جامعة غليزان
RELIZANE UNIVERSITY

Introduction à Maple

premiers calculs et premières applications

Maple

Maple est un langage de manipulation symbolique qui permet grâce à une large bibliothèque de fonctions très bien adaptées, de réaliser aussi ben des calculs numériques que des calculs formels.

Je fournis un cour sous forme des exemple dont de nombreux exemples d'utilisation incorrecte des fonctions étudiées afin d'entraîner l'étudiant à décrypter et interpréter les messages d'erreurs qu'il pourra rencontrer lors de son utilisation ultérieure de Maple.

Ce cour s'adresse en priorité aux étudiants du premier cycle universitaire et aux étudiants ingénieurs et aux étudiants de troisième cycle.

L'ensemble est organisé en chapitres largement indépendants les uns des autres, chacun étant centré sur un thème bien précis. Dans chacun des chapitre, les exemples présentés sont volontairement simples, afin de permettre à l'étudiant de se concentrer sur les syntaxes utilisées.

Table des Matières

1. Introduction

1-1 Premiers pas
1-1-1 Entré d'une expression
1-1-2 Opérateurs, fonctions et constantes
1-1-3 Premiers calculs
1-2 Affectation et évaluation
1-2-1 Identificateurs
1-2-2 Affectation
1-2-3 Variables libres et évaluation
1-2-4 Règle d'évaluation complète
1-2-5 Utilisation des apostrophes : évaluation différée.
1-3 Opérations fondamentales
1-3-1 La fonction *expand*
1-3-2 La fonction *factor*
1-3-3 La fonction *combine*
1-3-4 La fonction *normal*
1-3-5 Utilisation de *convert* en trigonométrie
1-3-6 Première approche de la fonction *simplify*
1-3-7 Les fonctions *collect* et sort
1-4 Première approche de fonctions
1-4-1 Fonctions d'une variable
1-4-2 Fonctions de plusieurs variables
1-4-3 Ne pas confondre fonctions et expressions
1-4-4 Passerelles entre expressions et fonctions

2. Arithmétique

2-1 Divisibilité
2-1-1 Quotient et reste
2-1-2 P.G.C.D et Algorithme d'Euclide
2-1-3 Décomposition en facteurs premiers
2-1-4 Congruences
2-2 Résolution d'équations modulo n

3. Nombres Réels, Nombres Complexes

3-1 Les nombres réels
3-2 Les nombres complexes
3-2-1 Les différents types de complexes
3-2-2 Forme algébrique des complexes
3-2-3 Forme trigonométrique des complexes

4. Programmation : branchements, boucles et procédures

4-1 Branchements
4-2 Boucles
4-3 Procédures

5. Graphiques en 2D

5-1 Courbes y=f(x)
5-1-1 Représentation graphique d'une expression
5-1-2 Représentation graphique d'une fonction
5-1-3 Tracé simultané de plusieurs courbes
5-1-4 Tracé d'une famille de courbes
5-2 L'environnement de plot
5-2-1 Le menu de plot sous Windows
5-2-2 Les options de plot
5-3 Courbes paramétrées en cartésiennes
5-3-1 Tracé d'une courbe paramétrée
5-3-2 Tracé simultané de plusieurs courbes paramétrés
5-3-3 Tracé d'une famille de courbes paramétrées
5-4 Courbes en polaires
5-4-1 Tracé d'une courbe en polaire
5-4-2 Tracé d'une famille de courbes en coordonnées polaires
5-5 Courbes définies implicitement
5-5-1 Tracé d'une courbe définie implicitement
5-5-2 Tracé simultané de plusieurs courbes implicites
5-5-3 Précision du tracé des courbes implicites
5-6 Traces polygonales
5-6-1 Tracé de lignes polygonales
5-6-2 Tracé de polygones

6. Equations et Inéquations

6-1 Résolution Exacte
6-1-1 Equations polynomiales à une variable
6-1-2 Autres équations à une variable
6-1-3 Systèmes d'équations
6-1-4 Résolution de certaines inéquations
6-2 Résolution approchée d'équations : fsolve
6-2-1 Equation algébrique à une variable
6-2-2 Autres équations à une variable
6-2-3 Systèmes d'équations

7. Limites, Dérivées

7-1 Limites
7-1-1 Limite d'expressions
7-1-2 Limite de fonctions
7-1-3 Limite d'expression dépendant de paramètres
7-1-4 Limites des fonctions de plusieurs variables
7-2 Dérivées
7-2-1 Dérivée des expressions d'une seule variable
7-2-2 Dérivées partielles des expressions de plusieurs variables
7-2-3 Dérivées des fonctions d'une variable
7-2-4 Dérivées partielles des fonctions de plusieurs variables

8. Développement limités

8-1 La fonction *series*
8-1-1 Obtention de développements limités
8-1-2 Développement généralisé
8-1-3 Partie régulière d'un développement
8-1-4 Obtention d'un équivalent
8-1-5 Les limites de la fonction *series*
8-2 Opération sur les développements limités
8-2-1 Sommes, quotients, produits de développements limités
8-2-2 Composés et inverses de développements limités
8-2-3 Intégration d'un développement limité
8-3 Développement d'une fonction implicite
8-3-1 Evaluation de RootOf(p,y,b)
8-3-2 Développement limité d'une fonction implicite

9. Intégration et Sommation

9-1 Intégration
9-1-1 Calcul exact d'intégrales et de primitives
9-1-2 Intégrales généralisées
9-1-3 Forme inerte Int
9-1-4 Evaluation numérique d'intégrales
9-2 Opérations sur les formes non évaluées
9-2-1 Intégration par paries
9-2-2 Changement de variable dans un intégrale
9-2-3 Dérivation sous le signe Intégral
9-2-4 Développement limité d'une primitive
9-3 Sommation discrète
9-3-1 Sommes indéfinies
9-3-2 Sommes finies

10. Algèbre linéaire

10-1 Création de vecteurs et de matrices
10-1-1 La bibliothèque *linalg*
10-1-2 Vecteurs
10-1-3 Matrices
10-2 Calcul vectoriel et matriciel
10-2-1 Opérations sur vecteurs et matrices
10-2-1-1 Combinaison linéaire de vecteurs
10-2-1-2 Combinaison linéaire de matrices
10-2-1-3 Transposition de matrice
10-2-1-4 Produit d'une matrice par un vecteur
10-2-1-5 Produit de matrice
10-2-1-6 Inverse d'une matrice
10-2-1-7 Puissance de matrices carrées
10-3 Système d'Equations linéaires
10-3-1 Système linéaire donné matricielle ment
10-3-2 Système linéaire donné par des équations
10-3-3 Utilisation de *linsove*

10-4 Réduction des Matrices
10-4-1 Déterminant, Polynôme caractéristique
10-4-2 Elements propres d'une matrice

MAPLE

Maple : est un Logiciel de Calcul formel
Maple : est un Langage de manipulation "sympolique"
qui permet grâce à une Large "bibliothèque" de
fonctions trés bien adaptés.

Introduction

Au démarrage d'une session, Maple affiche un prompt (le symbole >)
L'utilisateur peut lui demander d'exécuter "quelque chose".
Pour cela il doit entrer au clavier une assertion Maple.

1 Premiers pas :

1.1 Entré d'une expression :

Exemple 1.1

```
> 1+1;  # ne pas oublier de taper <Entrée>
          2
> 2^10;
          1024
> 1+2*3+4;
          11
```

Exemple 1.2

```
> 1+2*+4;    # si on oublie par exemple de saisir 3
        syntax error :
    1+2*+4;
       ^
```

Exemple 1.3

```
> 1+1
              # oubli du point-virgule Maple attend
> ;
              # le point virgule permet de conclure la ligne précédente
      2
```

Remarque : Dés que Maple rencontre le symbole # sur une ligne,
il ignore tous les caractéres qui le suivent

1-2 Opérateurs, fonctions et constantes

* Opérateurs élémentaires :

Opérateur	Notation	Exemple
Factorielle	!	7!
Puissance	^^ ou ^	2.5^^3
Quotient de division (N)	iquo(·,·)	iquo (17,3)
Reste de division entière	irem(·,·)	irem (17,3)

* Fonctions élémentaires "Maple"

exp(x), log(x) ou ln(x), log10(x), abs(x), sqrt(x)

sin(x), cos(x), tan(x), cot(x), sinh(x), cosh(x), tanh(x), cotanh(x)

arcsin(x), arccos(x), arctan(x), arccotan(x) ; arcsinh(x), arccosh(x), ...

* Constantes : Maple

Identificateur	Description
Pi	3.14159
E	la base des logarithmes néperiens
I	racine de -1
infinity	+ l'infini
gamma	La constante d'Euler

1-3 Premiers calculs :

* Evaluer en flottant . c.à.d en virgule flottante

Exemple 1.3.1

```
> 10/3 ;
                    10/3
> evalf(10/3) ;
                  3.333333
```

Exemple 1.3.2

```
> Digits := 20; # Digits : determine le nombre de chiffres significatifs
                # attention à la majuscule !
                Digits := 20
> evalf(10/3);
                3.3333333333333333333
> 3 + sqrt(2);
                3 + √2
> evalf(") ;      # le ditto " evaluer la dernière expression
                4.4142135623730950488
```

Exemple 1.3.3

```
> ln(2) ; evalf(",17) ; # Evaluer la dernière expression avec 17
                        # chiffres significatifs
```

$\ln(2)$

0.69 314 718 055 99 45 31

Exemple 1.3.4

```
> Digits := 15 ;
```

Digits := 15

```
> evalf(Pi) ;
```

3.14 159 26 53 58 979

```
> evalf(E) ;
```

2.718 28 18 28 45 905

Exemple 1.3.5

```
> sin(Pi/2) ;
```

1

```
> sin(pi/2) ;
```

$\sin(\frac{1}{2}\pi)$

```
> sin(Pi/8) ;
```

$\frac{1}{2}\sqrt{2-\sqrt{2}}$ # valeur stockée dans une table

```
> sin(alpha/2) ;
```

$\sin(\frac{1}{2}\alpha)$

2 Affectation et Evaluation

2.1 Identificateurs :

Maple : travaille sur des objets tels que

- des valeurs numériques réelles, complexes
- des fonctions, des procédures, ...
- des ensembles, des listes ou des tableaux ...

2.2 Affectation :

Exemple 2.2.1

```
> x1 := 3-2 ;   # x1 : est un identificateur ; 3-2 : expression affectée
```

x1 := 1

Exemple 2.2.2

```
> x1 ;
```

1

```
> 2*x1 ;
```

2

```
> y := x1+3 ;
```

y := 4

```
> y ;
```

4

2.3 Variables libres et évaluation

Exemple 2.3.1

```
> restart ;      # pour ré-initialiser le contexte et donc liberer les
                 # variables ;
> P:= x^2 +x+1 ;
```

$$P := x^2 + x + 1$$

Exemple 2.3.2

```
> x := 1;
```

$$x := 1$$

```
> P;
```

$$3$$

```
> x:= 2 ;
```

$$x := 2$$

```
> P;
```

$$7$$

Exemple 2.3.3

```
> x:='x';   # transformer seulement x en variable libre
```

$$x := x$$

```
> P ;
```

$$x^2 + x + 1$$

2.4 Régle d'évaluation complète

Exemple 2.4.1

```
> restart ;      # pour ré-initialiser le contexte
> P:= x^2 +x+1;
```

$$P := x^2 + x + 1$$

```
> x:= y^2;
```

$$x := y^2$$

```
> P;
```

$$y^4 + y^2 + 1$$

```
> y:=2 ; P;      # on peut mettre plusieurs instructions sur
                 # une même ligne
```

$$y := 2$$

$$21$$

Exemple 2.4.2

```
> t:=1; t:=t+1; t;   # la régle d'évaluation complète impose de prendre garde
                     # aux références circulaires
```

$$t := 1$$

$$t := 2$$

$$2$$

```
> t:='t'; t:=t+1;    #    se contente d'abord d'un avertissement : Waring
```

$$t := t$$

```
Waring, recursive definition of name
> t ;                # mais si on demande une évaluation de t,
Error, STACK OVERFLOW                # on sature la pile
```

2.5 Utilisation des apostrophes : évaluation différée

Exemple 2.5.1

```
> x:=1 ; Φ:='x^3+1'; # les apostrophes empêchent l'évalution
                     # totale du membre de droite de la dernière affectation
                  x:=1
                  Φ:=x^3+1
```

Exemple 2.5.2

```
> x:=2 ; Φ ;   # x:=2 est suivi de : pour éviter un écho inutile
                  9
> x:=3 : Φ ;
                  28
```

Exemple 2.5.3

```
> restart ;          # Pour ré-initialiser le contexte et donc "liberer" les variables.
> P:=x^2+x+1 ;
                  P:=x^2+x+1
> x:=y^2 ;
                  x:=y^2
> y:=1 ;
                  y:=1
> eval(P,1) ;        # eval(Expr,n) retourne l'évaluation de Expr
                  x^2+x+1   #        au niveau n
> eval(P,2) ;
                  y^4+y^2+1
> eval(P,3) ;
                  3
```

3 Opérations Fondamentales

Exemple 3.1

```
> restart; ((x-y)*(x^2+x*y+y^2))/(x-y); # simplifie
                  x^2+xy+y^2
> (x^3-y^3)/(x-y);   # il ne simplifie pas qq expressions
                  (x^3-y^3)/(x-y)
> sin(x)^2+cos(x)^2 ;
                  sin(x)^2+cos(x)^2
```

Remarque : Maple garde aux expressions une forme semblable à la forme initialement rencontrée mais il simplifie qq expressions.

