

Pour télécharger le logiciel Geogebra, il faut aller sur le site : <http://www.geogebra.org>






1 Les suites (listes et séquences)

- **L={A,B,C}** : définit une liste contenant trois points A, B, et C créés auparavant.
- **L={(0,0),(1,1),(2,2)}** : définit une liste contenant les points définis, bien qu'ils n'aient pas été nommés.
- **Longueur[liste L]** : Longueur de la liste L (nombre d'éléments de la liste).
- **Elément[L, n]** : n^eélément de la liste L
- **Min[L]** : Plus petit élément de la liste L
- **Max[L]** : Plus grand élément de la liste L
- **Séquence[e, i, a, b]** : Liste des objets créés en utilisant l'expression e et l'indice i variant du nombre a au nombre b . (Se traduit par : de $i = a$ à $i = b$ calculer la valeur de e).
Exemple : $L = \text{Séquence}[(2, i), i, 1, 5]$ crée une liste de 5 points dont l'ordonnée varie de 1 à 5.
- **Séquence[e, i, a, b, s]** : Liste des objets créés en utilisant l'expression e et l'indice i variant du nombre a au nombre b avec un pas de s .
Exemple : $L = \text{Séquence}[(2, i), i, 1, 5, 0.5]$ crée une liste de 9 points dont l'ordonnée varie de 1 à 5 avec un pas de 0.5.
- **ItérationListe[f, x₀, n]** : Liste L de longueur $n+1$ dont les éléments sont les images itératives par la fonction f de la valeur x_0 .
Exemple : la commande $L = \text{ItérationListe}[x^2, 3, 2]$ vous donne la liste
 $L = \{3, 3^2, (3^2)^2\} = \{3, 9, 81\}$.
- **Définir une suite par sa formule générale $u_n = f(n)$**
Par exemple $u_n = 3n + 1$
→ On utilise **Séquence[3n+1, n, 0, 10]** pour obtenir les 10 premiers termes de la suite

→ **Séquence[(n,3n+1), n, 0, 10]** pour obtenir sa représentation graphique.

- **Définir une suite par la formule générale $u_{n+1} = f(u_n)$**
Par exemple $u_{n+1} = 2u_n + 1$ avec $u_0 = 1$.
On écrira donc
→ **u0=1** (juste pour avoir plus de clarté dans les formules, ce n'est pas nécessaire)
→ **L=Itération[2*x+1,u0,10]** pour avoir u_{10} .
→ **Séquence[Itération[2*x+1,u0,i],i,0,9]** pour obtenir la liste des valeurs jusqu'à u_9 .
→ **Séquence[(i,Itération[2*x+1,u0,i]),i,0,9]** Pour obtenir la représentation graphique des 10 premiers

2 Quelques icônes importants

-  Permet de définir une variable ou un paramètre qui appartient à un intervalle et que l'on pourra faire varier avec la souris.
-  Permet de tracer le lieu (la trace) d'un point dépendant d'un autre objet que l'on pourra faire varier ou déplacer.
-  Permet de calculer l'aire d'un polygone.
-  Permet de définir et de tracer les points d'intersection entre deux objets que l'on sélectionne avec la souris.
-  Permet de comparer deux objets que l'on sélectionne avec la souris.

3 Quelques fonctions de base

- **abs(x)** : Valeur absolue de x .
- **sgn(x)** : Renvoie $\frac{x}{|x|}$ pour avoir le signe de x .

- **sqrt(x)** : Renvoie la racine carrée de x .
- **exp(x)** : Renvoie l'exponentielle de x .
- **log(x)** : Renvoie le logarithme népérien de x .
- **lg(x)** : Renvoie le logarithme décimal de x .
- **ld(x)** : Renvoie le logarithme en base 2 de x .
- **cos(x)** : Renvoie le cosinus de x .
- **sin(x)** : Renvoie le sinus de x .
- **tan(x)** : Renvoie la tangente de x .
- **acos(x)** : Renvoie arc cosinus de x .
- **asin(x)** : Renvoie arc sinus de x .
- **atan(x)** : Renvoie arc tangente de x .
- **cosh(x)** : Renvoie le cosinus hyperbolique de x .
- **sinh(x)** : Renvoie le sinus hyperbolique de x .
- **tanh(x)** : Renvoie la tangente hyperbolique de x .
- **acosh(x)** : Renvoie arc cosinus hyperbolique de x .
- **asinh(x)** : Renvoie arc sinus hyperbolique de x .
- **atanh(x)** : Renvoie arc tangente hyperbolique de x .
- **floor(x)** : Renvoie le plus grand entier inférieur ou =.
- **ceil(x)** : Renvoie le plus petit entier supérieur =.
- **round(x)** : Renvoie l'arrondi à l'unité de x .
- **x(A)** : Renvoie l'abscisse de A .
- **y(A)** : Renvoie l'ordonnée de A .
- **cbrt(x)** : Renvoie la racine cubique de x .
- **random()** : Renvoie un nombre aléatoire entre 0 et 1.
- **gamma(x)** : Renvoie l'image de x par la fonction gamma.
- **x!** : Renvoie factorielle de x .

