FONCTIONS D'UNE VARIABLE RÉELLE : DÉFINITION, ENSEMBLE DE DÉFINITION, LIMITE, CONTINUITÉ, GRAPHE

Bernard Dupont

Bernard.Dupont@univ-lille1.fr

Avec ce chapitre, on aborde l'Analyse, domaine mathématique dont les économistes font un usage massif. Afin de se conformer aux usages en vigueur en matière de progression pédagogique , on commence ici par étudier les fonctions d'une variable réelle même s'il est patent que la théorie économique utilise la plupart du temps des fonctions de plusieurs variables, voire des systèmes de fonctions de plusieurs variables. Ce chapitre aborde les questions de la manière d'écrire une fonction en Maple, de l'ensemble de définition, des limites, de la continuité et de la représentation graphique. La dérivation, la différentiabilité, l'intégration, la représentation graphique approfondie seront étudiées dans les chapitres suivants.

Définitions préalables : fonctions, expressions

Cette section est importante à plus d'un titre. D'une part, elle montre qu'il existe deux manières de définir une fonction mathématique suivant qu'on insiste sur le triplet "ensemble de départ; ensemble d'arrivée; correspondance entre les éléments de l'ensemble de départ et ceux de l'ensemble d'arrivée" ou sur l'expression en jeu dans la correspondance, c'est à dire la formule de calcul.
Il faut bien comprendre que le terme fonction est extrêmement ambigu dans le contexte d'un logiciel de calcul symbolique. Une fonction informatique n'est pas une fonction mathématique. Elle est en effet une commande précise, résultat d'un programme, comprenant un ou plusieurs arguments et une ou plusieurs options.

Fonction mathématique définie par une procédure

Une fonction numérique d'une variable réelle regroupe trois éléments, ce que les usagers ont tendance à oublier : un ensemble de départ, un ensemble d'arrivée et un mode de correspondance entre un élément de l'ensemble de départ et un élément de l'ensemble d'arrivée.
On note généralement x un nombre quelconque pris dans l'ensemble de départ : c'est une variable muette. Et on note y le correspondant unique, appelé image, de x dans l'ensemble d'arrivée. Le mode de correspondance est le plus souvent une formule précisant comment calculer y à partir de x. En toute rigueur, une fonction ne doit pas s'écrire par le racourci $y = f(x)$ mais bien comme ceci : $x \to y = f(x)$.C'est ce que demande Maple.
Fondamentalement, il faut préciser le mode de correspondance en tapant d'abord la variable muette, puis en tapant à la suite les deux symboles **-** (trait d'union) et **>** (strictement supérieur à), enfin l'expression de l'image. On obtient alors un objet informatique du type "procédure".
Les deux exemples suivants sont parfaitement licites :

```
> restart;
  x->x^2+1;#le mode de correspondance est explicité
  whattype(%);
  c->u(c);#le mode de correspondance est non explicité
  whattype(%);
```

$$x \to x^2 + 1$$

$$procedure$$

$$c \to u(c)$$

$$procedure \qquad (1.1.1)$$

Bien que ce ne soit pas nécessaire stricto sensu, il est recommandé d'assigner ce mode de correspondance à un identificateur par la combinaison "deux-points" + "égal", soit **:=**. Par exemple, si on désire appeller F la fonction qui à x associe x^2+1, on écrira :

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

$$F := x \to x^2 + 1 \qquad (1.1.2)$$

On peut alors calculer n'importe quelle valeur prise par la fonction, par exemple **F(5)**ou **F(sqrt(2))**:

```
> F(5);F(sqrt(2));
```

$$26$$
$$3$$

Mais voici bien la preuve que x est une variable muette :

```
> F(t);F(alpha);F(jazz);
```

$$t^2+1$$
$$\alpha^2+1$$
$$jazz^2+1$$

Fonction mathématique définie par une expression informatique

On peut se contenter de se conforter à un usage solidement établi et définir une fonction en exprimant seulement l'image de x par f, soit $f(x)$. Dans ce cas, on assigne à un nom - **F1** dans l'exemple qui suit - une expression de x. Dans l'exemple suivant, Maple détecte une expression du type addition.

```
> F1:=x^2+1;
  whattype(F1);
```

$$F1 := x^2+1$$
$$`+`$$

Pour évaluer cette expression pour un argument symbolique ou chiffré de la variable **x**, on procédera par substitution en utilisant la commande **subs** ou par évaluation en invoquant la commande **eval**.

```
> subs(x=alpha,F1);#Evaluation de F1 en remplaçant x par
  alpha
  subs(x=10,F1);#Evaluation de F1 en remplaçant x par le
  nombre 10
```

$$\alpha^2+1$$
$$101$$

```
> eval(F1,x=beta);#Evaluation symbolique par utilisation de
  la commande eval
  eval(F1,x=10);#Evaluation chiffrée par utilisation de la
  commande eval
```

$$\beta^2+1$$
$$101$$

Dans les deux cas, la variable **x** est restée libre :

```
> about(x);
x:
  nothing known about this object
```

