **その TimeRange に入るまでは見えないが，入った後は見え続ける** ので，線分の数が次第に増えて行きます．

●`plot(Segments, upperCircle, lowerCircle, VisibleBeforeBegin = FALSE)`

[注11)]



注10) Default では 「`TimeRange = 0..10`」と成っています．
注11) **アニメーションパラメータ**はありませんが， `TimeRange` を指定しているので，全部で 10 秒間のアニメーションになります．

3 様々な空間のグラフ (high-level primitives)

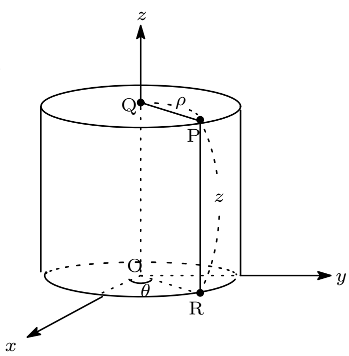
以下は，空間の high-level primitive の一例です.

`plot :: Function3d`	$z = f(x, y)$ のグラフ (陽関数表示) $\mathtt{Function3d(f(x,y), x = x_1..x_2, y = y_1..y_2)}$
`plot :: Implicit3d`	$f(x, y, z) = 0$ のグラフ (陰関数表示) $\mathtt{Implicit3d(f(x,y,z), x = x_1..x_2, y = y_1..y_2, z = z_1..z_2)}$
`plot :: Curve3d`	曲線のパラメータ表示 $\mathtt{Curve3d([x(t), y(t), z(t)], t = t_1..t_2)}$
`plot :: Surface`	曲面のパラメータ表示 $\mathtt{Surface([x(u,v), y(u,v), z(u,v)], u = u_1..u_2, v = v_1..v_2)}$

3.1 [円柱のプロット]

注12)
z 軸を中心軸, 底面の半径が ρ の円柱の側面上の点を P, 点 P(x, y, z) と z 軸との距離を ρ, P から z 軸におろした垂線の足を Q, xy 平面におろした垂線の足を R, OR と x 軸正方向との角を θ とすると図より

$$\begin{cases} x = \rho\cos\theta \\ y = \rho\sin\theta \\ z = z \end{cases} \qquad \cdots(*)$$

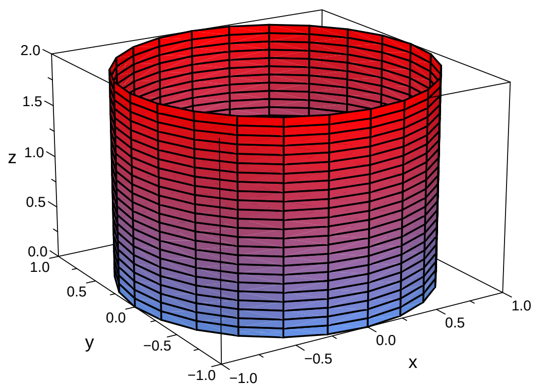


例えば,z 軸が中心軸，半径が 1，高さが 2 の円柱上の点は, 次のように表示されます.

$$(x, y, z) = (\cos\theta, \sin\theta, z)\ \ (0 \leqq \theta < 2\pi, 0 \leqq z \leqq 2)$$

よって，MuPAD では，次のようにします.

- $\mathtt{cylinderZ := plot :: Surface([cos(u), sin(u), z],\ u = 0..2*PI,\ z = 0..2):}$
- $\mathtt{plot(cylinderZ)}$



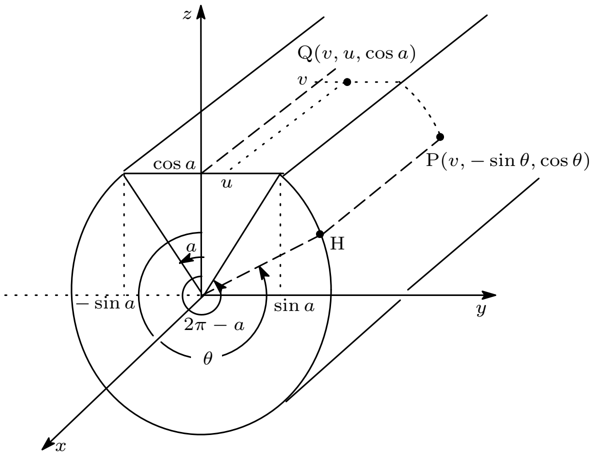
注12) 「low-level-primitive」の `plot :: Cylinder` を使っても描けます.