3.1 La fonction expand " développer "

Exemple 3.1.1

```
> restart ; expand( ( x+1)*(x+2)^2);
```

$$x^3 + 5x^2 + 8x + 4$$

```
>expand((cos(x) + sin(x))^2);        # Expression "polynomiale"
```

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

```
>expand( ln(x/y^2));
```

$$\ln(x) - 2\ln(y)$$

```
>expand ( cos(3*x));
```

$$4\cos(x)^3 - 3\cos(x)$$

```
>expand ( tan (3*x));
```

$$\frac{4\sin(x)\cos(x)^2 - \sin(x)}{4\cos(x)^3 - 3\cos(x)}$$

3.2 La fonction factor " factoriser "

Exemple 3.2.1

```
> factor (x^8 - 1);
```

$$(x-1)(x+1)(1+x^2)(1+x^4)$$

```
> factor ((x^2 - y^2));   # factoriser une expression poly d'une ou + variables
```

$$(x-y)(x+y)$$

```
> factor ( sin(x)^3 - cos(x)^3);   # Expression " polynomiale "
```

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

* Dans certains cas, Maplene réagit pas devant des factorisation immédiate

Exemple 3.2.2

```
> f:= x^2 - x*(exp(t) + exp(-t)) + 1;
```

$$f := x^2 - x(e^t + e^{-t}) + 1$$

```
> factor(f);
```

$$x^2 - xe^t * xe^{(-t)} + 1$$

* En revanche, on applique expand avant factor

Exemple 3.2.3

```
> g:= expand(f);
```

$$g := x^2 - xe^t - \frac{x}{e^t} + 1$$

```
> factor(g);
```

$$\frac{(xe^t - 1)(x - e^t)}{e^t}$$

3.3 La fonction combine "transformer"

Paramètre	Transformation exécutée
trig	linéarisation des exp² trig (circulaires ou hyp)
exp	$e^x * e^y \to e^{x+y}$, $(e^x)^a \to e^{x \cdot a}$; $e^{x+n\ln(y)} \to e^x \cdot y^n$
ln	$\ln(x) + \ln(y) \to \ln(x \cdot y)$; $a * \ln(y) \to \ln(y^a)$
power	$z^x \cdot z^y \to z^{x+y}$; $(z^x)^a \to z^{x*a}$; $\sqrt{-a} \to i\sqrt{a}$

Exemple 3.3.1

```
> combine(sin(a) * cos(b), trig);
```
$\frac{1}{2}\sin(a+b) - \frac{1}{2}\sin(-a+b)$

```
> combine(x^a * x^b + exp(x) * exp(y), exp);
```
$x^a \cdot x^b + e^{(x+y)}$

```
> combine(x^a * x^b + exp(x) * exp(y) + sin(x)^4, power);
```
$x^{(a+b)} + e^{(x+y)} + \sin(x)^4$

```
> combine(x^a * x^b + exp(x) * exp(y) + sin(x)^4, {trig, exp});
```
$x^a \cdot x^b + e^{(x+y)} + \frac{3}{8} + \frac{1}{8}\cos(4x) - \frac{1}{2}\cos(2x)$

3.4. La fonction normal "reduire"

Exemple 3.4.1

```
> Fr := (x^4 - y^4)/(x^2 - y^2);
```
$Fr := \frac{x^4 - y^4}{x^2 - y^2}$

```
> normal(Fr);
```
$x^2 + y^2$

```
> Fr := (sin(x)^3 - cos(x)^3)/(sin(x)^2 - cos(x)^2);
```
$\frac{\sin(x)^3 - \cos(x)^3}{\sin(x)^2 - \cos(x)^2}$

```
> normal(Fr);
```
$\frac{\sin(x)^2 + \sin(x)\cos(x) + \cos(x)^2}{\sin(x) + \cos(x)}$

Exemple 3.4.2

```
> f := (x*x - 2)/(x - sqrt(2));
```
$f := \frac{x^2 - 2}{x - \sqrt{2}}$

```
> normal(f);
```
$\frac{x^2 - 2}{x - \sqrt{2}}$

Difference entre le factor et normal

```
> factor(f);
```
$x + \sqrt{2}$

normal simplifie que si les "coeff" du numérateur
et du dénominateur sont rationnels

3.5 Utilisation de Convert en trigonométrie "Exprimer"

Option	Transformation effectuée
exp	exprime ttes les lignes trig hyperbolique et circulaire à l'aide d'exponentielles réelles ou complexes.
ln	exprime les fonctions réciproques des fonctions trigo à l'aide des logorithmes.
expln	combinaison des deux précédentes
exp sincos	exprime les lignes trigo circulaires à l'aide de sin et cos et les hyperboliques à l'aide de exp
tan	exprime les lignes trigo circulaires en fct de la tangente de l'arc moitié
trig	exprime les exp en sin, cos, sinh, cosh à l'aide des fomule d'Euler

Exemple 3.5.1

```
> restart;
> convert ( tanh(a) + tan(b) + exp(c) + exp(I*d) + sinh(e) + sin(f), trig );
```

$$\tanh(a) + \tan(b) + \cosh(e) + \sinh(c) + \cos(d) + I\sin(d) + \sinh(e) + \sin(f)$$

```
> convert(tanh(a) + tan(b) + exp(c) + exp(I*d), sincos);
```

$$\frac{\sinh(a)}{\cosh(a)} + \frac{\sin(b)}{\cos(b)} + e^{c} + e^{(Id)}$$

```
> convert(1 + tan(x)^2, sincos);
```

$$1 + \frac{\sin(x)^2}{\cos(x)^2}$$

Exemple 3.5.2

```
> convert (arcsin(x), ln);
```

$$\ln\left(x + \sqrt{x^2+1}\right)$$

Exemple 3.5.3

```
> f:= 1/(2 + cos(x));
```

$$f := \frac{1}{2 + \cos(x)}$$

```
> g:= convert (f, tan);
```

$$g := \frac{1}{2 + \frac{1 - \tan\left(\frac{1}{2}x\right)^2}{1 + \tan\left(\frac{1}{2}x\right)^2}}$$

```
> g:= normal (g);
```

$$g := \frac{1 + \tan\left(\frac{1}{2}x\right)^2}{\quad}$$

3.6 Première approche de la fonction simplify "simplifier"

* Première utilisation : simplify(..., symbolic) réalise tte tf du tableau :

Option	tranformation effectuée
power	$(a^b)^c \to a^{bc}$; $\exp(5 \times \ln(x)+1) \to x^5 \cdot \exp(1)$; $\ln(x*y) \to \ln(x) + \ln y$
radical	$8^{1/3} \to 2$, $(x^3 + 3x^2 + 3x + 1)^{1/3} \to (x+1)$
sqrt	$\sqrt{4} \to 2$; $\sqrt{x^2+2x+1} \to (x+1)$
trig	$1 + \tan(x)^2 \to 1/\cos(x)^2$; $\sin(x)^2 \to 1 - \cos x^2$; travail analogue pour les hyperbo

Exemple 3.6.1

```
> f:=(x^a)^b+(u^2+2*u+1)^(1/2)+8^(1/3);
```

$$f := (x^a)^b + \sqrt{u^2+2u+1} + 8^{1/3}$$

```
> g:=simplify(f, symbolic);        # ttes options
```

$$g := x^{(ab)} + u + 3$$

```
> h:=simplify(f, sqrt,power, symbolic);
```

(symbolic barré)

$$h := x^{(ab)} + u + 1 + 8^{1/3}$$

* La fonction simplify sans l'option symbolic

Exemple 3.6.2

```
> (1+1/2*cos(2*x))^2+sin(x)^2;
```

$$\left(1 + \frac{1}{2}\cos(2x)\right)^2 + \sin(x)^2$$

```
> simplify(");
```

$$\cos(x)^4 + 5/4$$

* Utilisation de régles particulières de simplifications.

Exemple 3.6.3

```
> f:= sin(x)^2-sin(x)^2*cos(x)^2 ;
```

$$f := \sin(x)^2 - \sin(x)^2 \cdot \cos(x)^2$$

```
> simplify(f);
```

$$\cos(x)^4 - 2\cos(x)^2 + 1$$

```
> simplify(f,[cos(x)^2 = 1-sin(x)^2]);
```

$$\sin(x)^4$$

Exemple 3.6.4

```
> restart: f:=(a^3+3*a*b^2)/(3*a^2*b+b^3);
```

$$f := \frac{a^3 + 3ab^2}{3a^2b + b^3}$$

```
> simplify(f,[a/b =t]);
```

error, (in grobner/gbasis) input must be polynomials over, [a,b,t]

```
> a:=b*t : simplify(f);      # en revanche
```

$$\frac{t(t^2+3)}{3t^2+1}$$

3.7 Les fonctions collect et sort

* La fonction collect "regrouper"

Exemple 3.7.1

```
> restart: f:=(x-a)*(x-b)*(x-c); f:=expand(f);
```

$$f := x^3 - x^2c - x^2b + xbc - ax^2 + axc + abx - abc$$

```
> collect(f,x);    # regrouper selon les puissances de x
```

$$x^3 + (-a-b-c)x^2 + (ac+ab+bc)x - abc$$

```
> collect(f,[a,b]); # regrouper selon les puissances de a et b
```

$$((x-c)b - x^2 + xc)a + (-x^2+xc)b + x^3 - x^2c$$

```
> collect(f,[a,b],distributed);    # avec distribution
```

$$x^3 - x^2c + (x-c)ab + (-x^2+xc)a + (-x^2+xc)b$$

exemple 3.7.2

```
> Expr:=expand(sin(3*x)+cos(3*x));
```

$$Expr := 4\sin(x)\cos(x)^2 - \sin(x) + 4\cos(x)^3 - 3\cos(x)$$

```
> collect(Expr,sin(x));
```

$$(4\cos(x)^2 - 1)\sin(x) + 4\cos(x)^3 - 3\cos(x)$$

```
> collect(Expr,sin(x),factor); # regrouper les sin(x) sous forme factorisée
```

$$(2\cos(x)-1)(2\cos(x)+1)\sin(x) + \cos(x)(4\cos(x)^2 - 3)$$

x La fonction sort "ordoner"