Si on affecte la variable **x**, l'expression **F1** est automatiquement calculée :

```
> x:=10.255;F1;
```

$$x := 10.255$$

$$106.165025 \tag{1.2.1}$$

La réaffectation de **x** déclenche une nouvelle évaluation de **F1** :

```
> x:=delta;F1;
```

$$x := \delta$$

$$\delta^2 + 1 \tag{1.2.2}$$

Pour retrouver l'expression initiale de **F1**, il faut désassigner la variable **x** :

```
> x:='x';F1;
```

$$x := x$$

$$x^2 + 1 \tag{1.2.3}$$

Passerelles entre expressions et fonctions

Soit *f* une fonction dépendant de la variable *x* assignée à **f** et *P* une expression assignée à **P**. Transformer une fonction en expression n'est pas compliqué : il suffit d'assigner au nom de l'expression **P** l'image de la fonction **f(x)**.

```
> f:=x->x^2;
  P:=f(x);
  x:=4;P;
  x:=5;P;
```

$$f := x \rightarrow x^2$$

$$P := x^2$$

$$x := 4$$

$$16$$

$$x := 5$$

$$25$$

Inversement, transformer une expression dépendant de la variable **x** en fonction de cette variable se fait avec la commande au nom des plus curieux (puisqu'il suggère le contraire de ce qu'elle fait) : **unapply(expression,variable)**.

```
> restart;P:=x^2;f:=unapply(P,x);f(2.5);
```

$$P := x^2$$

$$f := x \rightarrow x^2$$

$$6.25$$

On ne recommande pas la manipulation consistant à commencer par écrire une expression puis l'appeler en image d'une fonction. Elle semble logique mais risque de décevoir car elle interdit l'évaluation de la fonction en un point.

```
> g:=x->P;
  g(2.5);
```

$$g := x \rightarrow P$$
$$x^2 \qquad (1.3.1)$$

Propos d'étape

Soit la fonction mathématique $f : x \rightarrow y = f(x)$. Maple permet de manipuler l'expression $f(x)$ et la fonction f. Dans le premier cas, x est la variable indépendante nominative, dans le second cas le mode de correspondance convient à n'importe quel symbole. Il convient de bien faire la différence entre ces deux entités puisque les manipulations sont différentes. Au reste, on a vu que leur type informatique est nettement différencié.

Quelle que soit la solution adoptée, on peut aisément construire une fonction mathématique simple en exploitant les règles d'opération élémentaires. Ainsi, les fonctions puissance, monômes, polynômes, rationnelles font simplement appel aux touches du clavier :

```
> fpoly:=a*x^3+b*x^2+c*x+d;
  Fpoly:=unapply(fpoly,x);
  frat:=(a*x+b)/(c*x^2+d*x+e);
  Frat:=unapply(frat,x);
```

$$fpoly := a\,x^3 + b\,x^2 + c\,x + d$$
$$Fpoly := x \rightarrow a\,x^3 + b\,x^2 + c\,x + d$$
$$frat := \frac{a\,x + b}{c\,x^2 + d\,x + e}$$
$$Frat := x \rightarrow \frac{a\,x + b}{c\,x^2 + d\,x + e} \qquad (1.4.1)$$

Au delà, on a la possibilité de puiser dans l'éventail des fonctions prédéfinies.

