

# Aide-mémoire pour Maxima

Paternité, Copyright  
Date, version

Fabien Baillon, EMAC  
20 novembre 2010, v1.0

## I. Maxima

Site officiel de Maxima : <http://maxima.sourceforge.net/>

Site officiel de l'interface wxMaxima : <http://wxmaxima.sourceforge.net>

Tutoriels : <http://wxmaxima.sourceforge.net/wiki/index.php/Tutorials>

FAQ : <http://cadadr.org/maxima/faq.html>

### Description générale

**Maxima** est un logiciel libre pour le calcul formel.

Maxima permet de :

- développer ou factoriser des expressions ;
- résoudre des équations ou des systèmes d'équations ;
- dériver ou intégrer des fonctions ;
- tracer des courbes en deux et trois dimensions ;
- manipuler des polynômes, des matrices, des complexes, etc.

---

## A Généralités

### constantes prédéfinies

`%pi ;`

`%e;`

`%i;`

`inf ;`

`minf ;`

### fonctions prédéfinies

`sqrt(9) ;`

`log(3) ;`

`cos(%pi/3) ;`

20 novembre 2010, v1.0

### définir une variable

```
a:5 ;
```

### définir une fonction

```
log10(x):= log(x)/log(10) ;
```

### obtenir le code LaTeX pour représenter un résultat

```
tex(%);
```

le symbole % fait référence au dernier résultat.

On peut aussi rappeler une entrée ou une sortie spécifique par son numéro : %i12 ou %o34, par exemple.

---

## B Factorisation, Développement ou Simplification

### factorisation polynomiale

```
factor(4*x^2-5*x-6) ;
```

```
=> (x-2)*(4*x+3)
```

### développement polynomial

```
expand((x-3)^6) ;
```

```
=> x^6-18*x^5+135*x^4-540*x^3+1215*x^2-1458*x+729
```

### développement d'expressions trigonométriques

```
trigexpand(sin(2*x)+cos(2*x)) ;
```

```
=> -sin(x)^2+2*cos(x)*sin(x)+cos(x)^2
```

### simplification d'expressions rationnelles

```
ratsimp((x^2-1)/(x+1)) ;
```

```
=> x-1
```

### simplification d'expressions trigonométriques

```
trigsimp(2*cos(x)^2 + sin(x)^2) ;
```

```
=> cos(x)^2+1
```

20 novembre 2010, v1.0

## C Résolution de systèmes

### résolution d'une équation

```
solve(x^2-4,x) ;
```

```
=> [x=-2,x=2]
```

### résolution d'un système d'équations

```
solve([x - 2*y = 14, x + 3*y = 9],[x,y]) ;
```

```
=> [[x=12,y=-1]]
```

### résolution d'un système trigonométrique

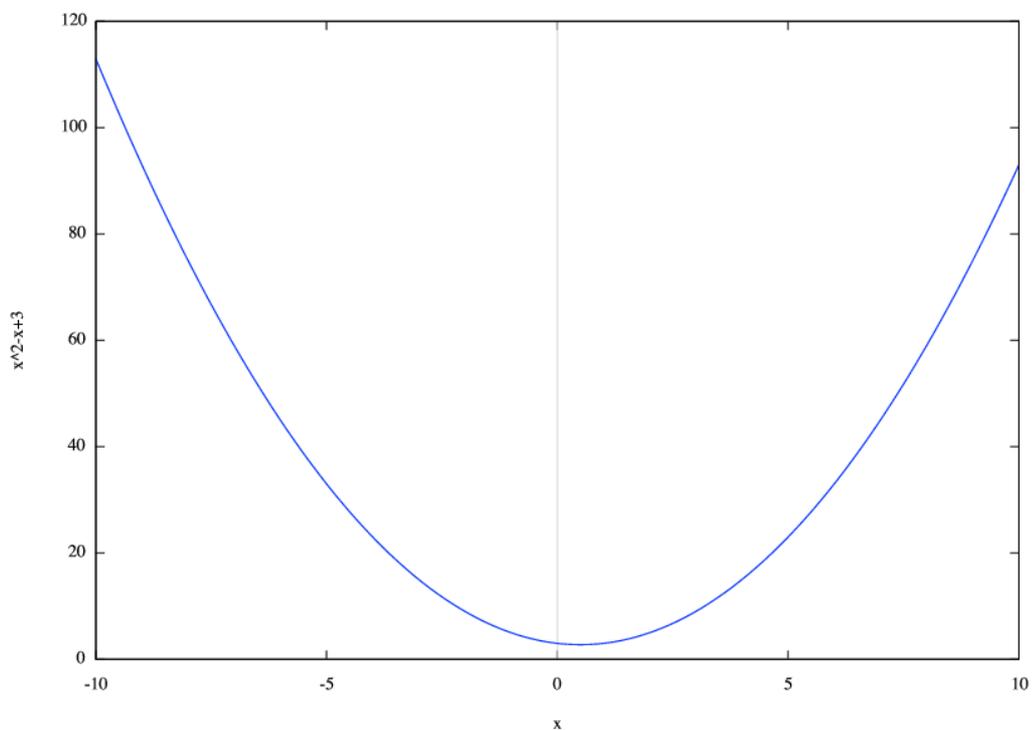
```
trigsimp(solve([cos(x)^2-x=2-sin(x)^2], [x])) ;
```

```
=> [x=-1]
```