Exemple 3.7.3

```
> expand((x^2+x+1)^5);
```

$$1 + 15x^2 + 45x^4 + 30x^3 + 45x^6 + 51x^5 + 15x^8 + 5x + 30x^7 + x^{10} + 5x^9$$

```
> sort(");
```

$$x^{10} + 5x^9 + 15x^8 + 30x^7 + 45x^6 + 51x^5 + 45x^4 + 30x^3 + 15x^2 + 5x + 1$$

```
> u:=expand((x^2+sin(y)+z)^3);
```

$$u := x^6 + 3\sin(y)x^4 + 3x^4z + 3\sin(y)^2x^2 + 6\sin(y)x^2z + 3x^2z^2 + \sin(y)^3 + 3\sin(y)^2z + 3\sin(y)z^2 + z^3$$

```
> sort(u,sin(y));
```

$$\sin(y)^3 + 3x^2\sin(y)^2 + 3z\sin(y)^2 + 6zx^2\sin(y) + 3x^4\sin(y) + 3z^2\sin(y) + 3z^2x^2 + 3zx^4 + x^6 + z^3$$

```
> sort(u,[x,z]);
```

$$x^6 + 3zx^4 + 3\sin(y)x^4 + 3x^2z^2 + 6\sin(y)x^2z + z^3 + 3\sin(y)^2x^2 + 3\sin(y)z^2 + 3\sin(y)^2z + \sin(y)^3$$

4 Premiere approche de fonctions

4.1 Fonctions d'une variable

Exemple 4.1.1

```
> Ma_Fonct := x -> x^2 + x + 1 ;   # -> opérateur arrow
```

$Ma\_Fonct := x \to x^2 + x + 1$

```
> Ma_Fonct(2);
```

7

```
> Ma_Fonct(sqrt(2));
```

$3 + \sqrt{2}$

```
> Ma_Fonct(t);
```

$t^2 + t + 1$

4.2 Fonctions de plusieurs variables

Exemple 4.2.1

```
> f_1 := (x,y) -> (x^3 - y^3)/(x-y) ;
```

$f\_1 := (x,y) \to \frac{x^3 - y^3}{x - y}$

```
> f_1(1,a);
```

$\frac{1-a^3}{1-a}$

4.3. Ne pas confondre fonction et expression

Exemple 4.3.1

```
> x := 1 ;
```

$x := 1$

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

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

```
> P := x^2 + x + 1 ;
```

$P := 3$

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

7

4.4. Passerelles entre expressions et fonctions

Exemple 4.4.1

```
> Φ := 'a*x*x + 1' ;   # On met des apostrophes par sécurité
```

$\Phi := ax^2 + 1$

```
> g := unapply(Φ, x) ;   # à la x -> Φ
```

$g := x \to ax^2 + 1$

Exemple 4.4.2

```
> restart; R := x^2 + x*y + y^2 ;
```

$R := x^2 + xy + y^2$

```
> h := unapply(R, x, y) ;
```

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

ARITHMETIQUE

2.1 Divisibilité

2.1.1 Quotient et reste

iquo(a,b) renvoie le quotient de la division entière de a par b

irem(a,b) renvoie le reste de la division entière de a par b

Exemple 1.1.1

```
> restart; iquo(24,7);
                                3
> irem(12,7);
                                5
```

Exemple 1.1.2

```
> iquo(24,9,r);
                                2
> r;
                                6
```

Exemple 1.1.3

```
> iquo(12,7,r);
        Error, wrong number (or type) of parameters in function iquo
```

Exemple 1.1.4

```
> r:=`r`: iquo(24,9,r): r;
                                6
> iquo(12,7,'r'): r;
                                5
```

2.1.2. P.g.c.d et algorithme d'Euclide

* Les fonctions igcd et ilcm

Exemple 1.2.1

```
> igcd(12,16);                  # Pgcd(12,16)
                                4
> ilcm(12,16);                  # pp.cm(12,16)
                                48
> ilcm(12,16,15);
                                240
```

* Algorithme d'Euclide étendu : La fonction igcdex

Exemple 1.2.2

```
> u:=`u`: v:=`v`: d:= igcdex(12,7,u,v);   # d= 12*u + 7*v
                                          d:= 1
> u; v;
                                3
                                -5
> 12*u+7*v-d;
                                0
```

2.1.3 Décomposition en facteurs premiers

Exemple 1.3.1

```
> isprime(561);   # retourne true (vrai) si n est premier et false si non
                  false
> ithprime(17);   # retourne le n^eme nombre premier
                  59
> prevprime(1007), nextprime(1007);   # le nb premier strictement supé à
                                      # " " " " " inférieur à 1007
                  994 , 1009
> ifactor(561);   # retourne la décomposition de n en facteurs premiers
                  (3)(11)(17)
> ifactor(2727);
                  (3)^3 (101)
```

2.1.4 Congruences

Exemple 1.4.1

```
> 14 mod 4;          # Congru à a modulo n
                  2
> 1/2 mod 17;        # retourne l'unique entier c ∈ [0,17[ tq 2xC congru à 1 modulo 17
                  9
> 4/3 mod 14;
                  6
> 1/2 mod 14;        # 2 et 14 ne sont pas premiers entre eux
Error, the modular inverse does not exist
```

2.2 Résolution d'équations modulo n

Exemple 2.1

```
> restart; sol := msolve(x^4 = 1, 33);   # msolve permet de resoudre des égs
                                           modulo n = 33
sol := {x=23}, {x=10}, {x=32}, {x=1}
```

Exemple 2.2

```
> Sol := msolve(x^2 = -1, 33);   # l'éq n'a aucune solution
                  sol :=
```

Exemple 2.3

```
> msolve(x^561 = x, 561);   # l'éq est vérifiée pour tte valeur de x
                  {x = x}
```

Exemple 2.4

```
> msolve({2*x + 3*y = 3, 4*x + y = 5}, 13);
                  {y=8, x=9}
```

* **Exemple 2.5**

```
> isolve(x*x - y*y = 9);   # Pour résoudre une éq en nb entiers
{x=5, y=-4}, {x=-5, y=-4}, {y=0, x=3}, {y=0, x=-3}, {x=-5, y=4}, {x=5, y=4}
```

Nombres Reels Nombres Complexes

3.1. Les nombres Reels

Exemple 3.1.1

```
> a:=2^10+1;
                    a:= 1025
> whattype(a);                 # wattype retourne le type d'une expression
            integer
```

Exemple 3.1.2

```
> a:=15*(-5^(-2))+3;
                         a:= 12/5
> whattype(a);
              fraction
```

Exemple 3.1.3

```
> a:=10. ; wattype(a);   # avec point décimal, a est de type float
                a:= 10.
                      float
```

Exemple 3.1.4

```
> Float(6023,20);      .6023 10^24    # désigne le nombre décimal a x10^b
> 10000.^10000 ;
             Float(1000 000 000, 39997)  # pour écrire le nb a x10^b
```

3.2. Les Nombres Complexes

3.2.1 Les différents types de complexes

Exemple 3.2.1

```
> restart; (1+2*I)^3;
                     -11 - 2I
> (1/2 + I/3)^3;
                 - 1/24 + 33/108 I
> (0.5 + I/3)^3;
              -0.4166666663 + .2129629629 I
```

Exemple 3.2.2

```
> (1+sqrt(2)*I)^3;              # n'est pas automatiquement mise sous forme a+Ib
                  (1+√2 I)^3
```

Exemple 3.2.3

```
> z:=(1+I*sqrt(2))^3 ; z1:=expand(z);
                             z:=(1 + I√2)^3
                                   z1:=-5 + I√2
> whattype(z); wattype(z1);
                         ^
                         +
> type(z, complex(realcons)); type(z1, complex(realcons));
             false    true       # z est écrit sous la forme a+Ib
```

3.2.2. Forme algébrique des complexes

Exemple 3.2.2.1

```
> z:=(1+I)/(1-2*I);
> u:=(sqrt(2)+I)*(1+I);
> evalc(u);          # la fonction evalc
> v:=(sqrt(2)+I)/(1+sqrt(2)*I);
> evalc(v);
```

$z := -\frac{1}{5} + \frac{3}{5} I$

$u := (1+I)(\sqrt{2}+I)$

$\sqrt{2} - 1 + I(1+\sqrt{2})$

$v := \frac{\sqrt{2}+I}{1+I\sqrt{2}}$

$\frac{2}{3}\sqrt{2} - \frac{1}{3} I$

Exemple 3.2.2.1

```
> z:=(1+I)(1+I*sqrt(2));
> a:=Re(z);
> b:=Im(z);
```

$z := (1+I)(1+I\sqrt{2})$

$a := 1 - \sqrt{2}$

$b := 1 + \sqrt{2}$

Exemple 3.2.2.2

```
> z:=(1+I)/(1-sqrt(2)*I);
> a:=Re(z), b:=Im(z);
> evalc(a), evalc(b);
```

$z := \frac{1+I}{1-I\sqrt{2}}$

$a := \Re\left(\frac{1+I}{1-I\sqrt{2}}\right)$, $b := \Im\left(\frac{1+I}{1-I\sqrt{2}}\right)$

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

Exemple 3.2.2.3

```
> z1:=1+2*I, z2:=(1+I)/(1-sqrt(2)*I);
> conjugate(z1), conjugate(z2);
```

$z1 := 1+2I$, $z2 := \frac{1+I}{1-I\sqrt{2}}$

$1-2I$, $\frac{1-I}{1+I\sqrt{2}}$

3.2.3. Forme trigonométrique des complexes / Calcul sur des expressions à coeff complexes

Exemple 3.2.3.1

```
> z:=2+3*I, abs(z);
> argument(1+I);
> argument(-3-2*I);
```

$z := 2+3I$, $\sqrt{13}$

$\frac{1}{4}\pi$

$\arctan\left(\frac{2}{3}\right) - \pi$

Exemple 3.2.3.2

```
> restart; z:=(a+I)^3, evalc(z);
> z:=sin(x+I*y): evalc(z);
> z:=a+I: abs(z), evalc(abs(z));
```

$z := (a+I)^3$, $a^3 - 3a + I(3a^2 - 1)$

$\sin(x)\cosh(y) + I\cos(x)\sinh(y)$

$|a+I|$, $\sqrt{a^2+1}$

Programmation : branchements, boucles, et Procédures.

4-1 Branchements :

Le branchement conditionnel : if ... then ... elif ... else ... fi

cond — suite d'instuction

Exemple 4.1.1 :

```
> if a=0 then if b=0 then print ('Tout_nombre_est_solution');
  else print ('pas_de_solution') fi;
  else print ('Une_solution_unique_:' - b/a)
  fi;
```

Exemple 4.1.2

```
> if a=0 and b=0 then print (Tout_nombre_est_solution);
  elif a=0 then print ('pas_de_solution)
  else print ('Une_solution_unique:' - b/a)
  fi;
```

4.2. Boucles :

Boucle for avec compteur numérique :

for compteur from b to fin by pas do <suite_d'instruction> od :

Exemple 4.2.1

```
> for i from 0 to 5 by 2 do print (i^3); od;
```

0
8
64

Exemple 4.2.2

```
> restart; for i from 1 to 3
> do
> expand (cos (i*x));
> subs (cos(x)=t,'');
> od;
```

$\cos(x)$
t
$2\cos(x)^2-1$
$2t^2-1$
$4\cos(x)^3-3\cos(x)$
$4t^3-3t$

4.3 Procedures :

Definition et appel d'une procédure :

proc(var1, ..., varp)
Local lvar1, ..., lvarq global gvar1, ..., gvar.r
option <seq_options>; <instruction1>; ... <instruction_n>;
end:

Exemple 4.3_1

```
> fact:=poc(n) if n=0 then 1 else n*fact (n-1); fi; end;

  fact:=proc (n) if n=0 then 1 else n*fact(n-1) fi end
```

Exemple 4.3.2

```
> fact1:=proc(n) local i; y; y:=1; for i to n do y:=i*y; od;
                                                        end:
```

Exemple 4.3.3

```
> fact2:=proc(n) local y; global i; y:=1; for i to n do y:=i*y; od;
                                                        end:
```