Fonctions prédéfinies

Maple prédéfinit les fonctions mathématiques "usuelles" suivantes :

```
> FunctionAdvisor( known_functions );
The functions on which information is available via
    > FunctionAdvisor( function_name );
are:
```

[AiryAi, AiryBi, AngerJ, BesselI, BesselJ, BesselK, BesselY, B, ChebyshevT, ChebyshevU, Chi, Ci, CoulombF, CylinderD, CylinderU, CylinderV, Dirac, Ei, EllipticCE, EllipticCK, EllipticCPi, EllipticE, EllipticF, EllipticK, EllipticModulus, EllipticNome, EllipticPi, FresnelC, FresnelS, Fresnelf, Fresnelg, Γ, GaussAGM, GegenbauerC, HankelH1, HankelH2, Heaviside, HermiteH, HeunB, HeunBPrime, HeunC, HeunCPrime, HeunD, HeunDPrime, HeunG, HeunGPrime, HeunT, HeunTPrime, Hypergeom, $\Im$, InverseJacobiAM, InverseJacobiCD, InverseJacobiCN, InverseJacobiCS, InverseJacobiDC, InverseJacobiDN, InverseJacobiDS, InverseJacobiNC, InverseJacobiND, InverseJacobiNS, InverseJacobiSC, InverseJacobiSD, InverseJacobiSN, JacobiAM, JacobiCD, JacobiCN, JacobiCS, JacobiDC, JacobiDN, JacobiDS, JacobiNC, JacobiND, JacobiNS, JacobiP, JacobiSC, JacobiSD, JacobiSN, JacobiTheta1, JacobiTheta2, JacobiTheta3, JacobiTheta4, JacobiZeta, KelvinBei, KelvinBer, KelvinHei, KelvinHer, KelvinKei, KelvinKer, KummerM, KummerU, LaguerreL, LambertW, LegendreP, LegendreQ, LerchPhi, Li, LommelS1, LommelS2, MathieuA, MathieuB, MathieuC, MathieuCE, MathieuCEPrime, MathieuCPrime, MathieuExponent, MathieuFloquet, MathieuFloquetPrime, MathieuS, MathieuSE, MathieuSEPrime, MathieuSPrime, MeijerG, Ψ, ℜ, Shi, Si, SphericalY, Ssi, StruveH, StruveL, WeberE, WeierstrassP, WeierstrassPPrime, WeierstrassSigma, WeierstrassZeta, WhittakerM, WhittakerW, Wrightomega, ζ, abs, arccos, arccosh, arccot, arccoth, arccsc, arccsch, arcsec, arcsech, arcsin, arcsinh, arctan, arctanh, argument, bernoulli, binomial, cos, cosh, cot, coth, csc, csch, csgn, dawson, dilog, doublefactorial, erf, erfc, erfi, euler, exp, factorial, harmonic, hypergeom, ln, lnGAMMA, log, pochhammer, polylog, sec, sech, signum, sin, sinh, stirling1, stirling2, tan, tanh]

La gamme est impressionnante. En économie, on se contente ordinairement des suivantes : `log`, `exp`, `cos`, `sin`, `tan`, `arccos`, `arcsin`, `arctan`, `factorial`, `bernouilli`, `binomial,Chi`, `hypergeom`. Quelques incursions dans des zones plus complexes ne sont pas à exclure, mais elles sont rares.

A chacune de ces fonctions prédéfinies correspond un programme appelé **procédure**. Leurs concepteurs sont des mathématiciens qui n'hésitent pas, quand c'est possible, à retenir le corps des nombres complexes comme ensemble de définition. Les utilisateurs seront donc parfois surpris par les résultats de calcul comme on le verra plus bas.

Fonctions définies par morceaux

Il arrive qu'on traite un phénomène économique qui est soumis à des variations brutales. On le décrit par une fonction définie par morceaux en utilisant la commande `piecewise`.

```
> restart;
  f:=x->piecewise(x>0,exp(-1/x^2),0);f(-1);f(1);
```

$$f := x \rightarrow piecewise\left(0 < x, \mathrm{e}^{-\frac{1}{x^2}}, 0\right)$$

$$0$$

$$\mathrm{e}^{-1}$$

La syntaxe complète de la commande est la suivante : `piecewise(condition1, f1, condition2, f2, ..., conditionN, fN, f)` où les conditions sont des tests booléens du type :

1. Egalité $x=a$, qui s'écrit `a=b`.
2. Inégalité $x<a$ qui s'écrit `x<a`.
3. Inégalité $x>a$ qui s'écrit `x>a`.
4. Différent de $x \neq a$ qui s'écrit `x<>a`.
5. Appartenance à un intervalle $a < x < b$ qui s'écrit `x<b and x>a` ou `x>a and x<b`.