4 Les Fonctions

- **f(x) = 3 * x^2 + 5** : Définit la fonction f qui à x associe $3x^2 + 5$ et trace sa représentation graphique.
- **Fonction[f,a,b]** : Trace C_f entre a et b .

- **Si**[C_1, f, g] : Renvoie f si condition C_1 sinon renvoie g . Permet de définir des fonctions par morceaux.
- **PointInflexion** [f] : Tous les points d'inflexion de la fonction f .
- **Extremum**[f] : Tous les extremums locaux de la fonction f .
- **$g(x)=f(x+a)$** : Définit g comme la fonction qui à x associe $f(x+a)$ et trace \mathcal{C}_g .
- **$g(x)=f(x)+a$** : Définit g comme la fonction $f+a$ et trace \mathcal{C}_g .
- **$g(x)=af(x)+b$** : Définit g comme la fonction $af+b$ et trace \mathcal{C}_g .
- **$g(x)=f(x)+h(x)$** : Définit g comme la fonction $f+h$ et trace \mathcal{C}_g .
- **$g(x)=f(x)-h(x)$** : Définit g comme la fonction $f-h$ et trace \mathcal{C}_g .
- **$g(x)=f(x)*h(x)$** : Définit g comme la fonction $f \times h$ et trace \mathcal{C}_g .
- **$g(x)=f(x)/h(x)$** : Définit g comme la fonction $\frac{f}{h}$ et trace \mathcal{C}_g .
- **$g(x)=f(x)^n$** : Définit g comme la fonction f^n et trace \mathcal{C}_g .
- **Translation**[f, v] : Translate \mathcal{C}_f par la translation de vecteur \vec{v} .
- **Itération**[f, x_0, n] : compose n fois l'image du nombre de départ x_0 par la fonction f .

5 Les équations

- **Racine** [f, a] : Une racine de f à partir de a (par la méthode de Newton).
- **Racine** [f, a, b] : Une racine de f sur $[a; b]$ (par la méthode de fausse position).
- **Racine**[f] : Toutes les racines de la fonction f .

6 Les fonctions dérivées

- **Dérivée**[f] ou **$f'(x)$** : Définit et trace la fonction dérivée de f .

- **$f'(x)$** : Définit et trace la fonction dérivée seconde de f .
- **Dérivée**[f, n] : Définit et trace la fonction dérivée n^e de f .

7 Intégrales et primitives

- **Intégrale**[f, a, b] : Renvoie le résultat de l'intégrale de f entre a et b et colorie l'aire entre \mathcal{C}_f , l'axe des abscisses et les droites $x=a$ et $x=b$.
- **Intégrale**[f, g, a, b] : Renvoie le résultat de l'intégrale de $f-g$ entre a et b et colorie l'aire entre $\mathcal{C}_f, \mathcal{C}_g$, l'axe des abscisses et les droites $x=a$ et $x=b$.
- **Intégrale**[f] : Renvoie une primitive de f .
- **SommeInférieure**[f, a, b, n] : Approximation inférieure de l'intégrale de f sur l'intervalle $[a; b]$ par n .
- **rectangles. Note** : Cette commande dessine aussi les rectangles.
- **SommeSupérieure**[f, a, b, n] : Approximation supérieure de l'intégrale de f sur l'intervalle $[a; b]$ par n .
- **rectangles. Note** : Cette commande dessine aussi les rectangles.

