

# Quelques fonctions de base de Xcas pour le lycée



Xcas est un logiciel de calcul formel d'apparence peu conviviale mais qui s'avère très performant et assez simple d'utilisation.

Il permet par ailleurs d'utiliser le calcul formel au sein d'un tableur ou en géométrie dynamique en 2 ou 3 dimensions.

Voici quelques points forts du logiciel :

- ★ Il allie au sein d'une même interface des zones de calcul, un tableur, de la géométrie dynamique en 2 ou 3 dimensions et permet de programmer ;
- ★ Le logiciel est développé par des français. Les instructions, en ligne de commande, sont francisées et donc assez naturelles pour les élèves.
- ★ Il existe une documentation importante en français (<http://www-fourier.ujf-grenoble.fr/~parisse/giac/doc/fr/tutoriel/>)
- ★ L'aide en ligne, en français, est très bien faite ;
- ★ Logiciel multiplate-forme (Windows, Mac et Linux) ;
- ★ Un export en LaTeX est possible.

La présentation qui suit est loin d'être exhaustive. Elle présente quelques fonctions utiles, liées aux besoins de chaque niveau du lycée.

Le menu « scolaire » permet d'accéder directement à la plupart des fonctions décrites dans ce document.

La dernière section propose quelques algorithmes.

## Calcul numérique

### Dès la seconde

Fonction	Syntaxe	Résultat	Remarques
<b>Puissance</b>	$4^2$	16	Appuyez sur <input type="text"/> puis sur la touche « espace ».
<b>Racine carrée</b>	<code>sqrt(2)</code>	$\sqrt{2}$	
<b>Valeur approchée</b>	<code>approx(sqrt(2),15)</code>	1.414213562373096	Le second paramètre est optionnel. On modifie le nombre de décimales par défaut via le menu « cfg/configuration du CAS ».
<b>Simplifier une expression</b>	<code>simplifier(2*sqrt(2)+3*sqrt(8))</code>	$8 \cdot \sqrt{2}$	Les fractions sont mises sous forme irréductible sans même utiliser la commande « simplifier ».
<b>Parties entière et fractionnaire</b>	<code>propFrac(27/4)</code>	$6 + \frac{3}{4}$	
<b>Décomposition en produit de nombres premiers</b>	<code>factoriser_entier(300)</code>	$2^2 \cdot 3 \cdot 5^2$	
<b>Test de primalité</b>	<code>est_premier(1997)</code>	2	Renvoie 0 lorsque le nombre n'est pas premier

## calcul algébrique

### En seconde

<b>Développer</b>	developper((x+1)*(x+3))	$x^2 + 4x + 3$	<i>La présence du signe * est obligatoire.</i>
<b>Factoriser</b>	factoriser((x+1)^2+(x+1)*(2x+3))	$(x + 1) \cdot (3x + 4)$	<i>Le signe * est obligatoire. Xcas a parfois besoin d'aide pour factoriser (voir la sous-section « en première »).</i>
<b>Résoudre dans <math>\mathbb{R}</math> une équation</b>	resoudre(x^2+2x=8) resoudre(3t+6,t)	$\begin{bmatrix} -4 & 2 \\ -2 \end{bmatrix}$	<i>Par défaut, l'inconnue est « x ». Sans le signe « = », le second membre est considéré comme nul. « [] » signifie que l'équation n'a pas de solution dans <math>\mathbb{R}</math> « [x] » signifie que tous les réels sont solutions</i>

### En première

<b>Forme canonique</b>	forme_canonique(x^2-2*x+3)	$(x - 1)^2 + 2$	
<b>Factoriser</b>	factoriser(x^2+5x-6) factoriser(x^2+2x-1,sqrt(2))	$\frac{(x - 1)(x + 6)}{(x - \sqrt{2} + 1)(x + \sqrt{2} + 1)}$	<i>Le second paramètre (<math>\sqrt{2}</math>) est indispensable.</i>
<b>Fonction rationnelle</b>	propFrac((x+2)/(x-1))	$1 + \frac{3}{x - 1}$	
<b>Formules trigonométriques</b>	trigexpand(cos(a+b))	$\cos(a) \cos(b) - (\sin(a) \sin(b))$	