On lit donc : si la variable indépendante `x` vérifie la `condition 1`, alors la fonction adopte le mode de correspondance `f1`; si la variable indépendante vérifie la `condition 2`, alors la fonction est `f2`; ...; si la variable indépendante ne vérifie aucune des conditions 1 à N, alors la fonction est `f`.

```
> restart;
  f:=x->piecewise(x<0 and x>4,exp(-1/x^2),x=0,1,x>1 and
  x<2,cos(x),x^3+1);
```

$$f := x \rightarrow piecewise\left(x < 0 \textbf{ and } 4 < x, \mathrm{e}^{-\frac{1}{x^2}}, x = 0, 1, 1 < x \textbf{ and } x < 2, \cos(x), x^3 + 1\right)$$

```
> f(-5);f(0);f(1);f(1.5);f(2);f(100);
```

$$-124$$
$$1$$
$$2$$
$$0.07073720167$$
$$9$$
$$1000001$$

La commande `piecewise` accepte des tests portant sur des paramètres dûment définis par la commande `assume` :

```
> assume(a<b,b<c);
  piecewise(x>a and x< b,1,x>b and x<c,2,3);
```

$$\begin{cases} 1 & a\sim < x \textbf{ and } x < b\sim \\ 2 & b\sim < x \textbf{ and } x < c\sim \\ 3 & otherwise \end{cases}$$

Ensemble de définition

En principe, un ensemble de définition se cherche "à la main", mais dans certains cas il peut être utile de demander à Maple de faire le travail à l'aide de la commande `singular (expression)` qui renvoie - si possible - les points ou les sous-ensembles sur lesquels la fonction, écrite sous la forme d'une expression, n'est pas définie.

```
> restart;
  f1:=x->1/((x-1)*(x-2));
  singular(f1(x));#exemple 1
  f2:=x->log(x+1);
  singular(f2(x));#exemple 2
```

$$f1 := x \rightarrow \frac{1}{(x-1)\,(x-2)}$$
$$\{x = 1\}, \{x = 2\}$$
$$f2 := x \rightarrow \log(x+1)$$
$$\{x = -1\}, \{x = \infty\}, \{x = -\infty\}$$

L'exemple 1 n'offre aucune difficulté d'interprétation : les 2 valeurs "interdites" sont bien 1 et 2. Par contre, l'exemple 2 demande à être décodé : Maple signale que la valeur -1 est interdite mais pas les nombres strictement inférieurs à -1; à quoi s'ajoute mystérieusement $-\infty$ et $+\infty$ sans qu'on sache vraiment pourquoi. La commande `singular` n'est donc pas la panacée et ses résultats doivent être pris avec beaucoup de recul critique. En revanche, Maple est intraitable dans les calculs farfelus et signale son mécontentement par un message d'erreur.

```
> log(0);tan(Pi/2);
Error, (in ln) numeric exception: division by zero
Error, (in tan) numeric exception: division by zero
```

Il faut cependant faire attention au fait que le logiciel accepte les fonctions d'une variable complexe, de sorte que les économistes, familiers des nombres réels, risquent d'être surpris par certaines réponses. Ainsi, le logarithme népérien d'un nombre réel strictement négatif est calculé sans état d'âme :

```
> log(-15);
```

$$\ln(15) + \mathrm{I}\,\pi \tag{2.1}$$

Limites et continuité

La recherche de limites et l'étude de la continuité sont de grands classiques de l'analyse. Maple offre une gamme de commandes puissantes pour résoudre de tels problèmes, en particulier les formes indéterminées.

Limites

On utilise la commande **`limit(expression,x=valeur,option éventuelle)`**ou **`limit(fonction(x),x=valeur,options éventuelles)`**suivant qu'on travaille sur une expression ou une fonction. La valeur de *x* est finie ou infinie. L'option éventuelle mpajeure est **`right`** ou **`left`** suivant qu'on recherche une limite à droite ou à gauche du point considéré.

```
> restart;
  P:=(x^2-1)*ln((1+x)/(1-x));
  f:=unapply(P,x);
  limit(P,x=1);#calcul de limite en un point sur une expression
  limit(f(x),x=1);#limite d'une fonction en un point
```

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

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

$$0$$

$$0$$

```
> limit(P,x=1,right);  #limite à droite
  limit(P,x=1,left);  #limite à gauche
```

$$0$$

$$0$$

```
> Q:=(x^2+1)/(x^3-1);
  limit(Q,x=infinity);#limite quand x tend vers plus l'infini
  limit(Q,x=-infinity);#limite quand x tend vers moins l'infini
  limit(Q,x=infinity,real);#limite quand x tend vers l'infini en valeur absolue
```

$$Q := \frac{x^2+1}{x^3-1}$$
$$0$$
$$0$$
$$0$$

Quand l'expression ne tend pas vers une limite finie, Maple retourne le message **`undefined`**.