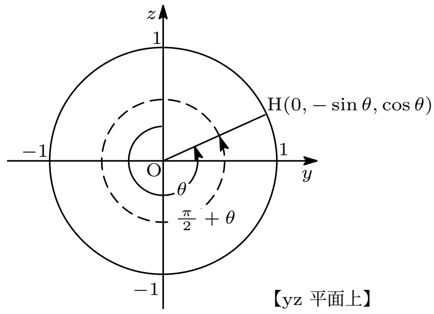
$\bullet$`cylinderZ := plot :: Cylinder(1, [0, 0, 0], [0, 0, 2])` ⋯半径が 1, 底面の中心が(0,0,0), 上面の中心が(0,0,2)の円柱

しかし `plot :: Cylinder` では，円柱を割った断面は描けません.

Tutorial 2

2 つの円柱 C_x : $y^2+z^2=1,\ -2 \leqq x \leqq 2$ と C_y : $x^2+z^2=1\ ,-2 \leqq y \leqq 2$ (C_x は x 軸に平行な高さが 4 の円柱, C_y は y 軸に平行な高さが 4 の円柱) の作図. さらに二つの円柱を 平面 $z=k$ で切ったときの断面をアニメーションで見る.





左上図のように z 軸正方向からの回転角 θ を取ったとき, 右上図から C_x 上の点 P は

$$\mathrm{P}\left(v, \cos\left(\frac{\pi}{2}+\theta\right), \sin\left(\frac{\pi}{2}+\theta\right)\right) = \mathrm{P}(v,\ -\sin\theta,\ \cos\theta)$$

とパラメータ表示されます. よって C_x 上で, 平面 $z=\cos a$ $(0 \leqq a \leqq \pi)$ より下側にある曲面は

$$C_x{}' :\ (x,y,z)=(v,\ -\sin\theta,\ \cos\theta),\quad -2 \leqq v \leqq 2,\ a \leqq \theta \leqq 2\pi - a \qquad \cdots ①$$

また平面 $z=\cos a$ 上で C_x の内部に含まれる長方形上の点 $\mathrm{Q}(x,y,z)$ は

$$\text{断面}:\ (x,y,z)=(v,u,\cos a),\quad -2 \leqq v \leqq 2,\ -\sin a \leqq u \leqq \sin a \qquad \cdots ②$$

と表示される. ゆえに左上図の曲面 (円柱とその断面) は, 次のような object になる.

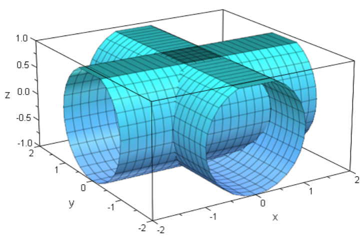
```
•cyX := plot :: Surface([v, −sin(t), cos(t)], v = −2..2, t = a..(2 ∗ PI − a));
•planeX := plot :: Surface([v, u, cos(a), v = −2..2, u = −sin(a)..sin(a));
```

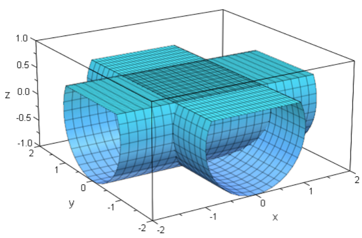
円柱 C_y とその断面は, 同様にして (x と y を入れ替えて),

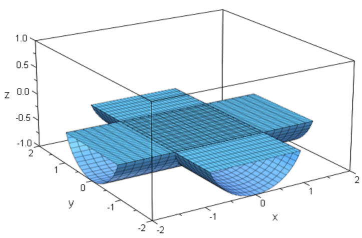
```
•cyY := plot :: Surface([−sin(t), v, cos(t)], v = −2..2, t = a..(2 ∗ PI − a));
•planeY := plot :: Surface([u, v, cos(a), v = −2..2, u = −sin(a)..sin(a));
```