Exemple 4.3.4

```
> i:='i' : fac_1(5); i;
```

120
i

```
> fact_2(5); i;
```

120
6

Graphiques en 2D

I. Courbes $y = F(x)$

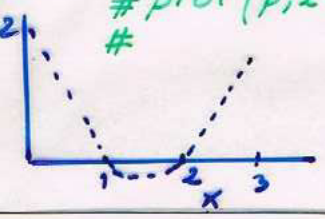
1.1 Représentation graphique d'une expression

Exemple 1.1

```
> p:= x^2 - 3*x + 2;
> plot(p, x=0..3);
#
```

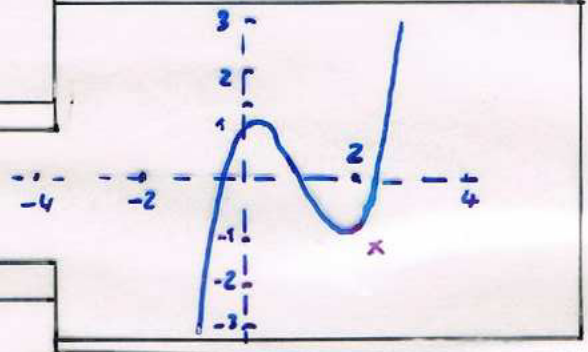
$p := x^2 - 3x + 2$

plot(p, x=a..b) # ouvre une fenêtre graphique et y dessine p pour $x \in [a,b]$



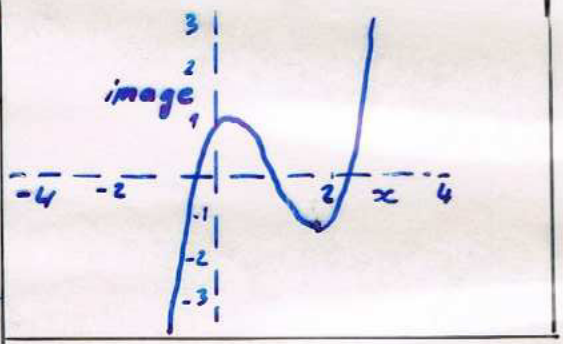
Exemple 1.2

```
> p:= x^3 - 3*x^2 + x + 1 :
> plot(p, x=-5..5, -3..3);
```



Exemple 1.3

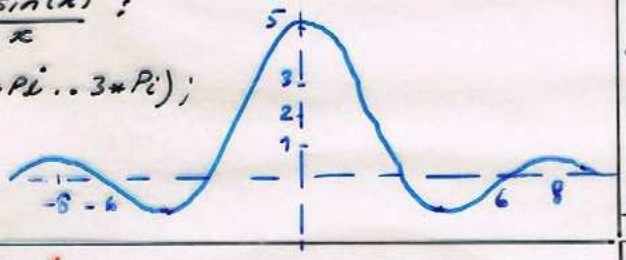
```
> plot(p, x=-5..5, image=-3..3);
```



1.2 Représentation graphique d'une fonction

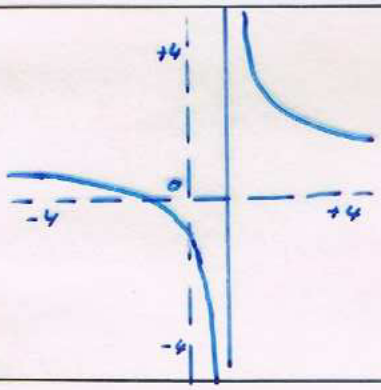
Exemple 1.2.1

```
> f:= x -> 5*sin(x)/x :
> plot(f, -3*Pi..3*Pi);
```



Exemple 1.2.2

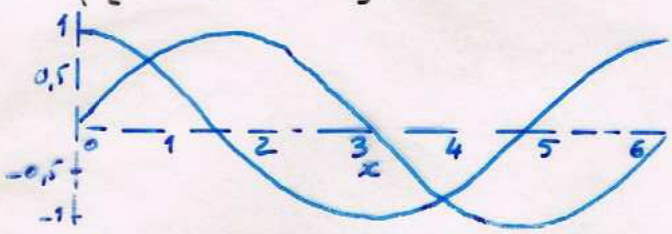
```
> f:= x -> (x+1)/(x-1) :
> plot(f, -5..5, -5..5);
```



1.3 Tracé simultané de plusieurs courbes.

Exemple 1.3.1

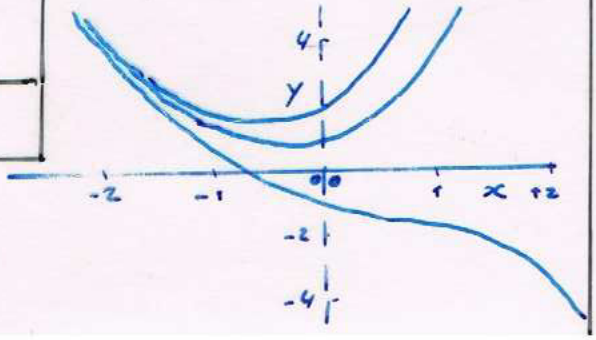
```
> plot({sin(x), cos(x)}, x=0..2*Pi, -1..1);
```



1.4 Tracé d'une famille de courbes.

Exemple 1.4.1

```
> p:= t -> t*exp(x) + x^2 : # (f_t)_{t∈ℝ}
> plot({p(-1), p(1), p(2)}, x=-3..3, y=-5..5);
```



II L'Environnement de Plot

2.1 le menu de plot sous Windows

la fenêtre graphique, peut être redimensionnée avec la souris:

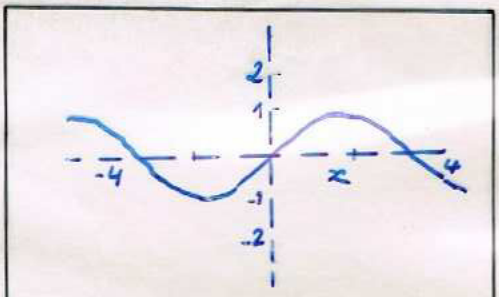
en haut { # la barre de menu : File, Edit, Style, Axes, projection (repère ⊥ ou non)
(imprimer, recopier, pt. ligne, position)
la barre d'outils : activer directement certaines options du menu précédent

en bas { # la barre d'état : permet l'affichage des coord du curseur de la souris.

2.2 Les options de plot

Ces options doivent être mises après les deux ou trois arguments déjà rencontrés.

- Options de présentation correspondant à certains choix du menu
- style = POINT ou style = LINE
- linestyle = n (n=1 : ——, n=2 - - , n=3 ··· , n=4 —·—)
- thickness = n : l'épaisseur de trait désirée
- axes = BOXED , axes = FRAMED (le compte pas) , axes = NORMAL , axes = NONE
- scaling = CONSTRAINED ou scaling = UNCONSTRAINED
- Options de présentation ne correspondant à aucun choix du menu
  - xtickmarks = n : affiche n valeurs / ox
  - ytickmarks = n : affiche n valeurs / oy
  - title = 'Titre du graphe' : touche ALTGR-7
  - color = n ou , color = black (en noir)
  - Labels = [NomX, NomY]

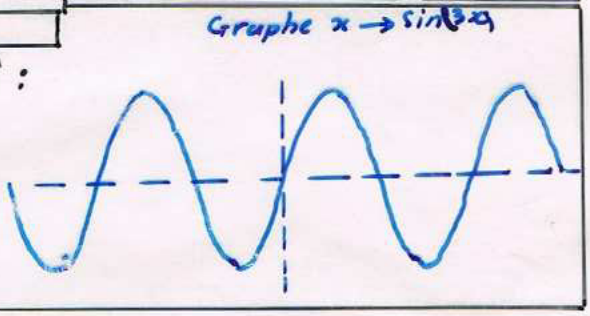


Exemple . 2 . 1

```
> plot ( sin(x) , x = -5..5, scaling = constrained , ytickmarks = 2) ;
```

Exemple . 2 . 2

```
> n:= 3 : Titre := ' Graphe de x -> sin( '.n.' x)' :
> plot (sin (n*x) , x = -Pi .. Pi , title = Titre);
```



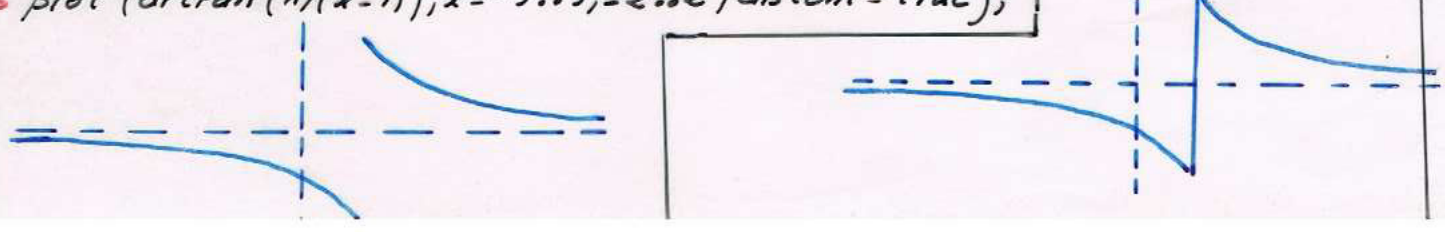
- Options permettant une amélioration de la qualité des tracés
  * L'option discont *

Exemple 2 . 3

```
> plot (arctan (1/(x-1)), x = -5..5, -2..2);
```

Exemple 2 . 4

```
> plot (arctan (1/(x-1)), x = -5..5, -2..2 , discont = true);
```

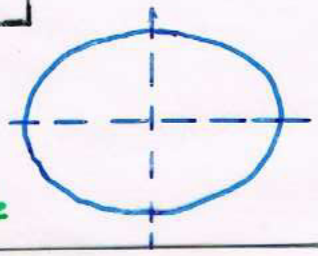


III Courbes Parametrees en cartesiennes

3-1 Tracé d'une courbe paramétrée

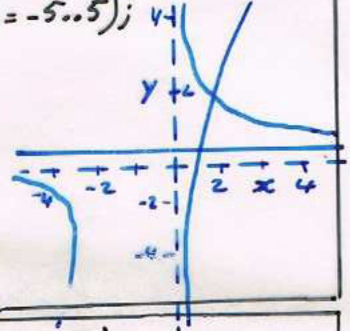
Exemple 3.1.1

```
> restart; plot([sin(u), cos(u), u=0..2*Pi]);
> plot([sin, cos, 0..2*Pi]); # m̂ syntaxe
> plot([sin, cos, 0..2*Pi], scaling=constrained); # cercle
```



Exemple 3.1.2

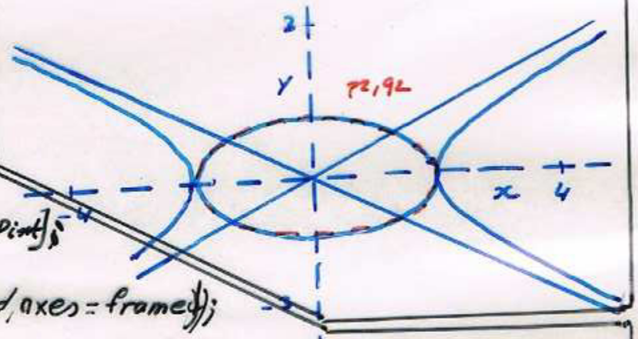
```
> plot([3/(u^2-2*u), (u^2-3)/u, u=-5..5], x=-5..5, y=-5..5);
                # branches infinies => calibrage
```