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

$$undefined$$

ou encore renvoie un intervalle en cas d'oscillations :

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

$$-1..1$$

ou renvoie l'expression non évalué en cas d'échec ou d'impossibilité :

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

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

Enfin, signalons qu'on peut profiter du fait que Maple est un logiciel de calcul formel en lui demandant des calculs de limite faisant intervenir des paramètres en sus de la variable.

```
> restart;
  F1:=(a*x+b)/(c*x+d);
  limit(F1,x=infinity);
```

$$F1 := \frac{a\,x+b}{c\,x+d}$$
$$\frac{a}{c}$$

Continuité

L'étude de la continuité et des discontinuités s'effectue à l'aide de calculs de limite. Une représentation graphique (voir la section suivante), même "légère", permet de se faire une idée des problèmes posés par une fonction.
Il est cependant bon de savoir que Maple propose un test de continuité d'une fonction grâce à la commande **`iscont(expression,x=a..b,option éventuelle)`**, où le test est mené sur l'intervalle (a,b), a étant un nombre réel mais pouvant prendre la valeur $-\infty$ et b étant un nombre réel plus grand que a mais pouvant prendre la valeur ∞. L'option est **`'closed'`** si l'intervalle est fermé, **`'open'`** si l'intervalle est ouvert.

```
> restart;
  f1:=x->x^2+3*x+4;
  f2:=x->1/(x-1);
  iscont(f1(x),x=-infinity..infinity);
  iscont(f2(x),x=-5..1,'closed');
```

$$f1 := x \to x^2+3\,x+4$$
$$f2 := x \to \frac{1}{x-1}$$
$$true$$
$$false$$

Graphe d'une fonction

La commande **plot** permet de tracer le graphe d'une fonction numérique si celle-ci est définie sans ambiguité de calcul explicite, en clair si elle ne comprend qu'une variable et ne fait pas intervenir de paramètres.

Représentation graphique d'une fonction

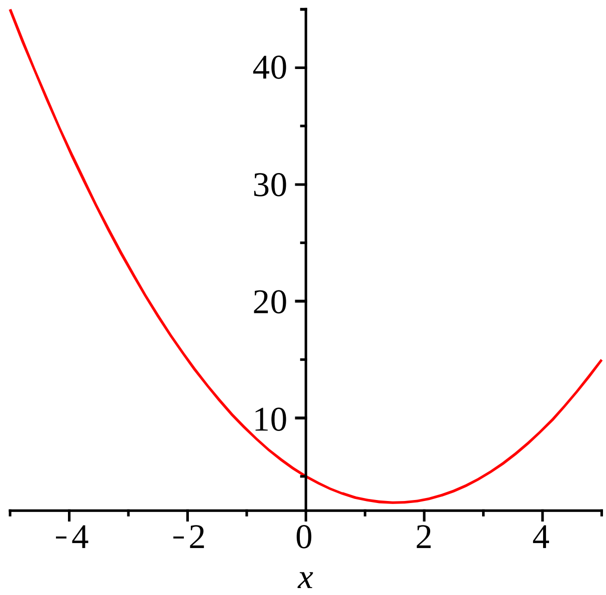
Deux cas se présentent, suivant que la fonction mathématique a été exprimée par une expression ou par une fonction.

A partir d'une expression

Une expression étant définie, on peut tracer la courbe représentative sur l'intervalle [*a*,*b*] (avec *a*<*b*) au moyen de la commande **plot(expression,x=a..b)**

```
> restart;
  F1:=x^2-3*x+5;
  plot(F1,x=-5..5);
```