パラメータを a $(0 \leqq a \leqq \pi)$ としてアニメーションさせます.

```
•plot(cyX, cyY, planeX, planeY, a = 0..PI, Scaling = Constrained, Color = RGB :: Aqua);
```







3.1.1 プレゼンテーションの解説

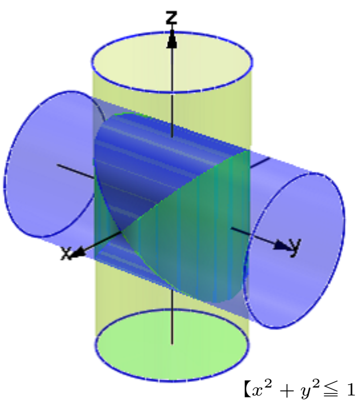
3 本の円柱

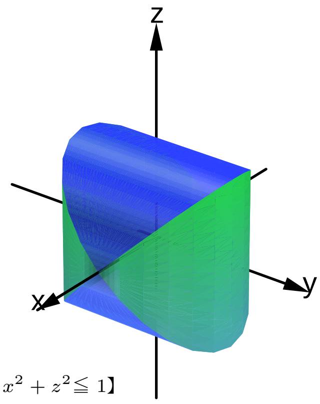
xyz 空間において

$$x^2+y^2 \leqq 1 \cdots ①,\quad x^2+z^2 \leqq 1 \cdots ②,\quad y^2+z^2 \geqq 1 \cdots ③$$

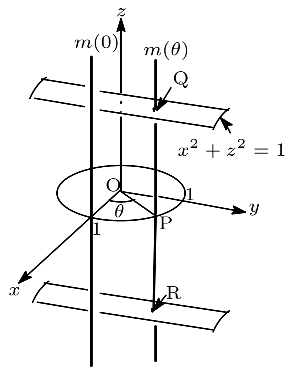
をみたす点全体からなる立体の体積を求めよ．(東大 · 理系 (改)　2005 年)

円柱の側面を $C_1: x^2+y^2=1,\ C_2: x^2+z^2=1,\ C_3: y^2+z^2=1$ とする．まず ① と② の共通部分の立体を D とするとき，D を単独に取り出した図を描く．(下中図)





【$x^2+y^2 \leqq 1$ かつ $x^2+z^2 \leqq 1$】



$C_1:\ x^2+y^2=1$ は 点 $\mathrm{P}(\cos\theta,\sin\theta,0)$ を通り z 軸に平行な直線 $m(\theta)$ の集まりである．

$$m(\theta): \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \cos\theta \\ \sin\theta \\ 0 \end{pmatrix} + t \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \cos\theta \\ \sin\theta \\ t \end{pmatrix} \quad (t \text{ は実数})$$

$m(\theta)$ と C_2 の方程式 $x^2+z^2=1$ を連立して

$$\cos^2\theta + t^2 = 1 \iff t = \pm\sqrt{1-\cos^2\theta} = \pm\sin\theta$$

よって，$m(\theta)$ と C_2 の交点の座標は

$$\mathrm{Q}(\cos\theta,\sin\theta,\sin\theta) \text{ と } \mathrm{R}(\cos\theta,\sin\theta,-\sin\theta)$$

よって D の表面のうち C_1 上にある曲面は

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \cos\theta \\ \sin\theta \\ v \end{pmatrix},\ (0 \leqq \theta \leqq 2\pi,\ -|\sin\theta| \leqq v \leqq |\sin\theta|) \qquad \cdots(*)$$

しかし MuPAD では，このような (θ が v の定義式に入っているような) 変域指定では駄目なので，拡大率 k をパラメーターに取り直し，次の様にしています． (他に良い方法があるかもしれません.)