3-2. Tracé simultané de plusieurs courbes paramétrés

Exemple 3.2.1

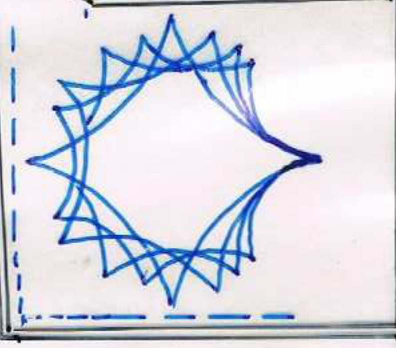
```
> p1:= 2*(1+u^2)/(1-u^2); q1:= 2*u/(1-u^2);
> p2:= 2*cos(u) : q2 := sin(u) :
> plot({[p2, q2, u=0..2*Pi], [p1, q1, u=-10..10]}, x=-5..5, y=-3..3);
```



3.3 Tracé d'une famille de courbes paramétrées

Exemple 3.3-1

```
> G:=t->[cos(u/t)+cos(u)/t, sin(u/t)-sin(u)/t, u=0..2*Pi*t];
> plot({G(3), G(4), G(5), G(6)}, scaling=constrained, axes=framed);
```

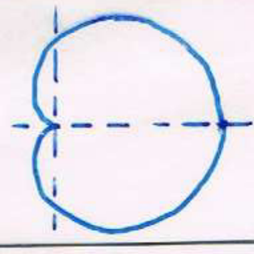


4 Courbes en polaires

4.1 Tracé d'une courbe en polaire

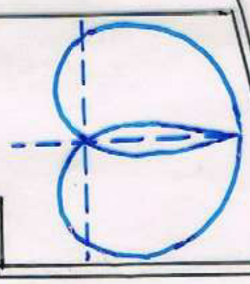
Exemple 4.1.1

```
> p:= 1+ cos(theta);
                    p:= 1+ cos(θ)
> plot([p, thetha, theta=0..2*Pi], coords=polar, scaling=constrained);
```



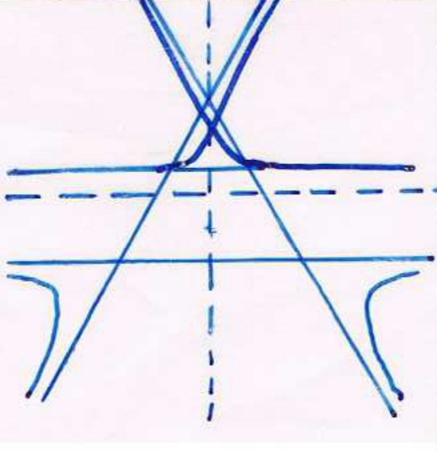
Exemple 4.1.2

```
> plot([cos(u), u-sin(u), u=0..2*Pi],
> coords=polar, scaling=constrained);
```



Exemple 4.1.3

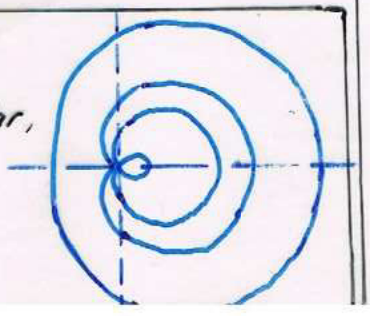
```
> plot([1/(sin(u)-sin(2*u)), u, u=0..2*Pi],
> x=-3..3, y=-3..3, coords=polar, scaling=constrained);
```



4.2 Tracé d'une famille de courbes en coord polaires

Exemple 4.2.1

```
> G:=t->[cos(u)+t, u, u=0..2*Pi]:
> plot({G(1/2), G(1), G(2)}, coords=polar,
> scaling=constrained);
```



5. COURBES DEFINIES IMPLICITEMENT

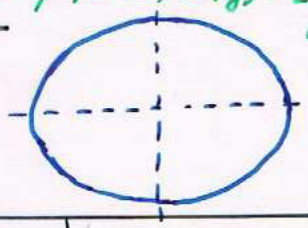
```
> with(plots, implicitplot):
```

5.1 Tracé d'une courbe définie implicitement

implicitplot(p, x=a..b, y=c..d) : trace la courbe d'éq p=0, (x,y) ∈ [a,b]×[c,d]

Exemple 5.1.1

```
> implicitplot(x^2+2*y^2-1, x=-2..2, y=-2..2);
```

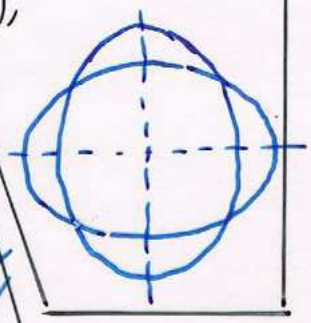


5.2 Tracé simultané de plusieurs courbes implicites

implicitplot({p1, p2, ..., pk}, x=a..b, y=c..d)

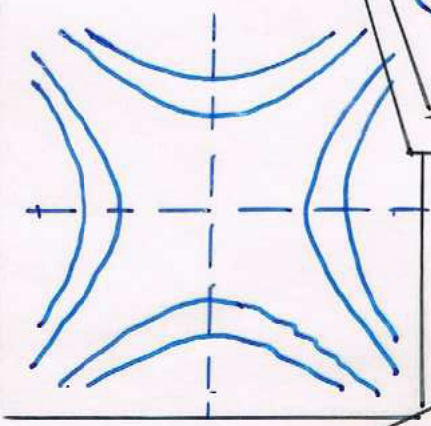
Exemple 5.2.1

```
> implicitplot({x^2+2*y^2=1, 2*x^2+y^2=1}, x=-2..2, y=-2..2);
```



Exemple 5.2.2

```
> F:=t -> x^2-y^2-t:
> implicitplot({F(-2), F(-1), F(1), F(2)}, x=-2..2, y=-2..2);
```



5.3 Précision du tracé des courbes implicites

Exemple .5.3.1

```
> f:=x^7+3*x*x*y*y+2*y^3:
> implicitplot(f, x=-2..2, y=-1..1);
```

```
> implicitplot(f, x=-1,.1, y=-1..1, grid=[50,50]);
```

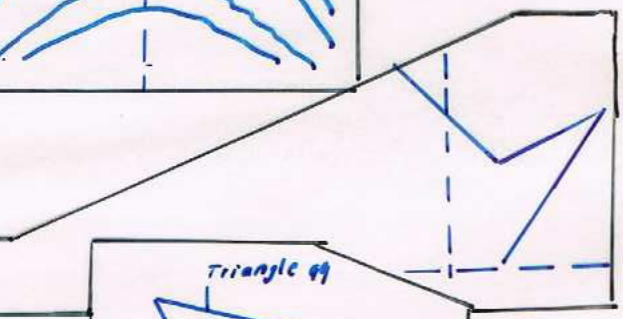
l'option grid=[m,n] (par défaut m=n=25)
m pts horizontal et n pts en vertical

6. TRACES POLYGONAUX

6.1 Tracé de lignes polygonales

Exemple 6.1.1

```
> plot([[1,1],[3,4],[1,3],[-1,5]]);
```

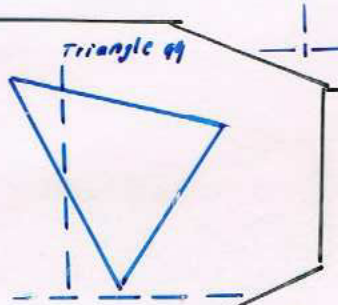


6.2. Tracé de polygones

```
> with(plots, polygonplot):
```

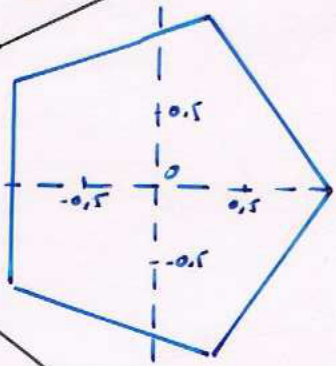
Exemple 6.2.1

```
> Triangle:=[[1,1],[3,4],[-1,5]];
> polygonplot(Triangle, title='Triangle quelconque');
```



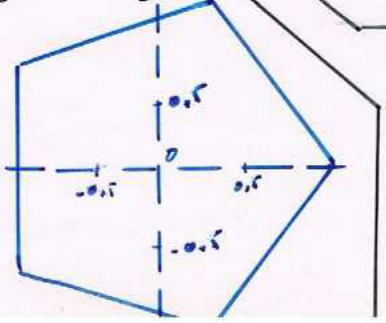
Exemple. 6.2.2

```
> Pentagone:=[seq([cos(2*k*Pi/5), sin(2*k*Pi/5)], k=0..4)]:
> polygonplot(Pentagone, scaling=constrained
  xtickmarks=3, ytickmarks=3);
```



Exemple. 6.2.3

```
> Pentagone:=[seq([cos(2*k*Pi/5), sin(2*k*Pi/5)], k=0..5)]:
> plot(Pentagone, scaling=constrained,
  xtickmarks=3, ytickmarks=3);
```



Equations et Inequations
Resolution Exacte

1 Equations polynomiales à une variable (de degré inférieur ou égal à 3)

Exemple 1.1

```
> Sol:= solve(x^3 - 3*x^2 * sqrt(2) + 4*sqrt(2) - 4);
```

solve(p,x) retourne la famille des racines sous forme d'une séquence

$Sol := 1+\sqrt{2}+\sqrt{3},\ 1+\sqrt{2}-\sqrt{3},\ -2+\sqrt{2}$

l'opérateur d'indiçage [] permet d'extraire un

```
> sol[1];
```

$1+\sqrt{2}+\sqrt{3}$

```
> evalf({sol}, 4);
```

$\{4.146, .682, -.586\}$

```
> Digits:=4 : solve(x^3 - 3*x^2 * sqrt(2) + 4*sqrt(2) - 4.0);
```

Le point décimal provoque un calcul en flottants.

$4.146 + 0.0002I,\ -.586 - .0001I,\ .682 - .0001I$

Exemple 1.2

```
> solve(x^3 - 6*x^2 + 8*x - 2);
```

$$\%2+\frac{4}{3}\%1+2,\ -\frac{1}{2}\%2-\frac{2}{3}\%1+2+\frac{1}{2}I\sqrt{3}\left(\%2-\frac{4}{3}\%1\right),\ -\frac{1}{2}\%2-\frac{2}{3}\%1+2-\frac{1}{2}I\sqrt{3}\left(\%2-\frac{4}{3}\%1\right)$$

$$\%1=\frac{1}{\left[1+\frac{1}{9}I\sqrt{111}\right]^{1/3}} \qquad \%2=\left(1+\frac{1}{9}I\sqrt{111}\right)^{1/3}$$

```
> u:= %2;
```

$u := \left(1+\frac{1}{9}I\sqrt{111}\right)^{1/3}$ Rq: $u := \left[27+3I\sqrt{111}\right]^{1/3}$

Exemple 1.3

```
> sol:= solve(1/3*x^2 + 3*x + a + 1, x);
```

$sol := -\frac{9}{2}+\frac{1}{2}\sqrt{69-12a},\ -\frac{9}{2}-\frac{1}{2}\sqrt{69-12a}$

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

avec un coeff. de type float

$-4.500+0.03000\sqrt{19170.-3333.a},\ -4.500-0.03000\sqrt{19170.-3333.a}$

Exemple 1.4

```
> sol:= solve(1/3*x^2 + 3*x + a + 1);
```

$sol := \left\{x = x,\ a = -\frac{1}{3}x^2 - 3x - 1\right\}$

Exemple 1.5

```
> s:= solve(x^2 - 2*x*sin(a) + 1, x);
```

$s := \sin(a)+\sqrt{\sin(a)^2-1},\ \sin(a)-\sqrt{\sin(a)^2-1}$

```
> s1:= simplify({s}, symbolic);
```

$s1 := \{\sin(a)+I\cos(a),\ \sin(a)-I\cos(a)\}$

```
> simplify(convert(s1, exp));
```

$\{-Ie^{(Ia)},\ Ie^{(-Ia)}\}$

2- Autres équations à une variable.