$$F1 := x^2 - 3\,x + 5$$

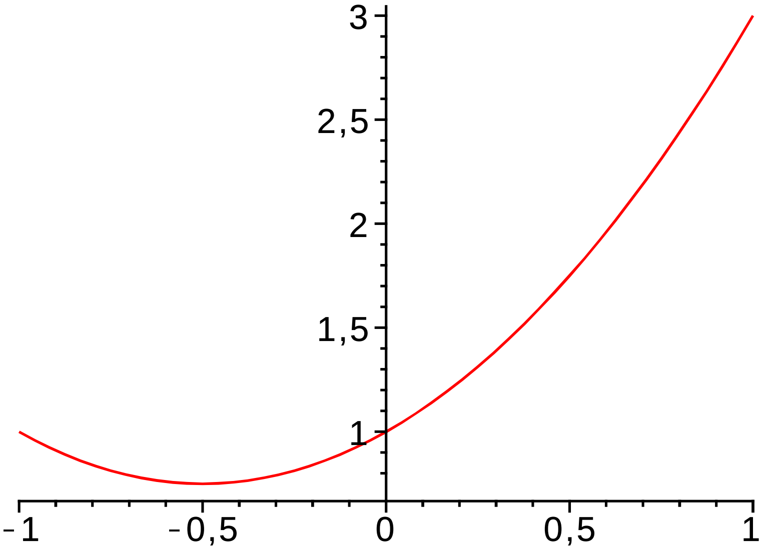


A partir d'une fonction

Une fonction **f** étant définie, on utilise la commande **plof(f,a..b)** où la variable **x** ne figure pas.

```
> restart;f:=x->x^2+x+1;plot(f,-1..1);
```

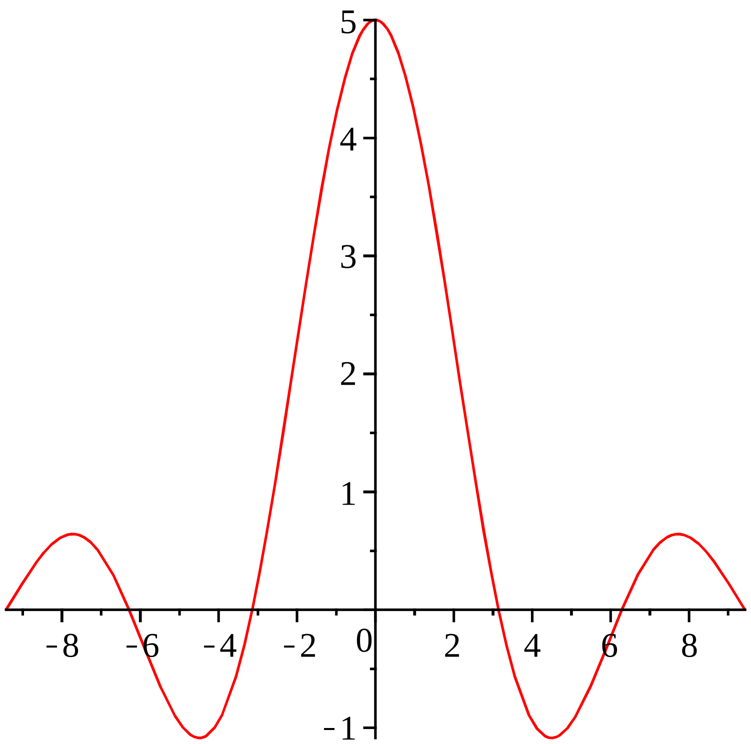
$$f := x \rightarrow x^2 + x + 1$$



Autre exemple, faisant intervenir une fonction prolongée par continuité.

```
> restart;
  f:=x->piecewise(x=0,5,5*sin(x)/x);
  plot(f,-3*Pi..3*Pi);
```

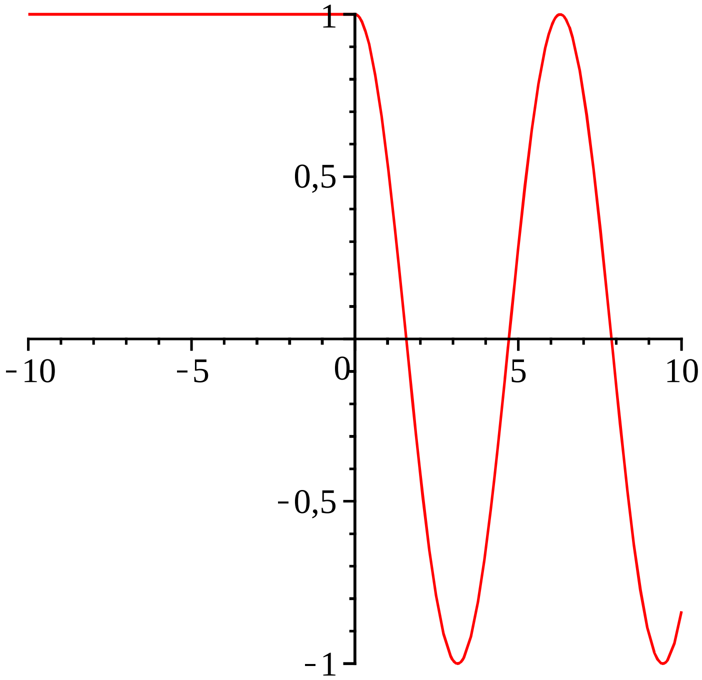
$$f := x \rightarrow piecewise\left(x = 0, 5, \frac{5 \sin(x)}{x}\right)$$