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \cos\theta \\ \sin\theta \\ k\cdot\sin\theta \end{pmatrix},\ (0 \leqq \theta \leqq 2\pi,\ -1 \leqq k \leqq 1) \qquad \cdots(**)$$

また D の表面のうち C_2 上にある曲面は, 同様にして (y と z を入れ替えて)

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \cos\theta \\ k \cdot \sin\theta \\ \sin\theta \end{pmatrix}, \ (0 \leqq \theta \leqq 2\pi, \ -1 \leqq k \leqq 1)$$

よって D の表面は, 次のように作ることができます. この時 uline (u が一定の線) を描きたくないので「ULinesVisible $= FALSE$」を指定します. 同様に「VLinesVisible $= FALSE$」も指定します.

●cyZ := plot :: Surface([cos(t), sin(t), k * sin(t)], t = 0..2 * PI, k = −1..1,
ULinesVisible = FALSE, VLinesVisible = FALSE) :

●cyY := plot :: Surface([cos(t), k * sin(t), sin(t)], t = 0..2 * PI, k = −1..1,
ULinesVisible = FALSE, VLinesVisible = FALSE) :

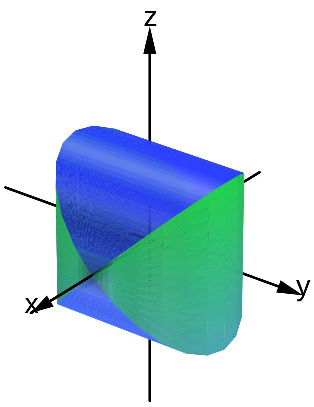
●plot(cyZ, cyY)

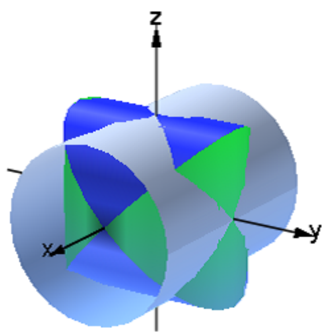
x, y, z 軸も作り 一緒に plot すると D が描けます.(左下図) 一方 C_3 のパラメーター表示は

● cyX := plot :: Surface([v, cos(t), sin(t)], t = 0..2 * PI, v = −2..2) :

● plot(cyZ, cyY, cyX)

今度は簡単です. x, y, z 軸も作り 一緒に plot すると 求める曲面が描けます.(右下図)





第4章　他の機能の簡単な紹介

1　微分, 積分

極限： $\lim_{x\to a} f(x)$	$\texttt{limit(f(x), x = a)}$
微分： $\frac{df}{dx}$	$\texttt{diff(f(x), x)}$
不定積分： $\int f(x)dx$	int(f(x),x)
定積分： $\int_a^b f(x)dx$	int(f(x),x=a .. b)

いろいろな関数の微分, 積分がそれぞれ，`diff`，`int` でできます．[注13)]

- `diff(cos(x), x)`　　>> $-\sin(x)$
- `int(sin(x), x)`　　>> $-\cos(x)$

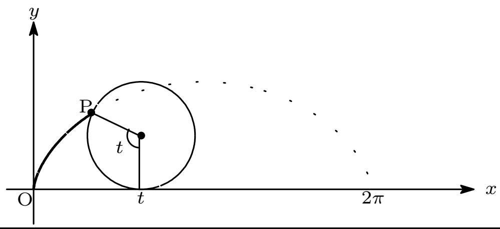
「$(\cos x)' = -\sin x$, $\int \sin x dx = -\cos x$」 を表しています．定積分もできます．

- `int(sin(x), x = 0..PI)`　　>> 2
- `int(sin(x)^4, x = 0..PI)`　　>> $\dfrac{3\cdot\pi}{8}$

π は，`PI` と入力します．「^」は,「累乗」を表します．したがって,「$\int_0^\pi \sin x dx = [-\cos x]_0^\pi = 2$」「$\int_0^\pi \sin^4 x dx = \frac{3\pi}{8}$」を表しています．面積や体積も求まります．

例 1

サイクロイド: $x = t - \sin(t)$, $y = 1 - \cos(t)$ $(0 \leqq t \leqq 2\pi)$ と x 軸で囲まれた領域 D の面積 S , および，D を x 軸の周りに回転してできる立体の体積 V を求めよ．



$$S = \int_0^{2\pi} y\ dx = \int_0^{2\pi} y\ \frac{dx}{dt}dt, \qquad V = \int_0^{2\pi} \pi y^2\ dx = \int_0^{2\pi} \pi y^2\ \frac{dx}{dt}dt$$