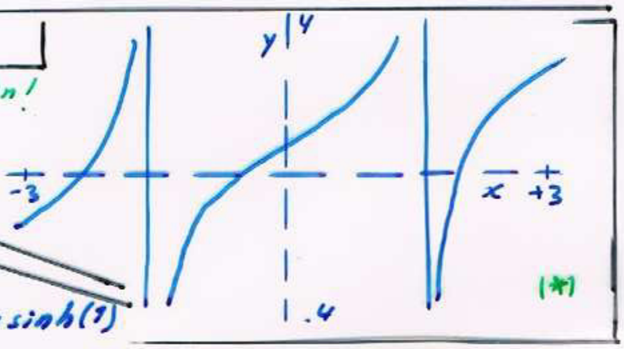
Exemple 2.1

```
> p:= sin(x) + sin(2*x) + sin(3*x) - 1 - cos(x) - cos(2*x) :
> solve(p,x);
> plot(p, x=0..2*Pi);
```

$\frac{1}{2}\pi,\ \frac{-1}{2}\pi,\ \frac{2}{3}\pi,\ \frac{1}{6}\pi,\ -\frac{5}{6}\pi,\ -\frac{2}{3}\pi$

Exemple 2.2

```
> p:=tan(x)+x+1 : S:= solve(p,x);    # bien!
  # retourner des séq. sol.
                                  S:=
> plot(p, x=-Pi..Pi, y=-5..5);
```



Exemple 2.3

```
> solve(cos(3*x)+7*cos(x), x);    # résol formelle
```

$\frac{1}{2}\pi,\ \frac{1}{2}\pi - I\,\mathrm{arcsinh}(1),\ \frac{1}{2}\pi + I\,\mathrm{arcsinh}(1)$

rien ne t'empêche de retourner des vals ∈ ℂ

3 systèmes d'équations

Exemple 3.1

```
> restart; sol:= solve({x+y=3, x*y=2}, {y,x});
```

$sol:=\{x=1, y=2\}, \{x=2, y=1\}$

```
> sol[1];    # L'opérateur d'indiçage [] permet d'extraire un élément de sol
```

$\{x=1, y=2\}$

```
> subs(sol[1], x);    # retourne la valeur de x, en laissant x libre
```

1

```
> subs(sol[2], [x,y]);    # retourne la liste formée des valeurs de x et y
```

$[2, 1]$

Exemple 3.2

```
> Eq:= {x + 2*y + 3*z = 0, 2*x + 3*y + 4*z = 1, 3*x + 4*y + 5*z = 2};
```

$Eq:=\{x+2y+3z=0,\ 2x+3y+4z=1,\ 3x+4y+5z=2\}$

```
> solve(Eq, {x,y,z});    # sys adm (∞) de sol
```

$\{y=-2z-1,\ x=z+2,\ z=z\}$

Exemple 3.3

```
> Eq:= { x+2*y+3*z=0, 2*x+3*y+4*z=1, 3*x+4*y+5*z=0};
```

$Eq:=\{x+2y+3z=0,\ 2x+3y+4z=1,\ 3x+4y+5z=0\}$

```
> s:= solve(Eq, {x,y,z});    # système impossible
```

$s:=$

Exemple 3.4

```
> s:= solve({tan(x)+x+1, y=1}, {x,y});    # voir graphe (*)
```

$s:=\{x=RootOf(\tan(-Z)+Z+1),\ y=y\}$

Exemple 3.5

```
> S:= { x+y+z=6, x^2+y^2+z^2=14, x*y*z=6};
```

$S:=\{xyz=6,\ x^2+y^2+z^2=14,\ x+y+z=6\}$

```
> solve(S, {x,y,z});
```

$\{y=1, z=2, x=3\}, \{y=1, z=3, x=2\}, \{z=1, y=2, x=3\}$
$\{x=1, y=2, z=3\}, \{z=1, x=2, y=3\}, \{x=1, z=2, y=3\}$

Systèmes linéaire dépendant de paramètres

Exemple 3.6

```
> eqn:= { cos(a)*x + sin(a)*y = u , sin(a)*x - cos(a)*y = v };
```

$$eqn:=\{\cos(a)x+\sin(a)y=u,\ \sin(a)x-\cos(a)y=v\};$$

```
> Sol:= solve(eqn,{x,y});
> simplify(Sol);
```

$$Sol:=\left\{x=\frac{\cos(a)u+\sin(a)v}{\cos(a)^2+\sin(a)^2},\ y=\frac{\sin(a)u-\cos(a)v}{\cos(a)^2+\sin(a)^2}\right\}$$

$$\{x=\cos(a)u+\sin(a)v,\ y=-\cos(a)v+\sin(a)u\};$$

Exemple 3.7

```
> Eq:= { a*x + y = 1 , x + a*y = 1 };
> solve(Eq,{x,y});
> Solve(subs(a=1,Eq),{x,y});
> Sol:= solve(subs(a=-1,Eq),{x,y});
```

$$Eq:=\{ax+y=1,\ x+ay=1\}$$

$$\left\{y=\frac{1}{a+1},\ x=\frac{1}{a+1}\right\}$$

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

$Sol:=$

le système est impossible, solve retourne une séquence vide

.4 Résolution de certaines inéquations.

Exemple 4.1

```
> solve(x^2 - 1 <= 0, x);
> solve(x^2 - 1 >= 0);
```

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

on peut Omettre le dernier x

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

Exemple 4.2

```
> solve(x^3 - 3*x^2 + x + 2 <= 0);
```

$$\left\{x\le \frac{1}{2}-\frac{1}{2}\sqrt{5}\right\},\left\{\frac{1}{2}+\frac{1}{2}\sqrt{5}\le x,\ x\le 2\right\}$$

Exemple 4.3

```
> solve((x-a)*(x-a-1) > 0);
> solve((x-a)*(x-a-1) > 0, x);
```

Ordonner les bornes

$$\{x<a\},\{a+1<x\}$$

error, (in solve) not implemented yet

II Resolution approcheé d'equations : fsolve

II_1 Equation algébriques à une variable.

Exemple II.1.1

```
> restart; p:= x^4 - 3*x^2 + x + 1: fsolve(p);
```

les racines réelles

$$-1.801937736,\ -.4450418679,\ 1.,\ 1.246979604$$

```
> fsolve(p, x=-1..0);
```

ds [-1,0]

$$-.4450418679$$

Exemple II.1.2

```
> p:= x^4 -3*x^2 +x+ Sqrt(2);
> Digits:= 4; fsolve (p);
```

$p := x^4 - 3x^2 + x + \sqrt{2}$

Digits := 4

-1.764, -.5632

```
> fsolve (p, x, complex);
```

-1.764, -.5632, 1.164 - .2627 I, 1.164 + .2627 I

Exemple II.1.3

```
> restart : p:= x^4 -3*x^2+x+1 : fsolve (p,x, maxsols=3);
```

-1.801937736, -.4450418679, 1.

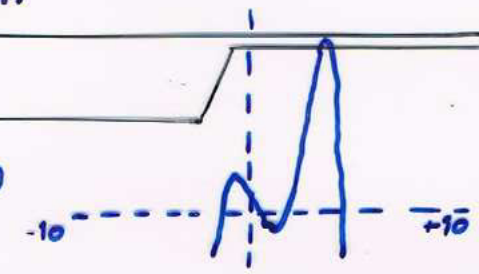
II.2. Autres équations à une variable

Exemple. II.2.1

```
> p:= x^3 -x +2 -exp(x) : fsolve (p);
```

1.806570269

```
> plot (p, x=-10..10, y=-2..10, numpoints=150);
```




```
> fsolve (p, x=-2..0);
```

-1.481974815

exemple II.2.2

```
> p:= (x^2 +1)*exp(x) : fsolve (p,x);
```

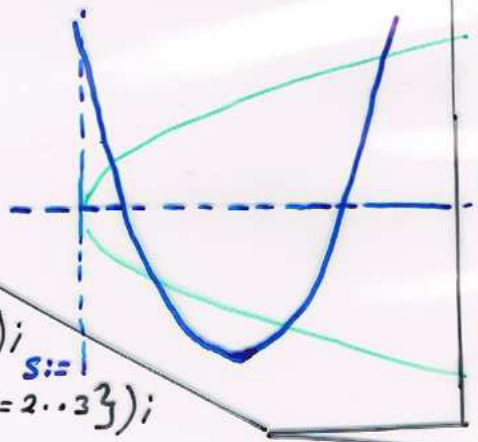
fsolve $((x^2+1)\exp(x), x)$

II.3 Systèmes d'équations

Exemple II.3.1

```
> fsolve ( { y^2-x, y -x^2 + 4*x -2 }, {x,y});
> plot ( { sqrt(x), -sqrt(x), x^2 -4*x +2 }, x=-1..5,
> y=-2.5..2.5);
```

{x = 4., y = 2.} #! eq = n. inc



Exemple II.3.2

```
> fsolve ( { y^2 -x, y-x^2 +4*x -2}, {x,y}, {y=0..1});
```

{x = .3819660113, y = .6180339888}

```
> S:= fsolve ( { y^2 -x, y-x^2 +4*x -2}, {x,y}, {y=-2..-1.4});
> S:= fsolve ( { y^2 -x, y-x^2 +4*x -2}, {x,y}, {y=-2..-1,4, x=2..3});
```

S := {x = 2.618033989, y = -1.618033989}

Exemple II.3.3

```
> restart; fsolve ( { a*x +1=0, a*y+1=0}, {x,y,a});
```

nb eq ≠ nb inc

Error, (in fsolve) ...

```
> fsolve ( { a*x +1=0, a*y+1=0}, {x,y});
```

on utilise pas les incs

Error, (in fsolve/gensys) ...

LIMITES / DERIVEES

I. LIMITES

I.1 Limite d'expressions

P : expression

- limit(p, x=a) : retourne la limite de p lorsque x tend vers a
- limit(p, x=a, right) : ... → $\lim_{x \to a^+} p$
- limit(p, x=a, left) : ..... → $\lim_{x \to a^-} p$
- limit(p, x=infinity) : .. → $\lim_{x \to +\infty} p$
- limit(p, x=-infinity) : - - - - - → $\lim_{x \to -\infty} p$
- limit(p, x=infinity, real) : - - - - - - - → $\lim_{|x| \to +\infty} p$

Exemple 1.1

```
> p:=(x^2-1)*ln((1+x)/(1-x));
> limit(p,x=1);
> limit(p/x,x=0)
> limit(ln(tan(x))/(tan(x)-1),x=Pi/4);
```

$p := (x^2-1)\ln\left(\frac{1+x}{1-x}\right)$

0

-2

1

Exemple 1.2

```
> limit(exp(1/x),x=0,right);
> limit(exp(1/x),x=0,left);
```

∞

0

Exemple 1.3

```
> limit(exp(x)/x,x=infinity); limit(exp(x)/x,x=-infinity);
> p:=(x^2+1)/(x^2-1): limit(p,x=infinity,real);
```

∞ , 0

1

Exemple 1.4

```
> x:=y+1 : limit(p,x=1)
> x:='x': limit(exp(1/x),x=0);
```

Error, (in limit) invalid arguments

undefined

Exemple 1.5

```
> limit(sin(1/x),x=0);
> limit(sin(x)+cos(x),x=infinity);
```

-1..1

-2..2

$|\sin(x)+\cos(x)| \leq \sqrt{2}$

Exemple 1.6

```
> limit(abs(sin(x)),x=infinity);
```

$\lim_{x \to \infty} |\sin(x)|$

I.2 limite de fonctions

Exemple I-2-1

```
> f:= x -> ((ln(x+1)/ln(x))^x - 1)*ln(x);
> limit(f(x), x=infinity);
```

$f := x \to \left(\left(\frac{\ln(x+1)}{\ln(x)}\right)^x - 1\right)\ln(x)$

1

Exemple I.2.2

```
> limit(f, 0);
```

Error, (in limit) invalid arguments

il ne faut omettre la variable de le second argument

I.3. limite d'expressions dépendant de paramètreo

Exemple I3.1

```
> p:= (1+ t/n)^n : limit(p, n=infinity);
```

e^t

Exemple I.3.2

```
> Lim := limit(exp(t*x), x=+infinity);
> limit(exp(abs(t)*x), x=infinity);
```

$Lim := \lim_{x\to\infty} e^{(tx)}$ $\quad \# = \begin{cases} 0 & t<0 \\ 1 & t=0 \\ +\infty & t>0 \end{cases}$

∞

Exemple I.3.3

```
> limit(exp(t*t*x), x=infinity);
```

$\lim_{x\to\infty} e^{(t^2 x)}$ # supose que les paramètreo ne sont pas réels

Exemple I.3.4

```
> subs(t=1, Lim)
> eval(subs(t=1, Lim));
> eval(subs(t=-1, Lim));
```

$\lim_{x\to\infty} e^x$ # eval est indispensable car subs n'évalue pas.

