Chapitre 1

Introduction et aspect super calculatrice

Le calcul formel consiste à manipuler des objets mathématiques sur ordinateur, l'objectif est de faire des calculs mathématiques sur ordinateur sans se limiter au calcul numérique comme c'est le cas dans les langages de programmation machines (Pascal,C, Fortran).

Exemple 1

calcul numérique (C, Pascal, Fortran)	$'\frac{1}{3}+\frac{1}{5}' : 0.333\,33 + 0.2 = 0.533\,33$
calcul formel (Maple, Mathematica ,...)	$\frac{1}{3}+\frac{1}{5} = \frac{3+5}{15} = \frac{8}{15}$

Exemple 2 *Une différence importante entre les langages de programmation machines (Pascal,C, Fortran) et les logiciels de calcul formels consiste en la capacité à manipuler des variables non assignées.*

calcul impossible sur C, Pascal, Fortran	$'x + x'$
calcul formel (Maple, Mathematica ,)	$x + x = 2x$

Les logiciels de calcul formel offrent la possibilitée de programmation numérique en plus de la capacitée symbolique, ils contiennent en outre plusieurs bibliothèques de procédures spécialisés pour permettre de répondre à la plupart des taches de calcul connus.

1.1 Premiers pas en Maple

- On termine une commande par un " **;**" si on désire afficher les résultats de l'instruction.
- On termine une commande par un " **:**" si on désire exécuter une commande sans afficher le résultat.
- On peut être amené à vouloir exécuter plusieurs opérations mais n'être intéressés que par le dernier résultats dans ce cas la commande shift+return permet d'introduire de nouvelles opérations sans provoquer d'évaluation prématurée.
- Si l'on affecte à une variable une valeur , la variable est dite assignée . si elle n'est associée à aucune valeur , la variable est dite non assignée . l'affectation sous Maple se fait par le biais de l'instruction :=
- Maple fait la distinction entre majuscule et minuscules.
- On ne peut pas utiliser les noms de constantes ou de fonctions Maple comme noms de variable.
- La commande **restart** permet d'annuler les affectations au cours d'une session Maple.
- Le symbole = est utilisé sous Maple pour désigner une équation.
- La commande % permet de faire référence au dernier résultat calculé par Maple et l'utiliser directement.

Exemple 3

```
> Delta:=b^2-4*a*c;X+x;X+X;
```

$$\Delta := -4ac + b^2$$

$$X + x$$

$$2X$$

```
> Pi:=3.14;
Error, attempting to assign to `Pi` which is protected.

 Try declaring `local Pi`; see ?protect for details.
```

1.1.1 Les nombres dans Maple

Les nombres sont divisés en 4 types :entier (integer), rationnel (fraction) ,réel (float) et complexe (complex), ll faut éviter la confusion entre les noms des types et les propriétés ma-

thématiques des nombres.

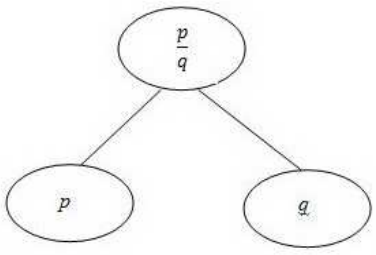
Type entier (integer)

Les calculs sur les entiers sont effectués en précision infinie.

Type rationnel (fraction)

Un rationnel est défini dans maple comme le rapport de deux entiers.

Maple traite les rationnels en utilisant des règles algébriques, chaque rapport d'entiers est remis automatiquement sous forme irréductible.



Représentation par arbre de l'objet fraction

Exemple 4

```
> restart:
  3./2;3/2;
```

$$1.500000000$$

$$\frac{3}{2}$$

```
> 60/45;16/12;
```

$$\frac{4}{3}$$

$$\frac{4}{3}$$

Type réel

Maple manipule les réels et les complexes en précision flottante finie arbitraire.

Un nombre réel se représente en Maple sous la forme de sa partie entière e suivi d'un point

puis de sa partie décimale d càd $e.d$

Une variable globale Digits permet de fixer le nombre de chiffres que Maple donnera dans ses résultats numériques.

La commande **evalf** permet de forcer une évaluation approchée d'un calcul.

evalf signifie : **Eval**uate using **f**loating point arithmetic.