ですから

[面積] • `int((1 - cos(t)) * diff(t - sin(t), t), t = 0..2 * PI)`　　>> 3π

[体積] • `int(PI * (1 - cos(t))^2 * diff(t - sin(t), t), t = 0..2 * PI)`　　>> $5\pi^2$

注13) `diff` は $differenciate$(微分する) の，`int` は $integral$ の略です．

2 方程式を解く

方程式 $f = 0$ を x に関し解く	$\mathtt{solve(f = 0, x)}$ または $\mathtt{solve(f, x)}$
不等式 $f > 0$ を x に関し解く	$\mathtt{solve(f > 0, x)}$
連立方程式 $f = 0, g = 0, \cdots$ を $x, y, \cdots$ に関し解く	$\mathtt{solve(\ \{f = 0, g = 0, \cdots\}, [x, y, \cdots]\ )}$

[注14)] 様々な方程式 (整式の不等式も) を解くことができます．$f(x) = 0$ の解を求めるには,「$\mathtt{solve(f = 0, x)}$ または $\mathtt{solve(f, x)}$」としますが，一変数の方程式では $\mathtt{solve(f)}$ のように 変数名は省略できます.

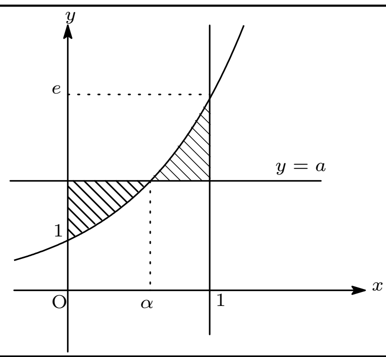
• $\mathtt{solve(x\hat{\ }2 - 3 * x + 2, x)}$ >> $\{1, 2\}$

$\mathtt{x\hat{\ }2}$ は x^2 を，$\mathtt{3 * x}$ は $3x$ を表します．すなわち,「$x^2 - 3x - 2 = 0$ の解が $\{1, 2\}$」 ということです.
[注15)] 連立方程式も解けます.「$x^2 - xy = 6,\ y^2 - yx = -2$」を解いてみます.

• $\mathtt{solve(\{x^2 - x * y = 6,\ y^2 - y * x = -2\}, [x, y])}$ >> $\{[x = -3, y = -1], [x = 3, y = 1]\}$

例 2

曲線 C : $y = e^x$, 直線 $l : y = a\ (1 < a < e)$ と，直線 $x = 1$, y 軸 によって囲まれる 2 つの領域の面積の和を S とする．このとき，S の式と，S を最小にする a の値を求めよ.



はじめに，a の変域を assume によって制限します．(パラメータを含む絶対値付きの関数の積分では, assume を使ってパラメータの範囲を制限します．)

• $\mathtt{assume(1 < a < E)}$ >> $(1, e)$

次に，絶対値をつけて積分します. (「e^x」は「exp(x)」とします.)

• $\mathtt{S := int(abs(exp(x) - a), x = 0..1)}$ >> $e - 3 \cdot a + 2 \cdot a \cdot \mathtt{ln}(a) + 1$

S を a で微分して，S が極小値を取る a の値を求めます.

• $\mathtt{diff(S, a)}$ >> $2 \cdot \mathtt{ln}(a) - 1$

• $\mathtt{solve(\%, a)}$ >> $\{\sqrt{e}\}$

「$a = \sqrt{e}$ のとき S は最小」です．ここで「$\mathtt{solve(\%, a)}$」の「%」**というのは，直前の出力を表します．**[注16)]

[注14)] 連立方程式, 不等式は { } を使っても，[ ] を使っても大丈夫です．例えば，$\mathtt{solve(\ \{f = 0, g = 0, \cdots\}, [x, y, \cdots]\ )}$ は，$\mathtt{solve(\ \{f = 0, g = 0, \cdots\}, \{x, y, \cdots\}\ )}$ や，$\mathtt{solve(\ [f = 0, g = 0, \cdots], [x, y, \cdots]\ )}$ でも大丈夫です.