8 Quelques longueurs

- **Longueur**[f, x_1, x_2] : Longueur de la portion de la courbe de la fonction f entre ses points d'abscisses x_1 et x_2 .
- **Longueur**[f, A, B] : Longueur de la portion de la courbe de la fonction f entre deux de ses points A et B .
- **Longueur**[c, t_1, t_2] : Longueur de la courbe c entre les deux points de paramètres t_1 et t_2 .
- **Longueur**[c, A, B] : Longueur de la courbe c entre deux de ses points A et B .

9 Les Tangentes

- **Tangente**[a, f] : Tangente à \mathcal{C}_f en $x=a$.

- **Tangente**[A, f] : Tangente à \mathcal{C}_f en $x=x(A)$.
- **Tangente**[A, c] : Tangente à la courbe c au point A .

10 Les Polynômes

- **Polynôme**[f] : Renvoie l'écriture polynomiale développée de la fonction f .
- **PolynômeTaylor**[f, a, n] : Renvoie le développement de Taylor de la fonction f à partir du point $x=a$ d'ordre n .
- **Racine**[f] : Toutes les racines du polynôme f .
- **Extremum**[f] : Tous les extremums locaux du polynôme f .
- **PointInflexion** [f] : Tous les points d'inflexion du polynôme f .

11 Les intersections

- **Intersection**[f_1, f_2] : Tous les points d'intersection entre les courbes \mathcal{C}_{f_1} et \mathcal{C}_{f_2} des polynômes f_1 et f_2 .
- **Intersection**[f_1, f_2, n] : même point d'intersection entre les courbes \mathcal{C}_{f_1} et \mathcal{C}_{f_2} des polynômes f_1 et f_2 .
- **Intersection**[f, g, A] : Premier point d'intersection entre \mathcal{C}_f et \mathcal{C}_g à partir de A (par la méthode de Newton).

12 Les fonctions d'arithmétique

- **Reste**[a, b] : Reste de la division euclidienne du nombre a par le nombre b .
- **Quotient**[a, b] : Quotient de la division euclidienne du nombre a par le nombre b .
- **Min**[a, b] : Minimum des deux nombres a et b .
- **Max**[a, b] : Maximum des deux nombres a et b .

13 Courbes paramétrées

- **Courbe**[e_1, e_2, t, a, b] : Courbe paramétrée de paramètre t variant dans l'intervalle $[a; b]$ l'abscisse d'un point étant expression e_1 et son ordonnée expression e_2 .

Exemple : $c = \text{Courbe}[2\cos(t), 2\sin(t), t, 0, 2\pi]$

- **Dérivée[c]** : Dérivée de la courbe c .
- **Valider c(3)** : retourne le point de la courbe c dont la position correspond à la valeur 3 du paramètre.

1 Transformations géométriques

- **Translation**[A, v] : Translaté du point A de vecteur v .
- **Translation**[g, v] : Translaté de la ligne g de vecteur v .
- **Translation**[c, v] : Translatée de la conique c de vecteur v .
- **Translation**[f, v] : Translatée de la courbe de la fonction f de vecteur v .
- **Translation**[$poly, v$] : Translation du polygone $poly$ de vecteur v .
- **Translation**[pic, v] : Translation de l'image pic de vecteur v .
- **Translation**[v, P] : Donne au vecteur v le point P comme origine.
- **Rotation**[A, ϕ] : Tourne le point A d'un angle ϕ autour de l'origine.
- **Rotation**[v, ϕ] : Tourne le vecteur v d'un angle ϕ .
- **Rotation**[g, ϕ] : Tourne la ligne g d'un angle ϕ autour de l'origine.
- **Rotation**[c, ϕ] : Tourne la conique c d'un angle ϕ autour de l'origine.
- **Rotation**[$poly, \phi$] : Tourne le polygone $poly$ d'un angle ϕ autour de l'origine.
- **Rotation**[pic, ϕ] : Tourne l'image pic d'un angle ϕ autour de l'origine.
- **Rotation**[A, ϕ, B] : Tourne le point A d'un angle ϕ autour du point B .
- **Rotation**[g, ϕ, B] : Tourne la ligne g d'un angle ϕ autour du point B .
- **Rotation**[c, ϕ, B] : Tourne la conique c d'un angle ϕ autour du point B .
- **Rotation**[$poly, \phi, B$] : Tourne le polygone $poly$ d'un angle ϕ autour du point B .
- **Rotation**[pic, ϕ, B] : Tourne l'image pic d'un angle ϕ autour du point B .