∞

0

I.4 limites des fonctions de plusieurs variables

```
limit(p, {x=a, y=b})
```

$\lim_{(x,y)\to(a,b)} p$

Exemple 1.4.1

```
> limit((x*x - y*y)/(x-y), {x=0, y=0});
> limit((x*y)/sqrt((x^2 + y^2)), {x=0, y=0});
L:=
> L = value(L);
```

0

$Limit\left(\frac{xy}{\sqrt{x^2+y^2}}, \{x=0, y=0\}\right)$

une forme non évaluée

Exemple 1.3.5

```
> assume(t>0); Lim;
> assume(t<0); Lim;
```

∞

0

II DERIVEES

II.1 Dérivées des expressions d'une seule variable

* Dériveé première d'une expression

exemple II.1.1

```
> diff(arctan(x),x);
```

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

```
> x:=y+1 : diff(arctan(x),x);
```

Error, wrong number (or type) of parameters in function diff

```
> restart; diff(arctan(f(x)),x);
```

$$\frac{\frac{\partial}{\partial x} f(x)}{1+f(x)^2}$$

* Dériveé d'ordre supérieur d'une expression

Exemple II.1.2

```
> diff(arctan(x),x,x,x);
```

$$-8\frac{x^2}{(1+x^2)^3} - 2\frac{1}{(1+x^2)^2}$$

```
> diff(arctan(x),x$4);
```

$$-48\frac{x^3}{(1+x^2)^4} + 24\frac{x}{(1+x^2)^3}$$

```
> normal(");
```

$$-24\frac{x(x^2-1)}{(1+x^2)^4}$$

pour réduire l'expression

II.2 Dérivées partielles des expressions de plusieurs variables

diff(p, $x_1, x_2, \ldots, x_n$) retourne $\frac{\partial^n p}{\partial x_1 \partial x_2 \cdots \partial x_n}$

Exemple II.2.1

```
> p:= ln(sqrt(x^2 + y^2)):
> r:= diff(p,x,x) ; s:= diff(p,x,y); t:= diff(p,y,y)
```

$$r := -2\frac{x^2}{(x^2+y^2)^2} + \frac{1}{x^2+y^2} \qquad s := -2\frac{xy}{(x^2+y^2)^2}$$

$$t := -2\frac{y^2}{(x^2+y^2)^2} + \frac{1}{x^2+y^2}$$

```
> normal(s*s - r*t);
```

$$\frac{1}{(x^2+y^2)^2}$$

II.3 Dérivées des fonctions d'une variable

* Dérivée première d'une fonction

Exemple II.3.1 :

```
> f := x -> arctan(x)/(1+x^2) : f_1 := D(f);
```

D : retourne une fct

$$f_{-1} := x \to \frac{1}{(1+x^2)^2} - 2\,\frac{\arctan(x)\,x}{(1+x^2)^2}$$

```
> normal(f_1);
> normal(f_1(x));
```

f_{-1}

$$\frac{1 - 2\arctan(x)\,x}{(1+x^2)^2}$$

```
> f_1 := unapply(", x);
```

transforme une expression en fonction

$$f_{-1} := x \to \frac{1 - 2\arctan(x)\,x}{(1+x^2)^2}$$

* Dérivée d'ordre supérieur d'une fonction

(D@@n)(f) retourne $f^{(n)}$

Exemple II.3.2

```
> f := x -> (1+x^2)^(-1) : f_3 := (D@@3)(f);
```

$$f_{-3} := x \to -48\,\frac{x^3}{(1+x^2)^4} + 24\,\frac{x}{(1+x^2)^3}$$

II.4 Dérivées partielles des fonctions de plusieurs variables.

D[$i_1, i_2, \ldots, i_r$](f) retourne $(x_1, x_2, \ldots, x_p) \to \dfrac{\partial^r f}{\partial x_{i_1}\,\partial x_{i_2}\cdots\partial x_{i_r}}(x_1, x_2, \ldots, x_p)$

Exemple II.4.1

```
> f := (x,y) -> arctan(y/x) : f_1 := D[2](f);
```

$$f_{-1} := (x,y) \to \frac{1}{x\left(1+\frac{y^2}{x^2}\right)}$$

```
> normal("(x,y));
```

$$\frac{x}{x^2+y^2}$$

```
> f_1 := unapply(", x, y);
```

$$f_{-1} := (x,y) \to \frac{x}{x^2+y^2}$$

Exemple II.4.2

```
> f_2 := D[1,2](f);
```

calcul de $\partial^2 f / \partial x\,\partial y$

$$f_{-2} := (x,y) \to -\frac{1}{x^2\left(1+\frac{y^2}{x^2}\right)} + 2\cdot\frac{y^2}{x^4\left(1+\frac{y^2}{x^2}\right)^2}$$

```
> normal("(x,y));
```

$$-\frac{x^2-y^2}{(x^2+y^2)^2}$$

```
> f_3 := unapply(normal(D[1,1,2](f)(x,y)), x, y);
```

$$f_{-3} := (x,y) \to 2\cdot\frac{x(x^2-3y^2)}{(x^2+y^2)^3}$$

DEVELOPPEMENTS LIMITES

I La Fonction Series

I.1 Obtention de développements limités

* Développement limité en a

- p : expression (x) ; a une expression indépendante de x, $n \in \mathbb{N}$, $n \neq 0$

  Series (p, x=a, n) retourne DL (p, au voisinage de a, à l'ordre n)

- Series (p, x=a) <=> Series (p, x=a, 6) (Défaut)
- $O((x-a)^m)$: notation de Landau, correspond DL à l'ordre m-1

Exemple I.1.1

```
> Series (1/(1+x), x=1, 4);
```

$$\frac{1}{2} - \frac{1}{4}(x-1) + \frac{1}{8}(x-1)^2 - \frac{1}{16}(x-1)^3 + O((x-1)^4)$$

```
> Series (1/(1-x^4), x, 5);        # sans peine ! m > n
```

$$1 + x^4 + O(x^8)$$

```
> Series ((sin(x)-x)/x^3, x=0, 7);        # ici on a m = n-3 → x^6
```

$$-\frac{1}{6} + \frac{1}{120}x^2 + O(x^4)$$

```
> restart; Series (Log(1+x) - x + x^2/2)/x^2, x=0, 5);        # ici, m = n-2
```

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

```
> series (Log(1+x) - x + x^2/2)/x^2, x=0, 10);
```

$$\frac{1}{3}x - \frac{1}{4}x^2 + \frac{1}{5}x^3 - \frac{1}{6}x^4 + \frac{1}{7}x^5 - \frac{1}{8}x^6 + \frac{1}{9}x^7 + O(x^8)$$

```
> series (Log(1+x) - x + x^2/2)/x^2, x=0, 5);        # mais ici, m = n
```

$$\frac{1}{3}x - \frac{1}{4}x^2 + \frac{1}{5}x^3 - \frac{1}{6}x^4 + O(x^5)$$

La fct Series conserve les DL calculés ds 1 table de remember

Exemple I.1.2

```
> series (sin(x), x, 8);        # par défaut on peut sous-entendre = 0
```

$$x - \frac{1}{6}x^3 + \frac{1}{120}x^5 - \frac{1}{5040}x^7 + O(x^8)$$

Exemple I.1.3

```
> f := 'f' : series (f(x), x, 4);
```

$$f(0) + D(f)(0)x + \frac{1}{2}D^{(2)}(f)(0)x^2 + \frac{1}{6}D^{(3)}(f)(0)x^3 + O(x^4)$$

* Développement limité en l'infini

```
> series (1/(1+x), x = infinity);
```

$$\frac{1}{x} - \frac{1}{x^2} + \frac{1}{x^3} - \frac{1}{x^4} + \frac{1}{x^5} + O\left(\frac{1}{x^6}\right)$$

I.2 Développement généralisés

Laurent (puissances entières positives et négatives)
Puiseaux (exposants fractionnaires)

Exemple I.2.1

```
> series(cot(x),x,6);          # Développement de Laurent
```

$$x^{-1} - \frac{1}{3}x - \frac{1}{45}x^3 - \frac{2}{945}x^5 + O(x^6)$$

```
> series(sqrt(x*(1+x)),x,4);   # Développement de Puiseux
```

$$\sqrt{x} + \frac{1}{2}x^{3/2} - \frac{1}{8}x^{5/2} + \frac{1}{16}x^{7/2} + O\left(x^{9/2}\right)$$

Exemple I.2.2

```
> p:=sin(x)/(1+x): series(p,x=infinity,6);
```

$$\frac{\sin(x)}{x} - \frac{\sin(x)}{x^2} + \frac{\sin(x)}{x^3} - \frac{\sin(x)}{x^4} + \frac{\sin(x)}{x^5} + O\left(\frac{1}{x^6}\right)$$

Exemple I.2.3

```
> p:=x^x : series(p,x,4);
```

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

Exemple I.2.4

```
> series(n!, n=infinity, 3);   # formule de Stirling (calculée à l'aide fct Gamma)
```

$$\frac{1}{\left(\frac{1}{n}\right)^n e^n}\cdot\left[\frac{\sqrt{2}\sqrt{\pi}}{\sqrt{\frac{1}{n}}} + \frac{1}{12}\sqrt{2}\sqrt{\pi}\sqrt{\frac{1}{n}} + \frac{1}{288}\sqrt{2}\sqrt{\pi}\left(\frac{1}{n}\right)^{3/2} + O\left(\left(\frac{1}{n}\right)^{5/2}\right)\right]$$

I.3 Partie régulière d'un développement

Exemple I.3.1

```
> p:=sin(x) : u:=series(p,x,6);
```

$$u := x - \frac{1}{6}x^3 + \frac{1}{120}x^5 + O(x^6)$$

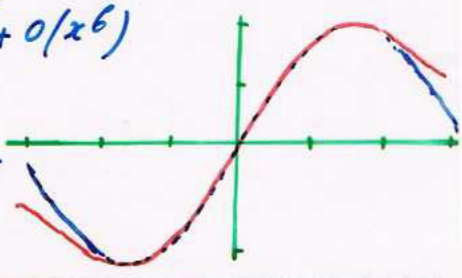
```
> subs(x=2,u);
```

Error, invalid substitution in series

```
> u_p := convert(u,polynom);
```

$$u_p := x - \frac{1}{6}x^3 + \frac{1}{120}x^5$$

```
> plot({p,u_p},x=-Pi..Pi);
```



I.4. Obtention d'un équivalent.

Exemple I.4.1

```
> f:=x->tan(sin(x))-sin(tan(x)) : series(leadterm(f(x)),x,5);
```

Error, (in series/leadterm) unable to compute leading term

```
> series(leadterm(f(x)),x,20);
```

$$\frac{1}{30}x^7$$

I.5 Les limites de la fonction series

Exemple I.5.1

```
> p:=exp(-1/x^2) : n:=4 : limit(diff(p,x$n),x=0);   # On peut utiliser d'autre valeurs de n
```

0

```
> series(p,x=0);
```

Error, (in series/exp) unable to compute series.