[注15)] 3 次,4 次方程式も解けます．しかし default では 2 次までの解の公式しか使用しませんから 3 次,4 次の解の公式を使用させる場合は「$\mathtt{MaxDegree = 3, Maxdegree = 4}$」のように指定します.「$\mathtt{MaxDegree = 3}$」を指定して「$x^3 + 6x - 2 = 0$」を解いてみます.

• $\mathtt{solve(x\hat{\ }3 + 6 * x - 2, x, MaxDegree = 3)}$

$$>> \left\{\sqrt[3]{4} - \frac{4^{\frac{2}{3}}}{2},\quad \frac{4^{\frac{2}{3}}}{4} - \frac{\sqrt[3]{4}}{2} - \frac{i}{2} \cdot \left(\sqrt[3]{4} + \frac{4^{\frac{2}{3}}}{2}\right) \cdot \sqrt{3},\quad \frac{i}{2} \cdot \left(\sqrt[3]{4} + \frac{4^{\frac{2}{3}}}{2}\right) \cdot \sqrt{3} - \frac{\sqrt[3]{4}}{2} + \frac{4^{\frac{2}{3}}}{4}\right\}$$

[注16)] さらに前の出力結果が欲しいときは「$\mathtt{\%2, \%3, \cdots}$」 とします.

3 式の変形 (因数分解, 展開, 簡略化)

f を展開	`expand(f)`
f を因数分解	`factor(f)`
f の単純化	`simplify(f)` または `Simplify(f)`

式の因数分解は `factor` でできます.

- eq1 := (a − b)^3 + (b − c)^3 + (c − a)^3 　　 $>> (a-b)^3 - (a-c)^3 + (b-c)^3$

長くなるので, $(\mathrm{a}-\mathrm{b})^3 + (\mathrm{b}-\mathrm{c})^3 + (\mathrm{c}-\mathrm{a})^3$ を, eq1 と名前をつけました.

- `factor(eq1)` 　　 $>> -3\cdot(b-c)\cdot(a-c)\cdot(a-b)$

逆に式の展開は `expand` でします.

- `expand`$((\mathrm{x}-\mathrm{y})^3)$ 　　 $>> x^3 - x^2\cdot y\cdot 3 + 3\cdot x\cdot y^2 - y^3$

式の変形 (展開, 因数分解, 通分, 無理数の有理化など）にはそれぞれ専用コマンドがありますが，とりあえず `simplify`, または `Simplify` で簡単にすることができます．(しかし，無理数の有理化は `radsimp` を使わないとうまく行かないことが多いです.)

- eq2 := (x^2)/(x^2 − 1) − (4 ∗ x − 3)/(x^2 − 1) 　　 $>> \dfrac{x^2}{x^2-1} - \dfrac{4\cdot x-3}{x^2-1}$

これを `simplify` で簡単にしてみます.

- `simplify`(%) 　　 $>> \dfrac{x-3}{x+1}$

「%」**というのは，直前の出力を表します．**(18 ページ) ここでは「`symplify(eq2)`」と同じです．次は「$\log_5 7\cdot\log_7 9\cdot(\log_3 5+\log_9 5)$」を簡単にします. (「$\log_a x$」 は「`log`(a, x)」と入力します. ）

- eq3 := `log`(5, 7) ∗ `log`(7, 9) ∗ (`log`(3, 5) + `log`(9, 5)) 　　 $>> (\log_3(5)+\log_9(5))\cdot\log_7(9)\cdot\log_5(7)$

`simplify` で簡単にしてみます.

- `simplify`(eq3) 　　 $>> (\log_3(5)+\log_9(5))\cdot\log_7(9)\cdot\log_5(7)$

ぜんぜん簡単になりません．しかし，`Simplify` なら簡単にできます.

- `Simplify`(eq3) 　　 $>> 3$

この `Simplify` は，MuPAD 3.0 から導入されました．`simplify` より時間はかかりますが, より正確です.

