# Práctica 1:Introducción a Maxima.
# Análise Matemática. Grao en Enxeñería Informática.
# E.S.E.I Ourense. Universidade de Vigo.

## 1 Primeros pasos con Maxima

Una vez descargado e instalado el programa, al ejecutarlo aparecerá una pantalla en blanco que será la hoja de trabajo en la que realizar las distintas operaciones:

(%i1) 3-7/8;
(%o1) $\frac{17}{8}$

(%i2) 2*5;
(%o2) 10

Una vez introducida una expresión hay que pulsar Enter ó Ctrl+Enter para obtener el resultado. Las expresiones (input) y sus correspondientes resultados (output) se agrupan en celdas a las que se les asigna una etiqueta %i1, %i2,... para las entradas y %o1, %o2,... para las salidas. El símbolo % hará referencia al último resultado obtenido

(%i3) 8*%o1;
(%o3) 17

(%i4) %+3;
(%o4) 20

Las operaciones habituales entre números se expresan con los operadores +,-,*,/,^ y se realizan en el orden de prioridad usual. Para alterar la prioridad entre las operaciones deben usarse paréntesis.

(%i5) 2-4;
(%o5) -2

(%i6) 3*6+2;
(%o6) 20

(%i7) 3*(6+2);
(%o7) 24

(%i8) 2^2+1;
(%o8) 5

(%i9) 2^(2+1);
(%o9) 8

(%i10) (7+1)/2^2;
(%o10) 2

En Maxima están definidas las principales constantes matemáticas:

(%i11) %e;
(%o11) %e

(%i12) %pi;
(%o12) $\pi$

```
(%i13) %i;
(%o13) %i
```

¡OJO!: No confundir con "e", "pi", "i" que en Maxima no representan por defecto ningún número.

En Maxima también están definidas las funciones de uso más común:
sqrt(x): raíz cuadrada de x
exp(x): exponencial de x
log(x): logaritmo neperiano de x
sin(x): seno de x
cos(x): coseno de x
tan(x): tangente de x
n!: factorial de n

¡OJO!: El ángulo x que aparece en las funciones trigonométricas debe estar expresado en radianes.

```
(%i14) sqrt(2);
```
(%o14) $\sqrt{2}$

```
(%i15) log(10);
```
(%o15) $\log(10)$

```
(%i16) exp(2);
```
(%o16) $\%e^{2}$

```
(%i17) sin(%pi);
(%o17) 0
```

```
(%i18) 7!;
(%o18) 5040
```

Maxima trabaja con dos clases de "números": exactos y aproximados (por ejemplo 1/3 y %pi son exactos mientras que 0.33333 y 3.14159 son aproximados). Cuando estemos utilizando solamente números exactos Maxima intentará dar la respuesta también de forma exacta. Sin embargo en cuanto algún número sea aproximado el resultado final en general también será aproximado.

```
(%i19) 1/4+1/5;
```
(%o19) $\frac{9}{20}$

```
(%i20) 0.25+1/5;
(%o20) 0.45
```

```
(%i21) sqrt(2)+sqrt(3);
```
(%o21) $\sqrt{3}+\sqrt{2}$

```
(%i22) sqrt(2.0)+sqrt(3.0);
(%o22) 3.146264369941973
```

Para obtener una aproximación decimal de un número exacto usaremos las instrucciones "float" o "bfloat" (disponibles en le menú "Numérico"):

```
(%i23) sqrt(2);
```
(%o23) $\sqrt{2}$

```
(%i24) float(sqrt(2));
(%o24) 1.414213562373095
```

```
(%i25) bfloat(sqrt(2));
(%o25) 1.414213562373095b0
```

```
(%i26) float(%pi);
(%o26) 3.141592653589793
```

```
(%i27) fpprec : 100$
       bfloat(%pi);
(%o28) 3.14159265358979323846264338327[43 digits]62862089986280348253421170679b0
```

Nota: para que se mostrasen en pantalla los cien primeros dígitos de %pi completos habría que utilizar
fpprec : 100$
set_display('ascii)$
bfloat(%pi);

## 2 Cálculo simbólico con Maxima

En Maxima podemos trabajar y operar también con expresiones simbólicas, por ejemplo:

1) Desarrollar una expresión

(%i29) expand((x+y)^4);

(%o29) $y^4+4\,x\,y^3+6\,x^2\,y^2+4\,x^3\,y+x^4$

2) Factorizar una expresión

(%i30) factor(y^4+4*x*y^3+6*x^2*y^2+4*x^3*y+x^4);

(%o30) $(y+x)^4$

3) Simplificar una expresión

(%i31) ratsimp((a^2-b^2)/(a-b));

(%o31) $b+a$

4) Resolver una ecuación o despejar una variable en una igualdad

(%i32) solve(x^2-3*x+2=0,x);

(%o32) $[x=1\,,x=2]$

(%i33) solve(a*x^2+b*x+c=0,x);

(%o33) $[x=-\frac{\sqrt{b^2-4\,a\,c}+b}{2\,a},x=\frac{\sqrt{b^2-4\,a\,c}-b}{2\,a}]$