2 Les coniques

- **Symétrie**[A, B] : Symétrique du point A par rapport au point B .
- **Symétrie**[g, B] : Symétrie de la ligne g par rapport au point B .
- **Symétrie**[c, B] : Symétrie de la conique c par rapport à B .
- **Symétrie**[$poly, B$] : Symétrie du polygone $poly$ par rapport au point B .
- **Symétrie**[pic, B] : Symétrie de l'image pic par rapport à B .
- **Symétrie**[A, h] : Symétrie du point A par rapport à la ligne h .
- **Symétrie**[g, h] : Symétrie de la ligne g par rapport à la ligne h .
- **Symétrie**[c, h] : Symétrie de la conique c par rapport à h .
- **Symétrie**[$poly, h$] : Symétrie du polygone $poly$ par rapport à la ligne h .
- **Symétrie**[pic, h] : Symétrie de l'image pic par rapport à h .
- **Homothétie**[A, f, S] : Image du point A par l'homothétie de centre S , de rapport f .
- **Homothétie**[h, f, S] : Image de la ligne h par l'homothétie de centre S , de rapport f .
- **Homothétie**[c, f, S] : Image de la conique c par l'homothétie de centre S , de rapport f .
- **Homothétie**[$poly, f, S$] : Image du polygone $poly$ par l'homothétie de centre S , de rapport f .
- **Homothétie**[pic, f, S] : Transformée de l'image pic par l'homothétie de centre S , de rapport f .
- **Ellipse**[F, G, a] : Ellipse de foyers F et G et dont la longueur de l'axe principal vaut a . Note : Condition : $2a > Distance[F, G]$.
- **Ellipse**[F, G, s] : Ellipse de foyers F et G et dont la longueur de l'axe principal vaut $a = Longueur[s]$.

- **Hyperbole**[F, G, a] : Hyperbole de foyers F et G dont la longueur de l'axe principal vaut a . Note : Condition : $0 < 2a < Distance[F, G]$.
- **Hyperbole**[F, G, s] : Hyperbole avec foyers F et G dont la longueur de l'axe principal vaut $a = Longueur[s]$.
- **Parabole**[F, g] : Parabole de foyer F et de directrice g .
- **Conique**[A, B, C, D, E] : Conique passant par les cinq points A, B, C, D , et E . Note : Quatre de ces points ne doivent pas être alignés.

3 Les angles

- **Angle**[$v1, v2$] : Angle entre deux vecteurs $v1$ et $v2$ (entre 0 et 360°).
- **Angle**[g, h] : Angle entre les vecteurs directeurs de deux lignes g et h (entre 0 et 360°).
- **Angle**[A, B, C] : Angle \widehat{ABC} , délimité par $[AB]$ et $[BC]$ (entre 0 et 360°). B représente donc le sommet de l'angle.
- **Angle**[A, B, α] : Dessine un angle α à partir de B avec pour sommet B .
- **Angle**[c] : Angle de l'axe principal de la conique c par rapport à l'horizontale.
- **Angle**[v] : Angle entre l'axe (Ox) et le vecteur v .
- **Angle**[A] : Angle entre l'axe (Ox) et le vecteur \overrightarrow{OA} .
- **Angle**[n] : Convertit un nombre en un angle (le résultat entre 0 et 2π).
- **Angle**[$poly$] : Tous les angles intérieurs du polygone direct $poly$.

4 Quelques points en géométrie

- **A=(a,b)** : Définit et place le pt A de coordonnées $(a; b)$.
- **Point**[g] : Point libre sur la ligne g .
- **Point**[c] : Point libre sur la conique c (par ex. cercle, ellipse, hyperbole).