4 行列, ベクトル

4.1 行列, ベクトルの定義

2×2 行列 $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$	`matrix([[a,b],[c,d]])`
列ベクトル $\begin{pmatrix} a \\ b \end{pmatrix}$, $\begin{pmatrix} a \\ b \\ c \end{pmatrix}$	`matrix([[a],[b]])`, `matrix([[a],[b],[c]])`
行ベクトル (a,b), (a,b,c)	`matrix([a,b])`, `matrix([a,b,c])`

行列・ベクトルは, `matrix` を使い, `matrix`**([[第一行],[第 2 行],$\cdots$,[第 n 行]])** のように, 定義できます.

- `matrix([[2,5,7]])` >> $(2,5,7)$
- `V := matrix([[3],[4],[5]])` >> $\begin{pmatrix} 3 \\ 4 \\ 5 \end{pmatrix}$

`matrix([[2,5,7]])` は要素がひとつ ([2,5,7] だけ) なので, 行ベクトルに, 一方, `matrix([[3],[4],[5]])` は要素が 3 つなので, 列ベクトルになります. 外側の [ ] は省略できません. 注17) 行列も同様に定義できます.

- `A := matrix([[1,2],[3,4]])` >> $\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$ $\cdots$ A の定義
- `B := matrix([[1,2],[-1,-2]])` >> $\begin{pmatrix} 1 & 2 \\ -1 & -2 \end{pmatrix}$ $\cdots$ B の定義

4.2 行列, ベクトルの四則計算

行列の加減乗除は実数と同じ様に計算できます.

- `2*A+3*B` >> $\begin{pmatrix} 5 & 10 \\ 3 & 2 \end{pmatrix}$ $\cdots$ 2A+3B
- `A*B` >> $\begin{pmatrix} -1 & -2 \\ -1 & -2 \end{pmatrix}$ $\cdots$ AB
- `B*A` >> $\begin{pmatrix} 7 & 10 \\ -7 & -10 \end{pmatrix}$ $\cdots$ BA
- `A^(-1)` >> $\begin{pmatrix} -2 & 1 \\ \frac{3}{2} & -\frac{1}{2} \end{pmatrix}$ $\cdots A^{-1}$
- `1/A` >> $\begin{pmatrix} -2 & 1 \\ \frac{3}{2} & -\frac{1}{2} \end{pmatrix}$ $\cdots A^{-1}$
- `A^2` >> $\begin{pmatrix} 7 & 10 \\ 15 & 22 \end{pmatrix}$ $\cdots A^2$

注17) 内側の [ ] を省略すると, 列ベクトルになります.

- `V := matrix([3,4,5])` >> $\begin{pmatrix} 3 \\ 4 \\ 5 \end{pmatrix}$

4.3 linalg (線形代数のライブラリ)

行列の対角化 (一般には Jordan 標準形への変形) も 簡単にできます．高度な行列やベクトルの演算は, linalg (library for linear algebra) を使います．ごく一例を挙げます.

- $\texttt{linalg} :: \texttt{scalarProduct}(\texttt{u}, \texttt{v})$ $\cdots \overrightarrow{u}$ と $\overrightarrow{v}$ の内積
- $\texttt{linalg} :: \texttt{angle}(\texttt{u}, \texttt{v})$ $\cdots \overrightarrow{u}$ と $\overrightarrow{v}$ のなす角
- $\texttt{linalg} :: \texttt{rank}(\texttt{A})$ $\cdots A$ の階数 $(rank)$
- $\texttt{linalg} :: \texttt{charpoly}(\texttt{A}, \texttt{x})$ $\cdots A$ の (変数を x とする) 固有方程式
- $\texttt{linalg} :: \texttt{eigenvalues}(\texttt{A})$ $\cdots A$ の固有値
- $\texttt{linalg} :: \texttt{eigenvectors}(\texttt{A})$ $\cdots A$ の固有ベクトル
- $\texttt{linalg} :: \texttt{jordanForm}(\texttt{A})$ $\cdots A$ の $jordan$ 標準形 (対角化)

2×2 行列でやってみます．(一般の行列でも同様です.)