Dernier exemple, tracé d'une fonction définie par morceaux :

```
> restart;
  f:=x->piecewise(x<0,1,cos(x));
  plot(f,-10..10);
```

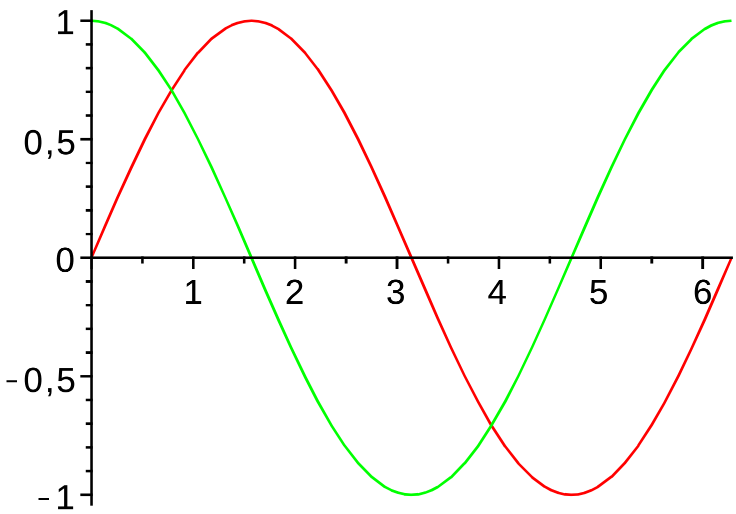
$$f := x \rightarrow piecewise(x < 0, 1, \cos(x))$$

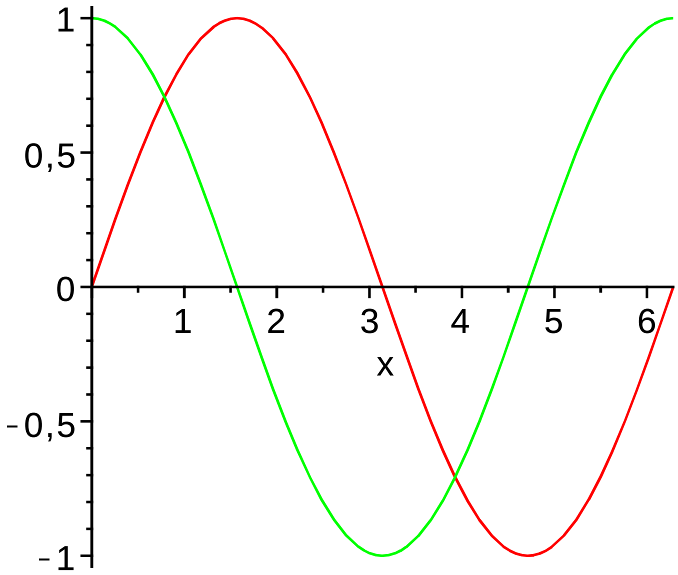


Réunion de plusieurs graphes dans la même figure

Il est aisé d'obtenir plusieurs courbes dans la même figure par la commande `plot({f1, f2,..,fN},a..b)` où les `f1,f2,..,fN` représentent les fonctions à tracer, qui sont alors mises entre accolades, ou par la commande `plot({expression1(x), expression2(x),..,expressionN(x)},x=a..b)` où les expressions à tracer sont alors mises entre accolades. Maple attribue automatiquement des couleurs différentes aux courbes.