### En terminale

<b>Linéarisation des fonctions trigonométriques</b>	lineariser_trigo(sin(a)^3)	$\frac{3 \sin(a)}{4} + \frac{\sin(3a)}{-4}$	
<b>Développer un logarithme</b>	lnexpand(ln(x^2/(1+x)))	$2 \ln(x) - (\ln(x + 1))$	
<b>Développer une exponentielle</b>	expexpand(exp(2*x+1))	$(\exp x)^2 \exp 1$	

## Fonctions

### En seconde

<b>Définir une fonction</b>	$f(x) := x^2 + 2x + 1$ $g(t) := 1/(t+1)$		
<b>Définir une fonction à partir d'une expression</b>	$U := R * I$ $u := \text{unapply}(U, I)$		<i>U est une expression. u est une fonction dont la variable est I.</i>
<b>Calculer une image</b>	$f(\sqrt{2})$	$2 + 2\sqrt{2} + 1$	<i>La réponse n'est pas forcément simplifiée. Utiliser « simplify(f(sqrt(2))) » pour obtenir une valeur simplifiée.</i>
<b>Déterminer les antécédents</b>	$\text{resoudre}(f(x)=1)$ $\text{resoudre}(g(t)=3, t)$	$\begin{bmatrix} -2 & 0 \\ 2 & -\frac{1}{3} \end{bmatrix}$	<i>Par défaut, l'inconnue est « x ».</i>
<b>Tracer la courbe représentative</b>	$\text{graphe}(f(x), x=-3..2)$	Trace la courbe représentative de $f$ sur $[-3; 2]$	<i>Le deuxième paramètre est optionnel.</i>
<b>Comparer graphiquement deux fonctions</b>	$\text{graphe}([f(x), g(x)], x=-3..2)$	Trace les courbes représentatives de $f$ et de $g$ sur $[-3; 2]$ .	<i>Noter l'utilisation de « g(x) » et non « g(t) ».</i>

### En première

<b>Dérivée d'une expression</b>	$f(x) := 2x^3 - 4x^2 + 4$ $g := \text{deriver}(f(x))$	$x \mapsto 2x^3 - 4x^2 + 4$ $2 \cdot 3 \cdot x^2 - 4 \cdot 2 \cdot x$	<i>g est alors une « expression » et non une fonction. L'expression n'est pas simplifiée. Pour cela, utiliser « g := simplifier(deriver(f(x))) ».</i>
<b>fonction dérivée</b>	$g := \text{unapply}(g, x)$ ou directement $g := \text{fonction\_derivee}(f)$	$x \mapsto 2 \cdot 3 \cdot x^2 - 4 \cdot 2 \cdot x$	<i>g est maintenant une fonction. On peut donc calculer des images, redériver...</i>

### En terminale

<b>Définir un paramètre</b>	$k := \text{element}(0..10, 5, 0.2)$	$k$ varie entre 0 et 10 avec un pas de 0,2. Sa valeur par défaut est 5. <b>Un curseur permet de modifier cette valeur.</b>	<b>Ouvrir au préalable, un graphique via le menu « edit/ouvrir/graph géo2D ».</b>
<b>Fonction dépendant d'un paramètre</b>	$f(t) := k \cdot \ln(t)$	$t \mapsto k \cdot \ln(t)$	<i>Un avertissement s'affiche indiquant que k est une « variable globale ».</i>
<b>Représenter une fonction dépendant d'un paramètre</b>	$\text{graphe}(f(t), t)$ ou $\text{graphe}(f(x))$	Représente la fonction $x \mapsto 5 \ln x$ .	<i>Par défaut, la variable pour un graphique est « x ».</i>

