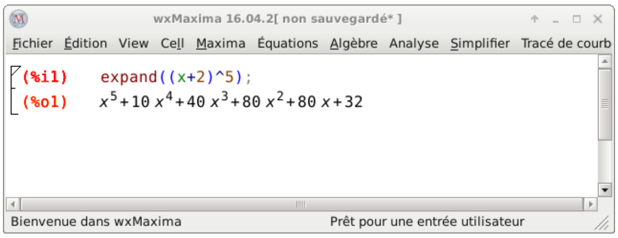
A. Introduction:

Maxima a été développé au départ par le MIT dans les années 1960 puis maintenu par un collectif de programmeurs qui ont mis le logiciel dans le domaine libre.

Maxima permet de faire des calculs sur les polynômes, les matrices, de l'intégration, de la dérivation, des nombres complexes. Par contre, les capacités graphiques sont très limitées en comparaison avec Mathematica et Maple.

Maxima est un logiciel de calcul formel et sa syntaxe d'utilisation est très stricte. Il possède un mode décimal permettant d'afficher.

B. Prise en main:



Dans la fenêtre de Maxima, il faut repérer :

- les lignes `(%i xxx)` sont les entrées de l'utilisateur ;
- les lignes `(%o xxx)` sont les résultats de Maxima

Lorsqu'on souhaite soumettre une instruction à Maxima, il faut utiliser la combinaison de touches :
"`Ctrl + Entrée`"

1. Manipulation algébrique :

Exemple :

Pour déterminer les réels a, b, c, réalisant l'identité :

$$(x+1)(a{\cdot}x^2+b{\cdot}x+c) = -2{\cdot}x^3+3{\cdot}x^2+8{\cdot}x+3$$

On utilise les deux commandes suivantes :

```
> rat((x+1)*(a*x^2+b*x+c));
> solve([a=-2,b+a=3,c+b=8,c=3],
        [a,b,c])
```

Exemples d'utilisations de Maxima pour la construction d'exercices :

rq78-3 rq78-14

2. Autour de l'analyse :

Exemple :

On cherche les équations réduites des tangentes à la courbe représentative de la fonction f définie par :

$$f(x) = x^2 - x + 2$$

On utilise les instructions ci-dessous :

```
> a:1;
> f:x^2-2*x+2;
> g:diff(f,x);
> t:ev(g,x=a)*(x-a)+ev(f,x=a);
> expand(t)
```

Et maintenant, voici les instructions pour déterminer l'abscisse du point de contact de la tangente à la courbe $\mathscr{C}_f$ passant par l'origine du repère :

```
> kill(all);
> f:x^2-2*x+2;
> g:diff(f,x);
> t:ev(g,x=a)*(x-a)+ev(f,x=a);
> solve(ev(t,x=0)=0)
```

Pour tracer la courbe de la fonction f et cette tangente, on saisit le code ci-dessous :

```
plot2d([x^2-2*x+2,2*sqrt(2)*x-2*x],
       [x,-4,4]);
```

Exemple :

On souhaite trouver les zéros de la fonction f définie par :

$$f(x) = e^x + x$$

Voici les deux commandes à utiliser pour afficher la courbe représentative de la fonction f :

```
> f:exp(x)+x;
> plot2d(f,[x,-20,20]);
> find_root(f,x,-5,5);
```

On affine les bornes d'affichages de l'axe des abscisses, puis on cherche la valeur approchée :

```
> f:exp(x)+x;
> plot2d(f,[x,-20,5]);
```

Exemples d'utilisations de Maxima pour la construction d'exercices :

rq78-7 rq78-12 rq78-19 rq78-26 rq78-32

3. Ecritures décimales :

Maxima est un logiciel de calcul formel. A ce titre, il ne travaille jamais avec les valeurs approchées à moins que cela lui soit demandé :

Exemple :

Comment établir l'égalité ci-dessous?

$$5^{100} + 2^2{\times}5^{100} = 5^{101}$$

Avec Maxima, il suffit de les calculer!

```
> 5^100+2^2*5^100;
> 5^101
```

Exemple :

La commande `float()` permet d'afficher la valeur approchée d'un nombre :


```
> fpprec:100;
> printf(false,"~1h",bfloat(1/17));
```

On peut repérer le développement décimal du nombre rationnel $\dfrac{1}{7}$ mais pas celui de $\dfrac{1}{17}$, voici les commandes à utiliser :

```
> f:exp(x)+x;
> plot2d(f,[x,-20,5]);
```

4. Nombres complexes :

Exemple :

Soit z un nombre complexe quelconque. On définit le nombre complexe z' par : $z' = \dfrac{(3+4{\cdot}i)\cdot z+5\cdot\overline{z}}{6}$

Pour établir que le nombre $\dfrac{z'-z}{1+2{\cdot}i}$ est un nombre réel :

```
z:x+%i*y;
zz:((3+4*%i)*z+5*conjugate(z))/6;
m:(zz-z)/(1+2*%i);
rectform(expand(m));
```

Exemple :

Considérons la fonction complexe f définie sur $\mathcal{C}\backslash\{2\}$ par :

$f : z \longmapsto z^2 + 4\cdot z + 3$

Pour déterminer, l'ensemble des nombres complexes ayant pour image un nombre réel, on utilise les commandes suivantes :

```
z:x+%i*y;
zz:((3+4*%i)*z+5*conjugate(z))/6;
m:(zz-z)/(1+2*%i);
rectform(expand(m));
```

Exemples d'utilisations de Maxima pour la construction d'exercices :

5. Matrices :

Exemple :

On considère les matrices : $A = \begin{pmatrix} 4 & 4 \\ 0 & 2 \end{pmatrix}$; $P = \begin{pmatrix} 0 & -2 \\ 1 & 2 \end{pmatrix}$

Voici les commandes utilisées pour montrer que la matrice $P\cdot A\cdot P^{-1}$ est diagonale et que, pour tout entier naturel n, la matrice A^n admet pour expression :

$$A = \begin{pmatrix} 4^n & 2\left(4^n - 2^n\right) \\ 0 & 2^n \end{pmatrix}$$

```
> A:matrix([4,4],[0,2]);
> P:matrix([0,-2],[1,2]);
> D:P.A.invert(P);
> B:invert(P).D^n.P;
> factor(B);
```

Exemples d'utilisations de Maxima pour la construction d'exercices :

rq78-8 rq78-10 rq78-16 rq78-36

6. Probabilités :

Exemples d'utilisations de Maxima pour la construction d'exercices :

rq78-30

7. Vers la programmation :

Maxima possède en interne un langage de programmation (structure conditionnelle, répétitive) qui permette de tester un grand nombre valeur et de n'afficher que celles qui réalisent une condition voulue.

Exemple :

Voici un exemple permettant d'afficher les premiers triplets pythagoriciens :

```
for a:1 thru 100 do
 for b:a thru 100 do
  block(c:sqrt(b^2+a^2),
   if (floor(c)#c) then go(loop),
   display([a,b,c]),
   loop
  );
```