- **Point[f]** : Point libre sur la courbe représentative de la fonction f .
- **Point[poly]** : Point libre sur la ligne polygonale frontière de poly.
- **Point[P, v]** : Image du point P dans la translation de vecteur v .
- **MilieuCentre[A,B]** : Milieu des points A et B .
- **MilieuCentre[s]** : Milieu du segment s .
- **CentreGravité[poly]** : Centre de gravité du polygone poly.
- **Intersection[g,h]** : Point d'intersection entre les lignes g et h .
- **Intersection[g,c]** : Tous les points d'intersection de la ligne g avec la conique c (max. 2).
- **Intersection[g, c, n]** : nème point d'intersection de la ligne g avec la conique c .
- **Intersection[c1, c2]** : Tous les points d'intersection entre les coniques c_1 et c_2 (max. 4).
- **Intersection[c1, c2, n]** : nème point d'intersection entre les coniques c_1 et c_2 .
- **Intersection[f1, f2]** : Tous les points d'intersection entre les courbes \mathcal{C}_{f_1} et \mathcal{C}_{f_2} des polynômes f_1 et f_2 .
- **Intersection[f1, f2, n]** : nème point d'intersection entre les courbes \mathcal{C}_{f_1} et \mathcal{C}_{f_2} des polynômes f_1 et f_2 .
- **Intersection[f, g]** : Tous les points d'intersection entre la courbe \mathcal{C}_f du polynôme f et la ligne g .
- **Intersection[f, g, n]** : nème point d'intersection entre la courbe \mathcal{C}_f du polynôme f et la ligne g .
- **Intersection[f, g, A]** : Premier point d'intersection entre \mathcal{C}_f et \mathcal{C}_g à partir de A (par la méthode de Newton).
- **Intersection[f, g, A]** : Premier point d'intersection entre \mathcal{C}_f et la ligne g à partir de A (par la méthode de Newton).

5 Les vecteurs

- **Vecteur[A,B]** : Vecteur AB .
- **Vecteur[A]** : Vecteur OA .
- **Direction[g]** : Vecteur directeur de la ligne g .
- **VecteurUnitaire[g]** : Vecteur directeur unitaire de la ligne g .
- **VecteurUnitaire[v]** : Vecteur unitaire de même direction et même sens que le vecteur donné v .
- **VecteurOrthogonal[g]** : Vecteur orthogonal à la ligne g .
- **VecteurOrthogonal[v]** : Vecteur orthogonal au vecteur v .
- **VecteurUnitaireOrthogonal[g]** : Vecteur orthogonal unitaire à la ligne g .
- **VecteurUnitaireOrthogonal[v]** : Vecteur orthogonal unitaire au vecteur v .
- **VecteurCourbure[A, f]** : Vecteur de courbure de la courbe représentative de la fonction f au point A .
- **VecteurCourbure[A, c]** : Vecteur de courbure de la courbe c au point A .
- **u*v** : Produit scalaire $\vec{u} \cdot \vec{v}$.

6 Objets courants de géométrie

- **Segment[A, B]** : Segment $[AB]$.
- **Segment[A, a]** : Segment d'origine le point A et de longueur a .
- **DemiDroite[A, B]** : Demi-droite $[AB)$.
- **DemiDroite[A, v]** : Demi-droite d'origine A et de vecteur directeur v .
- **Polygone[A, B, C, ...]** : Polygone défini par les points donnés A, B, C, \dots
- **Polygone[A, B, n]** : Polygone régulier à n sommets (points A et B inclus)
- **Droite[A, B]** : Droite (AB) .
- **Droite[A, g]** : Droite passant par A et parallèle à la ligne g .

- **Droite[A, v]** : Droite passant par A et de vecteur directeur v .
- **Cercle[M, r]** : Cercle de centre M et de rayon r .
- **Cercle[M, s]** : Cercle de centre M et de $rayon = Longueur[s]$.
- **Cercle[M, A]** : Cercle de centre M passant par A .
- **Cercle[A, B, C]** : Cercle circonscrit à ABC (i.e. cercle passant par A, B et C).
- **DemiCercle[A, B]** : Demi-cercle de diamètre le segment $[AB]$.
- **ArcCercle[M, A, B]** : Arc de cercle de centre M entre les deux points A et B .
- **ArcCercleCirconscrit[A, B, C]** : Arc de cercle passant par les trois points A, B , et C .
- **Arc[c, A, B]** : Arc entre les deux points A et B de la conique c (Cercle ou Ellipse).
- **SecteurCirculaire[M, A, B]** : Secteur circulaire de centre M entre les deux points A et B .
- **SecteurCirculaireCirconscrit[A, B, C]** : Secteur circulaire passant par les trois points A, B , et C .