Exemple 5

```
> Pi;evalf(Pi);Digits:=100;evalf(Pi);
```

$$\pi$$

$$3.141592654$$

$$Digits := 100$$

$$3.14159265358979323846264338327950288419716939937510582097494459230781640628620899862803482534211$$

Type complexe

Le type complexe est crée en manipulant la constante I base des nombres complexes sous Maple.

Constantes mathématiques :

En plus de I la base des imaginaires purs I^2=-1 plusieurs constantes mathématiques sont implémentés : Le nombre π est noté Pi, le symbôle ∞ est noté infinity.

Fonctions mathématiques :

La commande abs

- La commande abs tente de calculer la valeur absolue d'une expression x, si x n'est pas un nombre la commande essaie de déterminer le signe de l'expression pour ensuite donner sa valeur absolue.
- Si x est complexe alors la commande abs calculera son module.

Exemple 6

```
> abs(-3);abs(-x);abs(x^2);abs(x^2+1);abs(I+1);
```

$$3$$
$$|x|$$
$$|x|^2$$
$$|x^2+1|$$
$$\sqrt{2}$$

La commande sqrt : square root

- La commande sqrt(x) tente de calculer la racine carré d'une expression x.
- Si x est négatif ou complexe alors la commande calculera sa racine complexe.

Exemple 7

```
> sqrt(12.);sqrt(12);sqrt(-4);sqrt(3+4*I);sqrt(x^2);
```

$$3.464101615$$
$$2\sqrt{3}$$
$$2\,I$$
$$2+I$$
$$\sqrt{x^2}$$

1.2 Manipulations d'expressions

1.2.1 L'instruction simplify

Appel type : simplify(expression) ;

La commande simplify applique des règles de simplification prédéfinies à des expressions, les principes suivants sont appliqués :

- Réduction au même dénominateur.
- La règle $(x^p)^q = x^{p.q}$
- Règles usuelles reliant l'exponentielle au logarithme.
- Transformation d'une expression trigonométrique en polynôme trigonométrique.

L'instruction simplify + option

Appel type : simplify(expression,option1,option2...) ;

L'instruction simplify possède des options qui permettent d'appliquer des règles spécifiques à des types d'expressions particulières, nous citons quatres options utiles :

- L'option powers : elle est utilisée lorsque l'on veut simplifier des expressions contenant des puissances, des logarithmes ou des exponentielles.
- L'option radical : elles simplifie les expressions contenant des puissances fractionnaires.
- L'option sqrt : elle permet de simplifier les expressions qui font intervenir des racines carrées.
- l'option trig :elle est utilisée pour simplifier des expressions trigonométriques.

Exemple 8

```
> exp(b·ln(x));
```

$$e^{b \ln(x)}$$

```
> simplify(%);
```

$$x^{b}$$

```
> cos(x)^2 + sin(x)^2;
```

$$\cos(x)^2 + \sin(x)^2$$

```
> simplify(%);
```

$$1$$

```
> (x+1)^(4/3) - x·(x+1)^(1/3);
```

$$(x+1)^{4/3} - x\,(x+1)^{1/3}$$

```
> simplify(%, radical);
```

$$(x+1)^{1/3}$$

1.2.2 L'instruction expand

Appel type :**expand**(*expr*);

L'instruction expand permet de développer des expressions, elle applique les règles suivantes :

- Distribution des sommes sur le produit dans le cas d'un polynôme à plusieurs variables.
- Si l'expression est une fraction rationnelle P/Q l'instruction expand distribue 1/Q sur P et applique à P les règles cités précédemment.
- L'instruction expand peut être utilisée dans d'autres cas que celui du calcul polynomial.

Exemple 9

```
> expand((x+1)·(x+2));
                              x^2 + 3 x + 2
> expand((x+y)·(y+z));
                              x y + x z + y^2 + y z
> expand(cos(a+b));
                              cos(a) cos(b) - sin(a) sin(b)
```

1.2.3 L'instruction combine

L'instruction combine effectue dans plusieurs cas les transformations inverses d'expand, elle permet de recombiner des termes à l'intérieur d'une expression.

Considérons par exemple l'identité mathématique connue :

$$\sin(a+b) = \sin(a)\cos(b) + \cos(a)\sin(b)$$

L'instruction expand applique cette identité de gauche à droite alors que l'instruction combine applique l'inverse.

