

# Guía rápida de wxMaxima

Marduk Bolaños Puchet

octubre de 2012

Maxima es un Sistema de Algebra Computacional, que proporciona herramientas para la manipulación simbólica de expresiones matemáticas.

wxMaxima proporciona una interfaz gráfica en forma de bitácora para el intérprete **maxima**.

La bitácora está estructurada en **apartados** de distintos tipos:

- Comandos (Input cell)
- Texto
- Título
- Sección
- Subsección
- Imagen

Los apartados de comandos son evaluados por el intérprete **maxima** al presionar **Shift+Return**.

Para omitir el resultado de un comando, se debe agregar **\$** al final de la línea.

## Operadores

<code>+, -, *, /, ^</code> o <code>**</code>	operaciones aritméticas básicas
<code>!</code>	factorial
<code>:</code>	asignación de variables
<code>:=</code>	definición de funciones
<code>=</code>	igualdad
<code>#</code>	diferente
<code>&lt;, &gt;, ≤, ≥</code>	desigualdades
<code>.</code>	multiplicación de matrices
<code>and, or, not</code>	operadores lógicos

## Funciones

<code>abs</code>	valor absoluto
<code>fix</code>	parte entera inferior
<code>sqrt</code>	raíz cuadrada
<code>mod</code>	residuo de la división
<code>exp</code>	exponencial
<code>gcd</code>	máximo común divisor
<code>num</code>	numerador de una fracción
<code>denom</code>	denominador de una fracción
<code>realpart</code>	parte real de un número complejo
<code>imagpart</code>	parte imaginaria de un número complejo
<code>product(f,k,min,max)</code>	$\prod_{k=min}^{max} f(k)$
<code>sum(f,k,min,max)</code>	$\sum_{k=min}^{max} f(k)$
<code>divide</code>	división de polinomios (regresa el cociente y el residuo)
<code>log</code>	logaritmo natural
<code>sin, cos, asin, acos</code>	sin, cos, arcsin, arc cos
<code>allroot</code>	busca todas las raíces de un polinomio real
<code>interpolate(f,x,a,b)</code>	encuentra los ceros de $f(x)$ en $[a, b]$
<code>binomial</code>	coeficiente binomial
<code>cfdisrep([c1,c2,...])</code>	muestra una fracción continua

## Constantes

<code>%pi</code>	$\pi$
<code>%e</code>	$\exp(1)$
<code>%i</code>	$\sqrt{-1}$
<code>true, false</code>	constantes booleanas

## Listas

<code>[expr1, expr2,...]</code>	crea una lista
<code>&lt;lista&gt;[i]</code>	i-ésimo elemento de la <i>lista</i>
<code>append(list1, list2)</code>	concatena dos listas
<code>makelist(f(i),i,min,max)</code>	crea una lista

## Cálculo diferencial e integral

<code>depends(var1, var2)</code>	<i>var1</i> es función de <i>var2</i>
<code>x inf, minf</code>	$\infty, -\infty$
<code>limit(f,x,k)</code>	$\lim_{x \rightarrow k} f(x)$
<code>limit(f,x,k,PLUS(MINUS))</code>	$\lim_{x \nearrow (\searrow) k} f(x)$
<code>diff(f(x),x,n)</code>	n-ésima derivada de $f(x)$
<code>integrate(f(x),x)</code>	antiderivada de $f(x)$
<code>integrate(f(x),x,min,max)</code>	$\int_{x=min}^{x=max} f(x) dx$
<code>powerseries(expr,var,punto)</code>	desarrollo en serie de potencias
<code>taylor(fun, var, punto, pot)</code>	desarrollo de Taylor hasta $n = pot$