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

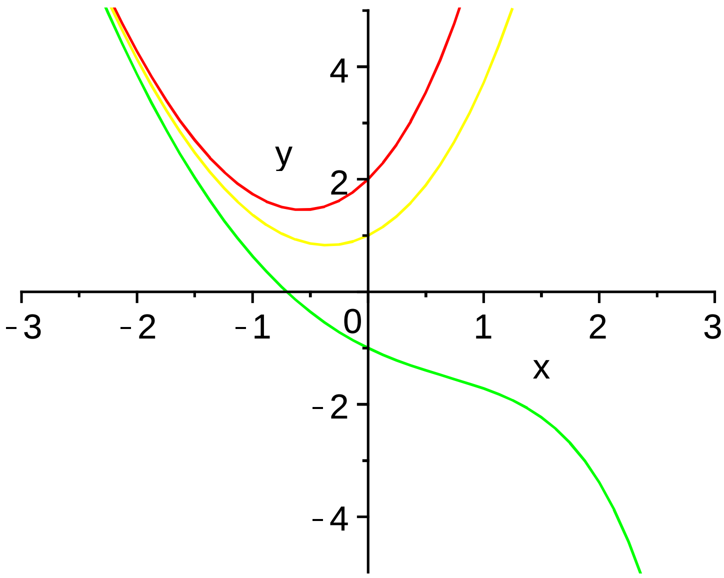




Pour tracer une famille de courbes dépendant d'un paramètre, la façon la plus rapide de travailler est de définir au préalable une fonction qui associe au paramètre a une expression qui dépend de a et de x.Ensuite, on demande le tracé de plusieurs courbes en attribuant des valeurs précises au paramètre.

```
> restart;p:=a->a*exp(x)+x^2;plot({p(-1),p(1),p(2)},x=-3..3,y=-5..5);
```

$$p := a \rightarrow a\, e^x + x^2$$



On expliquera dans un chapitre ultérieur comment construire des graphiques "parlants" à l'aide des nombreuses options de la commande `plot` et des ressources du package `plots`.

Exercices

Exercices mathématiques

Exercice M.1

Représentez dans le même graphique les fonctions définies par $f(x) = \ln(x)$ et $g(x) = \ln(-x)$. Vérifier que la commande `plot` tient compte de leur domaine de définition respectif.

Exercice M.2

Représentez dans le même graphique les trois fonctions définies par $f(x) = e^{x}$, $g(x) = e^{-x}$ et $h(x) = \frac{e^{x} + e^{-x}}{2}$. Que pensez-vous de la sortie graphique?

Exercice M.3

Représentez dans le même graphique les fonctions définies par $f(x) = tg(x)$ et $g(x) = Arctg(x)$. Améliorez le résultat graphique en retenant un intervalle significatif de $\mathbb{R}$.

Exercices économiques

Exercice E.1 (Fonction d'utilité)

Représentez dans le même graphique la fonction d'utilité logarithmique $u1(c) = \ln(c)$ et la fonction CIES (à élasticité de substitution intertemporelle constante) $u2(c) = \frac{c^{\frac{1}{4}}}{1 - \frac{1}{4}}$.

Vérifient-elles les propriétés usuelles d'une fonction d'utilié?

Exercice E.2 (Fonction de demande)

La demande d'un bien Q dépend de son prix P, du prix d'un bien substitut Ps et du revenu des ménages Y selon la relation $Q = -30\,P + 2\,Ps + 0.05\,Y$.

1. Créer trois fonction de demande **f1**, **f2** et **f3**, dépendant respectivement du prix du bien, du prix du bien substitut et du revenu.
2. Quel est l'effet d'une augmentation du prix du bien substitut sur la courbe de **f1**?
3. Quel est l'effet d'une augmentation du revenu sur la courbe de **f1**?
4. Quel est l'effet d'une augmentation du prix sur la courbe de **f2**?
5. Quel est l'effet d'une augmentation du revenu sur la courbe de **f2**?
6. Quel est l'effet d'une augmentation du prix sur la courbe de **f3**?
7. Quel est l'effet d'une augmentation du prix du bien substitut sur la courbe de **f3**?

Pour les questions 2 à 7, on devra attribuer des valeurs aux variables exogènes puis faire un graphe avec trois courbes représentatives.

Exercice E.3 (Fonction de demande)

La demande d'un bien Q dépend de son prix P selon la relation affine $Q = -0.3\,P + 50$. Suite à un changement du goût des consommateurs, la relation devient $Q = -0.4\,P + 50$. Visualisez l'effet de ce changement sur la courbe de demande.

Exercice E.4 (Equilibre sur un marché)

La demande d'un bien est donnée par $D = -a\,P + b$ et l'offre par $Q = c\,P + d$. Tous les paramètres a, b, c et d sont strictement positifs et $b > d$. Représentez graphiquement l'équilibre du marché.