La liste des options de l'instruction combine comprend également les options **ln, exp, power ,trig.**

Exemple 10

```
> sin(x)*cos(y)+cos(x)*sin(y);
                              sin(x) cos(y) + cos(x) sin(y)
> combine(sin(x)*cos(y)+cos(x)*sin(y));
                              sin(x + y)
> 2*cos(x)^2-1;
                              2 cos(x)^2 - 1
> combine(2*cos(x)^2-1);
                              cos(2 x)
```

1.3 Opérations élementaires d'analyse

1.3.1 Définition d'une fonction

L'opérateur d'association -> permet d'associer un (ou plusieurs) nom de variables à une (ou plusieurs) expressions pour créer une fonction au sens mathématique du terme.

Exemple 11

```
> f := x -> x^2 + 1;
```

$$f := x \to x^2 + 1$$

```
> f(0);
  f(alpha/2);
```

$$1$$

$$\frac{1}{4}\,\alpha^2 + 1$$

```
> g := (x, y) -> (x^2 + y^2, x*y);
```

$$g := (x, y) \to (x^2 + y^2, x\,y)$$

```
> g(1, 2);
```

$$5, 2$$

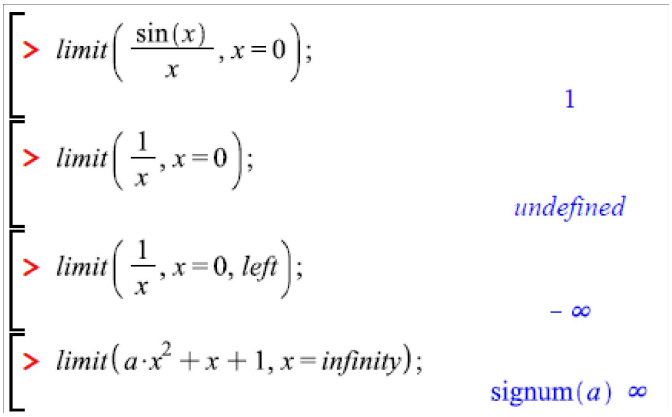
1.3.2 Calcul de limites

Appel type : limit$(f, x = a)$. **limit**$(f, x = a, left)$. **limit**$(f, x = a, right)$.

La commande **limit** tente quand c'est possible de calculer la limite d'une fonction f , la variable a peut indiquer un nombre réel ou une borne infinie. $(\pm\infty)$.

Si la fonction f contient des variables libres elles sont considérés par Maple comme des réels non nuls fixés.Maple arrive à repérer un certain nombre de cas de limites qui n'existent pas et renvoie la valeur "$undefined$".

Exemple 12

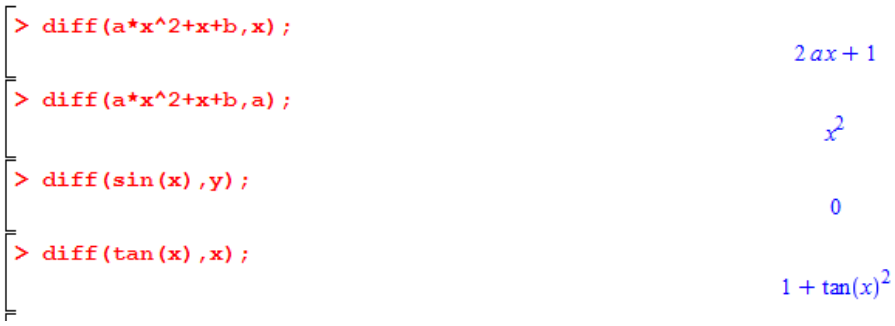


1.3.3 Dérivation d'expression : l'instruction diff

Appel type : **diff**$(expr, x)$

La commande diff demande au logiciel de dériver une expression *expr* par rapport à une variable, si l'expression contient d'autres variables, elles sont considérés comme constantes.

Exemple 13



1.3.4 fonction dérivée : l'opérateur D

Appel type : **D**(f)

Où f est une fonction déjà définie en cours de session ou une fonction mathématique interne.

L'opérateur de dérivation est un appel qui permet d'avoir la fonction dérivée, il peut servir également à obtenir la dérivée d'une expression.