- $\texttt{A} := \texttt{matrix}([[2, 1], [3, 4]])$ $>> \begin{pmatrix} 2 & 1 \\ 3 & 4 \end{pmatrix}$
- $\texttt{linalg} :: \texttt{rank}(\texttt{A})$ $>> 2$
- $\texttt{linalg} :: \texttt{charpoly}(\texttt{A}, \texttt{x})$ $>> x^2 - 6 \cdot x + 5$
- $\texttt{linalg} :: \texttt{eigenvalues}(\texttt{A})$ $>> \{1, 5\}$
- $\texttt{linalg} :: \texttt{eigenvectors}(\texttt{A})$ $>> \left[\left[1, 1, \begin{pmatrix} -1 \\ 1 \end{pmatrix}\right], \left[5, 1, \begin{pmatrix} \frac{1}{3} \\ 1 \end{pmatrix}\right]\right]$
- $\texttt{linalg} :: \texttt{jordanForm}(\texttt{A})$ $>> \begin{pmatrix} 1 & 0 \\ 0 & 5 \end{pmatrix}$

5 数列の和

$\sum_{k=1}^{n} a_k$	$\mathsf{sum}(\mathsf{a}(\mathsf{k}), \mathsf{k} = 1..\mathsf{n})$
$F(k+1) - F(k) = a(k)$ となる $F(k)$	$\mathsf{sum}(\mathsf{a}(\mathsf{k}), \mathsf{k})$

$\displaystyle\sum_{k=1}^{n} f(k)$ は「$\mathsf{sum}(\mathsf{f}(\mathsf{k}), \mathsf{k} = 1..\mathsf{n})$」で，また $f(k+1) - f(k) = F(k)$ をみたす $F(k)$ ($f(k)$ の原始関数) は,「$\mathsf{sum}(\mathsf{f}(\mathsf{k}), \mathsf{k})$」で求まります．[注18] 例えば

- $\mathsf{sum}(\mathsf{k}^2, \mathsf{k} = 1..\mathsf{n})$ $>> \dfrac{n \cdot (2 \cdot n + 1) \cdot (n + 1)}{6}$ $\cdots \sum_{k=1}^{n} k^2 = \frac{n(n+1)(2n+1)}{6}$
- $\mathsf{sum}(\mathsf{k}^2)$ $>> \dfrac{k}{6} - \dfrac{k^2}{2} + \dfrac{k^3}{3}$

残念ながら 漸化式は簡単な式しか解けません．(微分方程式は かなり難しい式も解けるのですが $\cdots$)

注18) 「$\texttt{sum(f(k), k = a..b)}$ と $\texttt{sum(f(k), k)}$」の関係は,「$\texttt{int(f(x), x = a..b)}$ と $\texttt{int(f(x), x)}$」の関係と同様です.

6 References

6.1 コマンド入力のヘルプ

オンサイトヘルプ	F2
入力補助	Ctrl + Space

【注】これらは，コマンドの入力中に使います．

6.2 コマンド入力時の 基本操作

計算結果を 表示させない	文末を「:」で終る
計算結果を 表示させる	「;」または「 」(空白) で終る
計算させずに 改行	Shift + Enter
入力式のコメント化	「//⋯ 」または「/*⋯*/」
直前の出力結果の 参照	「%」または「history()」
n 回前の出力結果の 参照	「% n」または「history(n)」

6.3 四則演算

和： $a+b$	a + b
差： $a-b$	a − b
積： $a \times b$	a * b
商： $\frac{a}{b}$	a/b
累乗： a^n	a^n

6.4 関数・定数表

$\sin x$	sin(x)
$\cos x$	cos(x)
$\tan x$	tan(x)
a^x	_power(a, x) または a^x
e^x	exp(x) または E^x
$\log_a x$	log(a, x)
$\log x$(自然対数)	ln(x)
絶対値 $\|x\|$	abs(x)
x の浮動小数表示	float(x)
π(円周率)	PI
e(自然定数の底)	E
∞(無限大)	infinity

【注】大文字と小文字は区別します．

6.5 リンク

私のサイト (MuPAD)	http://ogose.nendo.net/mupad/
私のサイト (今回の資料)	http://ogose.nendo.net/temp/koushiken.zip
ライトストーン (日本語版 発売元)	http://www.lightstone.co.jp/products/mupad/