# Les nombres complexes

## En terminale S

<b>Mise sous forme algébrique</b>	$2*i+(3+i)^4$ $(1+i)/(3-i)$ $\exp(i*\pi/2)$	$28 + 98 * i$ $\frac{1+2*i}{5}$ $i$	<i>La mise sous forme algébrique est automatique</i>
<b>Partie réelle, imaginaire et conjugué</b>	$[\text{re}(i), \text{im}(i), \text{conj}(i)]$	$[ \quad 0 \quad 1 \quad -i \quad ]$	
<b>Résolution d'une équation dans <math>\mathbb{C}</math></b>	$\text{resoudre\_dans\_C}(z^2+2*\text{conj}(z)+6,z)$ ou $\text{csolve}(z^2+2*\text{conj}(z)+6,z)$	$[ \quad 1+3*i \quad 1-3*i \quad ]$	<i>Le second membre est 0 par défaut. Le seconde paramètre, « z », est indispensable.</i>
<b>Représentation graphique</b>	$a:=1+i$ $A:=\text{point}(a)$ $b:=-2+1$ $B:=\text{point}(b)$ $C:=\text{point}(a*b)$ $c:=\text{affiche}(C)$	Place les points A, B et C d'affixes $a$ , $b$ et $c$ dans un repère	<i>Ouvrir un graphique via le menu « edit/ouvrir/graph géo2D Voir la section « Géométrie dynamique » pour plus d'informations.</i>
<b>Transformations complexes</b>	$f(z):=(z+i)/(z+1)$ $\text{re}(f(x+i*y))$ $\text{plotimplicit}(\text{re}(f(x+i*y)))$	$z \mapsto \frac{z+i}{z+1}$ $\frac{x(x+1)}{(x+1)^2+y^2} - \left( \frac{(y+1)(-y)}{(x+1)^2+y^2} \right)$ Trace la courbe	<i>La partie réelle demandée n'est pas simplifiée. Pour cela, utiliser « simplifier(re(f(x+i*y))) ». Voir la rubrique « géométrie » pour plus d'informations sur la fonction « plotimplicit ».</i>

## Statistiques

## Géométrie

# Programmation

Pour créer un programme :

1. choisir « programme » dans le menu « edit/ajouter » ;
2. choisir « configuration du cas » dans le menu « cfg ». Sélectionner alors « xcas » dans la liste « prog style » ;
3. saisir le code du programme ;
4. le valider en cliquant sur OK.

Pour utiliser le programme : il suffit de l'appeler par son nom (sans oublier les parenthèses).

Objectif	Programme	Remarques
<b>Fonction affine par morceaux</b>	<pre>f(x) := {   si (x&gt;0 &amp;&amp; x&lt;3)   alors return 2*x;   sinon return 3*x+1;   fsi ; }</pre>	<p><i>On peut alors faire des calculs d'images (f(4)), tracer la courbe (plot(f(x),x)).</i></p> <p><i>Par contre, la résolution d'équations ne fonctionne pas.</i></p>
<b>Plus grand nombre premier inférieur à <math>n \geq 2</math></b>	<pre>premierinf(n) := {   pour k de n jusque 2 pas -1   faire     si est_premier(k)     alors return k;   fsi   fpour }</pre>	<p><i>On ne peut pas utiliser « i » comme nom de variable.</i></p> <p><i>La fonction « return » stoppe la boucle « pour ».</i></p>
<b>Suite de Syracuse</b>	<pre>paire(n) := {   si n/2==round(n/2)   alors return true;   sinon return false;   fsi ; }  suite(n) := {   local p;   p := n;   Disp(p);   tantque p != 2   faire     si paire(p)==true     alors p := p/2;     sinon p := 3*p+1;   fsi   print(p);   ftantque }</pre>	<p><i>« Disp » permet d'afficher un résultat intermédiaire.</i></p>
<b>Nombres complexes et spirale</b>	<pre>spiral() := {   local a := 1;   local b := 1;   pour k de 1 jusque 96 faire     b := a*exp(i*pi/12)*0.99;     DispG(couleur(point(a),red);     DispG(segment(a,b));     a := b;   fpour }</pre>	<p><i>« DispG » ouvre une fenêtre graphique et dessine des éléments intermédiaires.</i></p>