Exemple 14 *L'opérateur crée une fonction qui associe à une variable la dérivée de la fonction*

f de l'appel à l'image de la définition mathématique.

$$D(f) : x \to f'(x)$$

```
> D(cos);
```

$$-\sin$$

```
> D(ln);
```

$$z \to \frac{1}{z}$$

```
> f := x→x^2 + 3·x;
  D(f);
```

$$f := x \to x^2 + 3x$$
$$x \to 2x + 3$$

```
> D(f)(5);
```

$$13$$

1.3.5 Compositions des fonctions : opérateur @

Appel type $f@g$, $f@@n$

L'opérateur @ permet de composer des fonctions entre elles, la commande $f \circ g$ correspond à la notation mathématique $f \circ g$.

La composition d'une fonction avec elle même n fois est obtenue en utilisant l'opérateur @@, la commande $f@@n$ permet d'obtenir $\underbrace{f \circ f ... \circ f}_{n\ fois}$.

Remarque 15 *Il faut remarquer que Maple utilise la notation usuelle de la dérivée* $n-ième$ *pour noter la composée* $n-ième$.

Exemple 16

```
f := x→x^2 + 1; g := x→cos(x) + 1;
```

$$x \to x^2 + 1$$
$$x \to \cos(x) + 1$$

```
f@g;
```

$$f@g$$

```
(f@g)(x);
```

$$(\cos(x) + 1)^2 + 1$$

```
simplify(%);
```

$$\cos(x)^2 + 2\cos(x) + 2$$

```
(f@@2)(x);
```

$$(x^2 + 1)^2 + 1$$

1.3.6 Singularié d'une fonction

Appel type : **singular**$(f(x))$, **singular**$(f(x), a..b)$

- Cette fonction essaie de déterminer l'ensemble des singularités d'une fonction, la commande **singular** ne fais pas de distinction entre les singularités qui sont simplifiables et les autres. .
- Si la fonction contient plus d'une variable l'utilisateur doit spécifier la variable par rapport à laquelle il souhaite chercher les singularités.
- Si on spécifie un intervalle seuls les singularités contenues dans cet intervalle seront affichés.

Remarque 17 *Parmis les résultats affichés par Maple on peut rencontrer plusieurs types de noms : _Zn désigne ici les entiers relatifs , _Nn désigne les entiers naturels.*

1.3.7 Etude de contuinuité d'une fonction (iscont & discont)

Appel type : **iscont**$(f(x), x = a..b)$

- Cette fonction permet de tester la continuité d'une expression sur un intervalle réel donné considéré ouvert par défaut. Il y a trois réponses possibles : true (la fonction est continue), false (elle ne l'est pas), FAIL (iscont ne peut pas répondre).
- Les extrémités de l'intervalle peuvent être de type réel ou infinity ou - infinity.
- On peut rajouter l'option **closed** à la fin pour signifier que l'intervalle est fermé.

Appel type : **discont**$(f(x), x)$

Cette fonction essaie de déterminer l'ensemble des points de discontinuité sur tout l'ensemble des réels pour une fonction.

Exemple 18

```
> singular(x/(x^2-1));
> singular((x-1)/(x^2-1));
> singular(1/(a*x^2));
> singular(1/(a*x^2),x);
> singular(tan(x));
```

$\{x = -1\}, \{x = 1\}$

$\{x = -1\}$

$\{a = 0, x = x\}, \{a = a, x = 0\}$

$\{x = 0\}$

$\left\{x = \frac{1}{2}\pi + \_Z1 \sim \pi\right\}$

1.3.8 Développement en série de Taylor :

Appel type : **taylor**$(f, x = a, n)$

L'instruction type taylor$(f, x = a, n)$ renvoie une série correspondant aux n premiers termes du dévellopement de Taylor autour de a de la fonction f considérée comme une fonction de la variable x.

Si aucun ordre n'est donnée la valeur par défaut est 6. si aucun voisinage n'est donné le voisinage par défaut est 0.

Maple n'affiche pas le reste tel que définit dans les ouvrages de références mathématiques, il n'affiche pas en effet le reste qui dépend de la dérivée d'ordre $(n+1)$ mais se contente d'utiliser l'écriture propre au dévellopement limité.