## D Tracés de courbes

### tracé de courbes 2D

```
plot2d(x^2-x+3,[x,-10,10]) ;
```

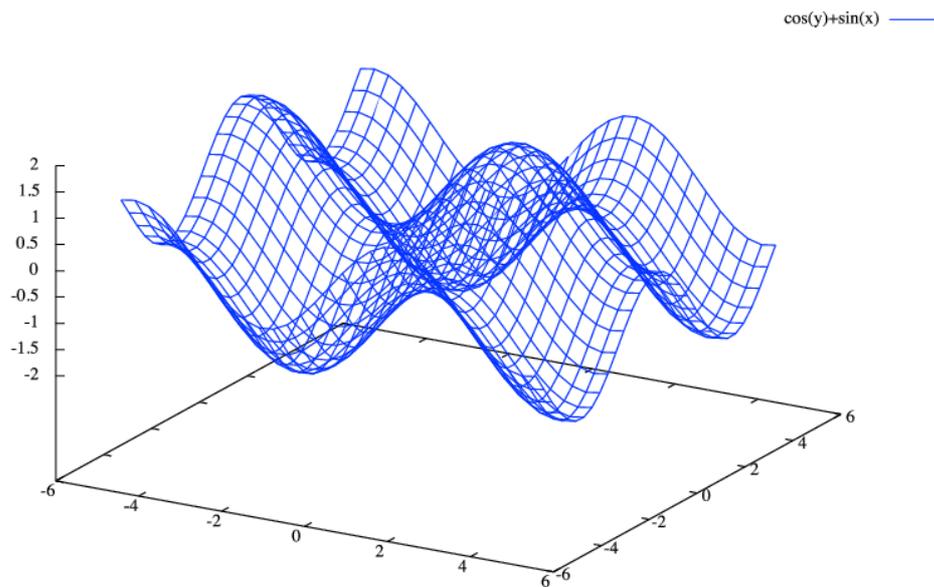


20 novembre 2010, v1.0

### tracé de courbes 3D

```
f(x,y):= sin(x) + cos(y) ;
```

```
plot3d(f(x,y), [x,-5,5], [y,-5,5]) ;
```



## E Limites et Différentielles

### limite d'une fonction

```
limit(2*(x^2-4)/(x-2),x,2) ;
```

=> 8

```
limit((1+1/x)^x,x,inf) ;
```

=> %e

### différentielle

```
diff(x^x, x) ;
```

=>  $x^x(\log(x)+1)$

On peut préciser un ordre supérieur, par exemple pour une différentielle à l'ordre 4 :

```
diff(tan(x), x, 4) ;
```

=>  $8*\sec(x)^2*\tan(x)^3+16*\sec(x)^4*\tan(x)$

---

## F Intégration

### intégration symbolique

`integrate(1/x, x) ;`

=>  $\log(x)$

### intégration définie en précisant les bornes

`integrate(x+2/(x-3), x, 0,1) ;`

=>  $-2*\log(3)+2*\log(2)+1/2$

---

## G Sommes et Produits

### Sommation

`sum(k, k, 1, n) ;`

=>  $\sum_{k=1}^n k$

On peut utiliser l'option *simpsum* pour simplifier le résultat :

`sum(k, k, 1, n), simpsum ;`

=>  $(n^2+n)/2$

### Produit

`product(k, k, 1, n) ;`

=>  $\prod_{k=1}^n k$

---

## H Séries

### développement en séries

selon *powerseries* ou

`niceindices(powerseries(%e^x, x, 0));`

=>  $\sum_{i=0}^{\infty} \left\{ \frac{x^i}{i!} \right\}$

selon la méthode de Taylor

`taylor(%e^x, x, 0, 5) ;`

=>  $1+x+x^2/2+x^3/6+x^4/24+x^5/120+\dots$

la fonction *trunc* permet de couper l'expression (pour supprimer le "+ ..."), notamment pour permettre le tracé du résultat

`plot2d([trunc(%), %e^x], [x,-5,5]);`

## Aide-mémoire pour Maxima > Maxima

20 novembre 2010, v1.0