## Algebra Lineal

<code>solve([eq1,...],[var1,...])</code>	resuelve un sistema de ecuaciones
<code>matrix([m11,m12,...],[m21,m22],...)</code>	crea una matriz
<code>adjoint</code>	encuentra la adjunta de una matriz
<code>col(mat,n), row(mat,m)</code>	obtiene la columna $n$ o el renglón $m$
<code>&lt;matriz&gt;[i,j]</code>	elemento $m_{ij}$ de la <i>matriz</i>
<code>charpoly(mat,var)</code>	calcula el polinomio característico
<code>determinant</code>	calcula el determinante
<code>eigenvalues</code>	calcula los valores propios
<code>eigenvectors</code>	calcula los vectores propios
<code>ident(n)</code>	matriz identidad de $n \times n$
<code>invert</code>	encuentra la matriz inversa (si existe)
<code>matrixmap(fun,mat)</code>	aplica la función a cada elemento de la matriz
<code>rank</code>	calcula el rango de una matriz
<code>transpose</code>	transpone una matriz
<code>echelon</code>	encuentra la matriz escalonada
<code>genmatrix(h,N,N)</code>	genera una matriz con elementos dados por la función $h$
<code>express([1,0,0]~[0,1,0])</code>	producto cruz de dos vectores

## Ecuaciones Diferenciales Ordinarias

<code>ode2(eq,y,x)</code>	resuelve una EDO
---------------------------	------------------

## Manipulación de expresiones

<b>factor</b>	factorizar la expresión
<b>expand</b>	lo contrario de factorizar
<b>xthru</b>	factorizar el denominador común
<b>subst(y=x,expr)</b>	substituir $y$ por $x$ en $expr$
<b>'expr</b>	escribe $expr$ en notación matemática, pero no la evalúa
<b>expr, numer</b>	evalúa numéricamente la expresión

## Gráficos con plot

<b>wxplot2d([f(x),g(x)],[x,min,max])</b>	gráfico de $f(x)$ y $g(x)$
<b>wxplot2d([parametric,f(t),g(t)],[t,min,max])</b>	curva paramétrica
<b>preambulo: [gnuplot_preamble, "set polar"]</b>	
<b>wxplot2d(r(ph),[ph,0,2*pi],preambulo)</b>	curva en coordenadas polares
<b>wxplot3d([f(x,y),[x,min,max],[y,min,max]])</b>	gráfico de $f(x,y)$

### Guardar gráficos en un archivo

```
archivo: [gnuplot_out_file,"plot.pdf"]$
terminal: [gnuplot_term,pdf]$
plot2d(f(x),[x,min,max],terminal,archivo)$
```

## Gráficos con draw

```
y = f(x)

load(draw);
draw2d(
color=blue,
terminal = pdf,
file_name = "plot",
explicit(f(x),x,min,max))
```

### Curva paramétrica

```
draw2d(
color=red,
color=blue, nticks=60,
parametric(x(t),y(t),t,min,max));
```

$z = f(x,y)$

```
draw3d(surface_hide=true,
explicit(f(x,y),x,min,max,y,min,max));
```

## Animaciones con draw y animate

```
load(draw)$
:lisp (defun $add_zeros (num) (format nil "~10,'0d" num))$
system("mkdir anim")$
```

```
for t: 0 thru t_final do
draw2d(
terminal = png,
file_name = concat("anim/cuadro",add_zeros(t)),
```

```
point_size = 1,
point_type = 5,
points([x1,x2],[y1,y2]),
xrange=[xmin,xmax]
);
```

```
system("animate -delay 5 anim/*.png");
```

## Punto flotante

<b>float, bfloat</b>	convierte a <i>float</i> o <i>bfloat</i>
<b>fpprec:val</b>	cuántas cifras decimales
<b>fpprintprec:val</b>	cuántos dígitos imprime
<b>?round</b>	redondea a un número entero
<b>?truncate</b>	trunca a un número entero

## Entrada y salida

<b>load('ruta/completa/archivo.mac')</b>	cargar un archivo con comandos de Maxima
<b>tex(expr,'archivo.tex')</b>	escribe $expr$ en notación de $\text{\LaTeX}$ en un archivo
<b>system('comando')</b>	ejecuta un <i>comando</i> del sistema operativo

## Programación

```
for var:<min>thru <max>step <num>do <block>    ciclo for
```

## Paquetes

<b>fourie</b>	series y transformada de Fourier
<b>linearalgebra</b>	hessiana, jacobiana
<b>eigen</b>	autovectores unitarios
<b>nchrpl</b>	traza
<b>sound</b>	generar, reproducir sonido
<b>dynamics, fractals</b>	fractales
<b>vect</b>	producto cruz de vectores

## maxima-init.mac

<b>alias(det,determinant)</b>	abreviar nombre de funciones
<b>set_plot_option([plot_format,gnuplot])</b>	guardar graficos con gnuplot