Remarque 19 *La commande convert (....,polynom) nous permet de convertir le dévellopement de taylor pour ne retenir que le polynôme, l'écriture type est donnée par :*

$$convert(taylor(f(x), x = a, n), polynom);$$

Exemple 20

```
> taylor( exp(x), x=0, 4 );
```

$$1 + x + \frac{1}{2}x^2 + \frac{1}{6}x^3 + O(x^4)$$

```
> taylor( 1/x, x=1, 3 );
```

$$1 - (x-1) + (x-1)^2 + O((x-1)^3)$$

```
> convert(taylor(exp(x),x=0,4),polynom);
```

$$1 + x + \frac{1}{2}x^2 + \frac{1}{6}x^3$$

1.3.9 Intégration & calcul de primitives

Appel type : **int**$(f(x), x = a..b)$

l'instruction **int** permet de calculer des primitives ou des intégrales.

- Si les bornes $a..b$ ne sont pas donnés alors la commande calculera la primitive de $f(x)$ par rapport à x.
- La primitive est donné dans sa forme principale, Maple n'affiche pas de constante d'intégration ainsi pour $> int\left(x^2, x\right)$; le logiciel affichera $\frac{x^3}{3}$ et non pas $\frac{x^3}{3} + c$.
- Si les bornes a, b sont donnés alors la commande calculera l'intégrale $\int_a^b f(x)\, dx$.
- Maple est capable d'identifier les singularités d'une fonction sur l'intervalle d'intégration et renvoie la valeur *undefined* comme résultat dans ce cas.

Exemple 21

```
> int(ln(x),x);
```

$$x\ln(x) - x$$

```
> int(alpha*sqrt(x),x);
```

$$\frac{2}{3}x^{3/2}\alpha$$

```
> int(exp(-x)/x,x);
```

$$-Ei(1, x)$$

Exemple 22

```
> int(1/(x-1),x=3..6);
```

$$-\ln(2)+\ln(5)$$

```
> int(a/(x-1),x=3..6);
```

$$-\ln(2)\,a+\ln(5)\,a$$

```
> int(exp(-x^3),x=3..6);
```

$$\int_3^6 e^{-x^3}\,dx$$

```
> evalf(%);
```

$$6.799019886\,10^{-14}$$

```
> int(exp(-x)/x,x=3..6);
```

$$Ei(1,3)-Ei(1,6)$$

```
> evalf(%);
```

$$0.01268829864$$

```
> int(1/(x-1),x=-2..2);
```

undefined

1.3.10 Fonctions définis par morçeaux

Appel type : **piecewise**$(condition1, f_1, condition2, f_2, ..., conditionN, f_N, f_{ailleurs})$

MAPLE permet, avec **piecewice** de définir des expressions composées par intervalles. Il suffit de donner dans l'ordre des arguments, un domaine défini par une condition suivie de la définition de la fonction dans cet intervalle, et ce, autant de fois que nécessaire. Le dernier argument définit la fonction partout ailleurs (otherwise).

Si la fonction $f_{ailleurs}$ n'est pas donné dans la commande le logiciel assumme qu'elle est nulle.

Il faut bien noter que la commande **piecewice** ne définit pas une fonction mais une expression, si on souhaite créer une fonction défini par morceaux il faut en plus utiliser l'opérateur d'association ->.

Les conditions sont donnés en général sous formes d'inégalités, ou par composition d'inéga-

lités par des opérateurs logiques AND, OR, le tableau ci dessous détaille quelques possiblités :

Ecriture mathématique	Notation Maple	Ecriture mathématique	Notation Maple
$x < a$	x<a	$x > a$	x>a
$x \leq a$	x<=a	$x \geq a$	x>=a
$a < x < b$	a<x and x<b	$a \leq x \leq b$	a<=x and x<=b
$x < a$ *ou* $x > b$	x<a or x>b	$x \neq a$	x<>a

Exemple 23

```
> piecewise(1<x and x<2,x^2,x>2,abs(x));
```

$$\begin{cases} x^2 & 1 < x \text{ and } x < 2 \\ |x| & 2 < x \end{cases}$$

```
> f:=x->piecewise(1<x and x<2,x^2,x>2,abs(x));
```

$$f := x \rightarrow piecewise(1 < x \text{ and } x < 2, x^2, 2 < x, |x|)$$

```
> f(2);
```

$$0$$

```
> f(1.5);
```

$$2.25$$

1.3.11 Résolution d'équations :

Appel type : **solve**$(equ1, x)$;

solve$(\{equ1, equ2, ..., equr\}, \{x1, x2, ...xs\})$;