(%i34) solve(x^2-2*y=1,y);

(%o34) $[y=\frac{x^2-1}{2}]$

5) Evaluar una expresión en un valor indicado de la variable

(%i35) subst(2,x,x^2+2);

(%o35) 6

(%i36) subst(-1,a,a-4);

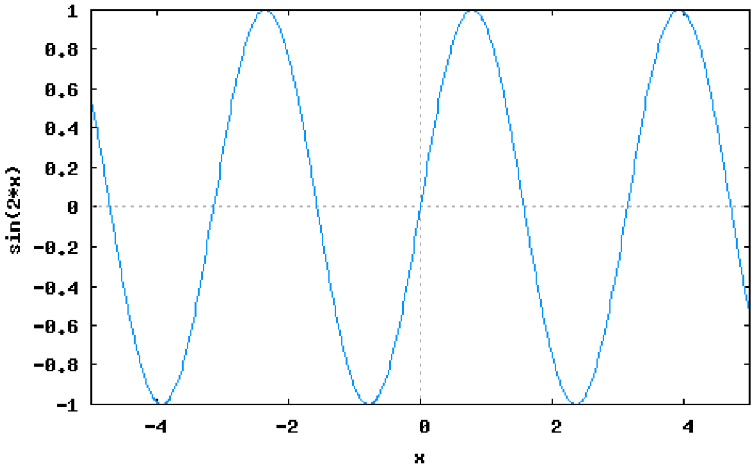
(%o36) -5

Otras operaciones simbólicas están disponibles en el menú "Simplificar"

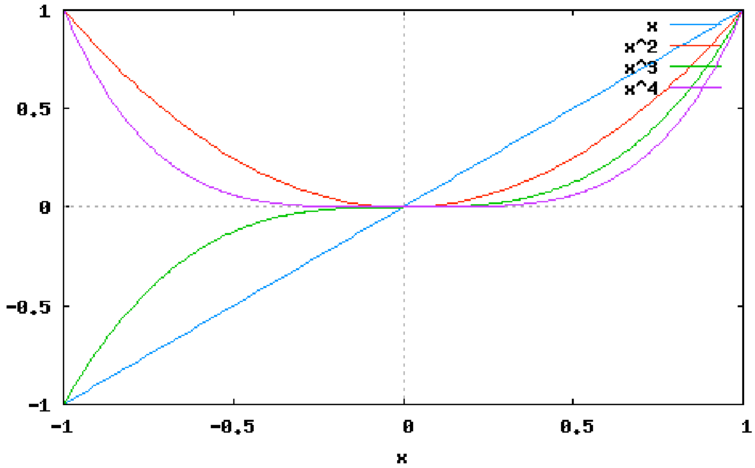
## 3 Gráficos con Maxima

Maxima permite representar gráficamente curvas en dos dimensiones mediante su ecuación explícita

```
(%i37) wxplot2d([sin(2*x)], [x,-5,5])$
```

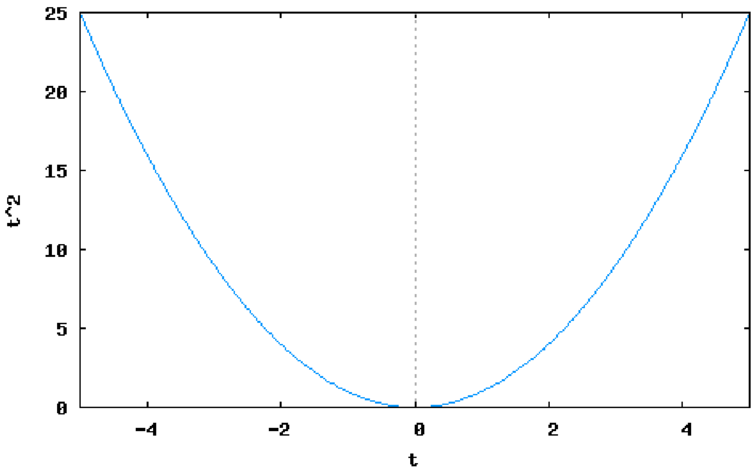
(%t37) 


```
(%i38) wxplot2d([x,x^2,x^3,x^4], [x,-1,1])$
```

(%t38) 

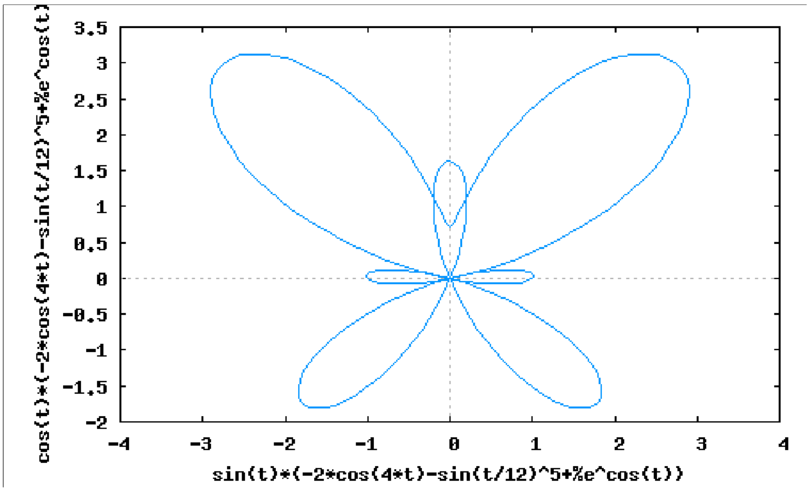

o sus ecuaciones paramétricas

```
(%i39) wxplot2d([['parametric, t, t^2, [t, -5, 5], [nticks, 300]]], [x,-5,5])$
```