II Operation sur les Developpements limités

II-1 Sommes, quotients, produits de développements limités

Exemple II-1-1

```
> restart; u:= series(sinh(x),x); v:= series(cos(x),x);
```

$$u := x + \frac{x^3}{6} + \frac{1}{120}x^5 + O(x^6) \qquad v := 1 - \frac{1}{2}x^2 + \frac{1}{24}x^4 + O(x^6)$$

```
> expand(u*v);
```

$$\left(x + \frac{x^3}{6} + \frac{1}{120}x^5 + O(x^6)\right)\left(1 - \frac{1}{2}x^2 + \frac{1}{24}x^4 + O(x^6)\right)$$

```
> series(u*v,x);
```

$$x - \frac{1}{3}x^3 - \frac{1}{30}x^5 + O(x^6)$$

```
> series(u/v,x);
```

$$x + \frac{2}{3}x^3 + \frac{3}{10}x^5 + O(x^6)$$

Exemple II-1-2

```
> u:= series(sinh(x),x,5): v:= series(cos(x),x,8): w:= series(u*v,x,12);
```

$$w := x - \frac{x^3}{3} + O(x^5)$$

II.2 Composés et inverses de développements limités

* Composés de développements limités

Exemple II.2.1

```
> restart; u:= series(sin(x),x,5);
```

$$u := x - \frac{x^3}{6} + O(x^5)$$

```
> v:= series(tan(u),x);
```

$$v := x + \frac{x^3}{6} + O(x^5)$$

* Développement limité d'une "fonction" réciproque

Exemple II.2.2

```
> u:= series(tan(x),x);
```

$$u := x + \frac{1}{3}x^3 + \frac{2}{15}x^5 + O(x^6)$$

```
> solve(u=y,x);
```

$$y - \frac{1}{3}y^3 + \frac{1}{5}y^5 + O(y^6)$$

\# solve(u=y,x) ↔ DL f^{-1} au voisinage $b = f(a)$

```
> solve(series(tan(x),x=Pi/4,3)=y,x);
```

$$\frac{1}{4}\pi + \frac{1}{2}(y-1) - \frac{1}{4}(y-1)^2 + O\left((y-1)^3\right)$$

Exemple II.2.3

```
> u:= series(cos(x),x,6);
```

$$u := 1 - \frac{1}{2}x^2 + \frac{1}{24}x^4 + O(x^6)$$

```
> solve(u=y,x);
```

$$I\sqrt{2}\sqrt{y-1} - \frac{1}{12}I\sqrt{2}(y-1)^{3/2} + O\left((y-1)^{5/2}\right)$$

```
> solve(u=1-h,x);
```

$$\sqrt{2}\sqrt{h} + \frac{1}{12}\sqrt{2}h^{3/2} + O\left(h^{5/2}\right)$$

```
> subs(h=1-y,"");
```

$$\sqrt{2}\sqrt{1-y} + \frac{1}{12}\sqrt{2}(1-y)^{3/2} + O\left((1-y)^{5/2}\right)$$

```
> s:= series(cot(x),x);
```

$$x^{-1} - \frac{1}{3}x - \frac{1}{45}x^3 + \frac{2}{945}x^5 + O(x^6)$$

```
> u:= solve(s=y,x);
```

u :=

II.3 Intégration d'un développement limité

Exemple II.3.1

```
> u: series(1/(x^2+x+1), x=1,3);
```

$$\frac{1}{3} - \frac{1}{3}(x-1) + \frac{2}{9}(x-1)^2 + O((x-1)^3)$$

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

$$\frac{1}{3}(x-1) - \frac{1}{6}(x-1)^2 + \frac{2}{27}(x-1)^3 + O((x-1)^4)$$

```
> u:= series(cos(x)/x, x, 4);
```

$$u := x^{-1} - \frac{1}{2}x + O(x^3)$$

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

$$\ln(x) - \frac{1}{4}x^2 + O(x^4)$$

III. Développement d'une fonction implicite

III.1 Evaluation de RootOf(p,y,b) # représente la racine p=0 "la plus proche" de b

Exemple 3.1.1

```
> restart; P:= x^3 - 3*x + 1 : S:= RootOf(P,x,2);
```

$$S := \text{RootOf}(-Z^3 - 3Z + 1, 2)$$

```
> evalf(S);
```

1.532088886

```
> evalf(RootOf(P, x, 1));
```

1.532088886

```
> evalf(RootOf(P,x,-1.1));
```

-1,879385242

III.2 Développement limité d'une fonction implicite

Series et de RootOf permet de calculer des DL de fct implicites

Series(RootOf(p,y,b), x=a) ↻ DL de g(x) tq f(x,g(x))=0

Exemple III.2.1

```
> p:= x^3 + y^3 - 3*x*y - 2;
```

INTEGRATION ET SOMMATION

I INTEGRATION

I.1. Calcul exact d'intégrales et de primitives

p: une expression de la variable 'x'

int(p, x=a..b) # $\int_a^b p\,dx$; int(p,x) # une primitive de p % à x

Exemple 1.1.1

```
> restart; int(sin(x)^2, x=0.. Pi/2);
```

$\frac{1}{4}\pi$

```
> p:=(2*x^2+x+3)*exp(x);
```

$p := (2x^2+x+3)e^x$

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

$2e^x.x^2 - 3e^x x + 6e^x$ # aucune const d'intégration

I.2 Intégrales généralisées.

Exemple 1.2.1

```
> f:= 1/sqrt((1-x)*(x+1)) : int(f, x=-1..1);
```

π # $\int_{-1}^{+1} \frac{dx}{\sqrt{(1-x)(1+x)}}$

```
> int( sin(x)/x , x=0.. infinity); # ∫_0^∞ sin(x)/x dx
```

$\pi/2$

Exemple 1.2.2

```
> int(sin(x)^2/(1+x^(3/2)), x=0..infinity); # intégrale convergente
```

$$\int_0^\infty \frac{\sin(x)^2}{1+x^{3/2}}\,dx$$

```
> int(sin(x)^2/(1+x^(1/2)), x=0.. infinity); # integrale divergente
```

$$\int_0^\infty \frac{\sin(x)^2}{1+\sqrt{x}}\,dx$$

Maple un resultat ne préjuge en rien de la convergence de ∫

```
> int (1/x, x=0..1);
```

divergente

```
> int(abs(sin(x)/x), x=0.. infinity);
```

∞ $\int_0^\infty \left|\frac{\sin(x)}{x}\right| dx$ # divergente

```
> int(x*sin(x), x=0.. infinity);
```

undefined # sans tendre vers ±∞

```
> evalf( int(sin(x)^2/(1+x^(3/2)), x=0.. infinity);
```

1.117915171

I.3 Forme inerte Int (intégrale sous forme non évaluée)

la fonction value $\xrightarrow{tf}$ int

Exemple I.3.1

```
> Intg := Int(sin(x)^2, x=0..Pi/2);
> Intg = value(Intg);
> Int(1/(x^(2/3)*(1-x)^(1/3)), x=0..1): " = value(");
> Intg := Int(1/sqrt(2*cos(x)^2 + 3*sin(x)^2), x=0..Pi/2);
> simplify(Intg);
```

$$Intg := \int_0^{\pi/2} \sin(x)^2\,dx$$

$$\int_0^{\pi/2} \sin(x)^2\,dx = \pi/4$$

$$\int_0^{1} \frac{1}{x^{2/3}(x-1)^{1/3}}\,dx = \frac{2}{3}\pi\sqrt{3}$$

$$Intg := \int_0^{\pi/2} \frac{1}{\sqrt{2\cos(x)^2 + 3\sin(x)^2}}\,dx$$

appliquer simplify, convert, ...

$$\int_0^{\pi/2} \frac{1}{\sqrt{-\cos(x)^2+3}}\,dx$$

I.4. Evaluation numérique d'intégrales :

Exemple I.4.1

```
> p := 1/sqrt(2*cos(x)^2 + 3*sin(x)^2): int(p, x=0..Pi/2): evalf(");
1.001077380
> int(sin(x)^2/(1+x^(3/2)), x=0..infinity): evalf(",5);
1.1179
> p := 1/sqrt(2*cos(x)^2 + 3*sin(x)^2): Intg := Int(p, x=0..Pi/2): evalf(Intg);
1.001077380
> Digits := 20: evalf(Intg);
1.0010773804561062361
```

II. Operations Sur les formes non Evaluees.

II-1 Intégration par parties _ Changement de variable dans une intégrale

II-3 Dérivation sous le signe Intégral :

- diff(int(f(t,x), x=a..b), t) # la valeur de int(diff(f(t,x),t), x=a..b)
- diff(Int(f(t,x), x=a..b), t) # Int(diff(f(t,x),t), x=a..b)

Exemple 2.3.1

```
> restart: F := Int(ln(x^2*cos(t)^2 + sin(t)^2), t=0..Pi/2); F1 := diff(F,x);
```

$$F := \int_0^{\pi/2} \ln\left(x^2\cos(t)^2 + \sin(t)^2\right)dt \quad , \quad F1 := \int_0^{\pi/2} 2\cdot\frac{x\cos(t)^2}{x^2\cos(t)^2 + \sin(t)^2}\,dt$$

II.4. Développement limité d'une primitive (au voisinage de 0)

Exemple II.4.1

```
> restart; intgr:=int(exp(-x^3),x); series(intgr,x=0,6);
```

$intgr := \int e^{-x^3}\,dx$, $x - \frac{1}{4}x^4 + O(x^7)$

```
> series(intgr,x=1,6);
```

Error, (in series/int) invalid arguments

```
> intgr:=int(exp(-x^3),x=0..t);
```

$intgr := \int_0^t e^{-x^3}\,dx$

```
> series(intgr,t=1,3);
```

$\int_0^1 e^{-x^3}dx + e^{-1}(t-1) - \frac{3}{2}e^{-1}(t-1)^2 + O((t-1)^3)$

Exemple II.4.2

```
> intgr:=int(sin(1+t^3)/t,t): series(intgr,t);
```

D.L. de puissance entière

$\sin(1)\ln(t) + \frac{1}{3}\cos(1)t^3 + O(t^6)$

```
> intgr:=int(exp(-sqrt(x^3)),x): series(intgr,x);
```

D.L. de puissance non entière

Error, (in series/int) unable to compute series

```
> int(series(exp(-sqrt(x^3)),x),x);
```

$x - \frac{2}{5}x^{5/2} + \frac{1}{8}x^4 - \frac{1}{33}x^{11/2} + O(x^7)$

III SOMMATION DISCRETE

III.1 Sommes indéfinies:

$f(k)$: expression math de 'k' ; sum(f(k),k) ↔ $\sum f(k)$ (Bernoulli-Maenck)

Exemple III.1.1

```
> sum(k^3,k);
```

$\frac{1}{4}k^4 - \frac{1}{2}k^3 + \frac{1}{4}k^2$ # $\sum_k k^3$

```
> factor(");
```

$\frac{1}{4}k^2(k-1)^2$

```
> sum(k*k!,k);
```

$k(k-1)!$ # $\sum_k k.k!$

```
> sum(cos((2*k+1)*a),k);
```

$\frac{\sin(ak)\cos(ak)}{\sin(a)}$ # $\sum_k \cos((2k+1)a)$

Exemple III.1.2

```
> sum(sinh((2*k+1)*a),k);
```

$\sum_k \sinh((2k+1)a)$

```
> S:=eval(convert(",exp));
```

$S := \frac{1}{2}\,\frac{e^a\left((e^{ax})^4+1\right)}{\left((e^a)^2-1\right)(e^{ak})^2}$

```
> S:=simplify(simplify(convert(S,trig)),{cosh(a)^2=sinh(a)^2+1});
```

$\frac{1}{2}\,\frac{-1+2\cosh(ak)^2}{$

III.2. Sommes finies

sum(f(k), k=n..m) # • $n \le m$ $f(n)+f(n+1)+\cdots f(m-1)+f(m)$

• $n > m$ $-\left(f(m+1)+f(m+2)+\cdots f(n-2)+f(n-1)\right)$

Exemple III.2.1 :

```
> k:='k'; sum(1/k, k=1..30);          # obs(n-m) < 100, sinon indéfinie
```

$$\frac{9304682830147}{232908956280}$$