Le système Maple met un *solveur* à la disposition de l'utilisateur ,c'est à dire une instruction qui permet de résoudre (dans le plan complexe) des systèmes d'équations à plusieurs indeterminées, il s'agit de l'instruction **solve** qui s'utilise géneriquement sous la forme : **solve**$(\{equ1, equ2, ..., equr\}, \{x1, x2, ...xs\})$;

où $x_1, x_2, ...x_s$ désignent les variables par rapport auxquelles on cherche à résoudre les équations $equ1, eq2, ...equr$,il faut noter cepandant que Maple ne trouve pas en général toutes les solutions complexes du système à résoudre, en revanche il garantit que les valeurs trouvées sont bien des solutions du système considérée.

Exemple 24

```
> solve(x^2+3*x+1,x);
```

$$\frac{1}{2}\sqrt{5}-\frac{3}{2},-\frac{3}{2}-\frac{1}{2}\sqrt{5}$$

```
> solve(a*x^2+1,x);
```

$$-\frac{1}{\sqrt{-a}},\frac{1}{\sqrt{-a}}$$

```
> solve(x^2+2*x+1,x);
```

$$-1,-1$$

```
> solve(x^4-2*x^3-3*x^2-1,x);
```

$$RootOf\left(\_Z^4-2\_Z^3-3\_Z^2-1,index=1\right),RootOf\left(\_Z^4-2\_Z^3-3\_Z^2-1,index=2\right),$$

$$RootOf\left(\_Z^4-2\_Z^3-3\_Z^2-1,index=3\right),RootOf\left(\_Z^4-2\_Z^3-3\_Z^2-1,\right.$$

$$\left.index=4\right)$$

```
> evalf(%);
```

$$3.027099072,\ 0.07338492846+0.5253892595\,I,\ -1.173868929,\ 0.07338492846-0.5253892595\,I$$

```
> fsolve(x^4-2*x^3-3*x^2-1,x);
```

$$-1.173868929,\ 3.027099072$$

```
> solve(x^2+y^2=1,{x(t),y(t)});
```

$$\left\{x=\frac{1}{\sqrt{t^2+1}},y=\frac{t}{\sqrt{t^2+1}}\right\},\left\{x=-\frac{1}{\sqrt{t^2+1}},y=-\frac{t}{\sqrt{t^2+1}}\right\}$$

1.3.12 Formes inertes

Les formes inertes sans des commandes sans conséquences sur les calculs qui permettent d'afficher des constantes mathémtiques ou des ordres sans provoquer l'évaluation par le logiciel.

Exemple 25

```
> cos(pi/6)=cos(Pi/6);
```

$$\cos\left(\frac{1}{6}\pi\right)=\frac{1}{2}\sqrt{3}$$

```
> Limit(sin(x)/x,x=0)=limit(sin(x)/x,x=0);
```

$$\lim_{x\to 0}\frac{\sin(x)}{x}=1$$

```
> Diff(sin(cos(x)),x)=diff(sin(cos(x)),x);
```

$$\frac{d}{dx}\sin(\cos(x))=-\sin(x)\cos(\cos(x))$$

```
> Int(x^2+x+1,x)=int(x^2+x+1,x);
```

$$\int\left(x^2+x+1\right)dx=\frac{1}{3}x^3+\frac{1}{2}x^2+x$$

1.3.13 Graphe de fonction

La commande plot permet de dessiner des graphes sous Maple, la syntaxe de base est la suivante

plot(f(x),x=a..b)

Où f(x) est une expression qui dépend de x.

a..b les bornes de l'intervalle pour lequel on souhaite afficher le graphe.

Si l'intervalle a..b n'est pas spécifié alors il est considéré par défaut comme ayant les bornes -10..10

Le mot infinity permet d'utiliser des bornes infinies pour les graphes.

Le graphe est obtenue aux bornes infinies en composant automatiquement la fonction souhaité avec arctan et en traçant le graphe de la fonction composée aux bornes -1..1

Exemple 26

```
> plot(exp(x),x=0..10);
```