(%t39) 

```
(%i40) wxplot2d([['parametric, sin(t)*(exp(cos(t))-2*cos(4*t)-sin(t/12)^5), cos(t)*(exp(cos(t))-2*cos(4*t)-sin(t/12)^5),
        [t, -%pi,%pi], [nticks, 300]]], [x,-4,4])$
```

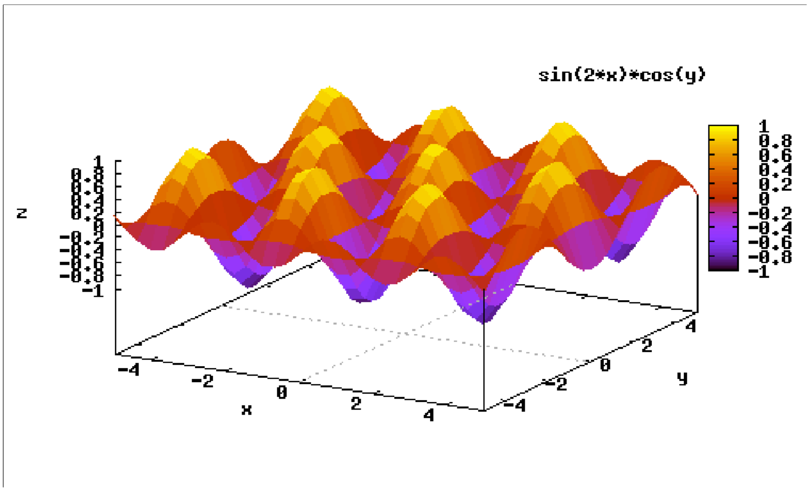
(%t40)




Maxima también permite representar gráficamente superficies en tres dimensiones:

```
(%i41) wxplot3d(sin(2*x)*cos(y), [x,-5,5], [y,-5,5])$
```

(%t41)




# 4 *Variables y funciones definidas por el usuario*

En Maxima también se puede programar definiendo bucles, instrucciones condicionales,...
En particular se pueden asignar valores a variables para ser usadas posteriormente:

```
(%i42) a:3;
(%o42) 3
```

```
(%i43) a^2;
(%o43) 9
```

```
(%i44) log(a);
(%o44) log(3)
```

Para evitar posibles conflictos y resultados inesperados es aconsejable borrar
todas las variables que hayan podido ser definidas previamente.

```
(%i45) kill(all);
(%o0) done
```

```
(%i1) a;
(%o1) a
```

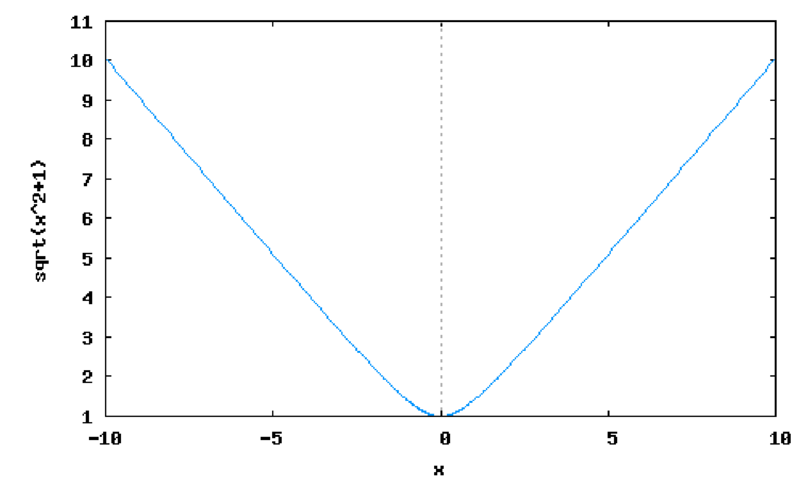
Como usuarios también podemos definir funciones:

```
(%i2) kill(all);
      f(x):=sqrt(x^2+1);
(%o0) done
```

(%o1) $f(x):=\sqrt{x^2+1}$

```
(%i2) wxplot2d([f(x)], [x,-10,10])$
```

(%t2)




Para definir una función a trozos usaremos la instrucción condicional if ... then ... else

```
(%i3) kill(all);
      f(x):=if x<0 then -x^2 else x^2+1;
(%o0) done
```

(%o1) $f(x):=\text{if } x<0 \text{ then } -x^2 \text{ else } x^2+1$

```
(%i2) wxplot2d([f(x)], [x,-1,1])$
```

(%t2)