7 Droites particulières

- **axeX** : Axe des abscisses.
- **axeY** : Axe des ordonnées.
- **Perpendiculaire[point A, ligne g]** : Droite passant par A et perpendiculaire à la ligne g .
- **Perpendiculaire[point A, vecteur v]** : Droite passant par A et orthogonale au vecteur v .
- **Médiatrice[point A, point B]** : Médiatrice du segment $[AB]$.
- **Médiatrice[segment s]** : Médiatrice du segment s .
- **Bissectrice[A, B, C]** : Bissectrice de l'angle .
- **Bissectrice[g, h]** : Les deux bissectrices des lignes g et h .

- **Tangente[A, c]** : (Toutes les) tangentes à c passant par A .
- **Tangente[g, c]** : Toutes les tangentes à c parallèles à g .
- **Tangente[a, f]** : Tangente à \mathcal{C}_f en $x = a$.
- **Tangente[A, f]** : Tangente à \mathcal{C}_f en $x = x(A)$.
- **Tangente[A, c]** : Tangente à la courbe c au point A .
- **Asymptote[h]** : Les deux asymptotes à l'hyperbole h .
- **Directrice[p]** : Directrice de la parabole p .
- **Axes[c]** : Les deux axes de la conique c .
- **PremierAxe[c]** : Axe principal de la conique c .
- **SecondAxe[c]** : Axe secondaire de la conique c .
- **Polaire[A, c]** : Droite polaire de A par rapport à la conique c .
- **Diamètre[g, c]** : Diamètre de la conique c parallèle à g .
- **Diamètre[v, c]** : Diamètre de la conique c ayant pour vecteur directeur v .
- **Courbure[A, f]** : Courbure de la courbe représentative de f au point A .
- **Courbure[A, c]** : Courbure de la courbe c au point A .
- **Rayon[c]** : Rayon du cercle c .
- **Circonférence[c]** : Retourne la circonférence de la conique c (cercle ou ellipse).
- **Périmètre[poly]** : Périmètre du polygone $poly$.
- **Paramètre[p]** : Paramètre de la parabole p (distance entre la directrice et le foyer).
- **LongueurPremierAxe[c]** : Longueur du premier axe (axe principal) de la conique c .
- **LongueurSecondAxe[c]** : Longueur du second axe de la conique c .
- **ExcentricitéLinéaire[c]** : Excentricité linéaire de la conique (ellipse ou hyperbole) c (à savoir : la demi distance focale).
- **RapportColinéarité[A, B, C]** : Retourne le rapport de colinéarité de 3 points A, B , et C alignés, tel que $AC = \lambda \times AB$ ou $C = A + \lambda \times AB$
- **Birapport[A, B, C, D]** : Birapport de 4 points A, B, C , et D alignés, tel que :

$$\lambda = \frac{\text{RapportColinéarité}[C, B, A]}{\text{RapportColinéarité}[D, B, A]}$$

8 Les nombres de géométrie

- **Longueur[v]** : Norme du vecteur v .
- **Longueur[A]** : Distance OA .
- **Aire[A, B, C, ...]** : Aire du polygone défini par les points A, B , et $C \dots$
- **Aire[c]** : Aire délimitée par la conique c (cercle ou ellipse).
- **Distance[A, B]** : Distance AB .
- **Distance[A, g]** : Distance d'un point A à une ligne g .
- **Distance[g, h]** : Distance des lignes g et h .
- **Pente[g]** : Pente d'une ligne g . Note : Cette commande trace aussi le triangle permettant de visualiser la pente (quand j'avance de 1, je monte de « pente »).

1 Feuille de calcul

- « ; » exécute une commande en affichant le résultat. Par exemple `1+2/3` ;
- « \$ » exécute une commande sans afficher le résultat. Par exemple `a :2 $`
- « % » rappelle le dernier calcul effectué
- `? plot2d` affiche l'aide en ligne sur l'instruction `plot2d`
- `example(expand)` affiche des exemples d'utilisation de l'instruction `expand`
- `kill(all)` réinitialise le système

2 Opérateurs

- les quatre opérations usuelles `+`, `-`, `*`, `/`
- opérateur « ^ » élévation à une puissance. `x^3` est x^3
- opérateur « # » non égal à (ou différent de)
- opérateurs de comparaison `=`, `<`, `<=`, `>`, `>=`
- opérateur « : » d'affectation.
`a :3` donne la valeur 3 à la variable `a`.
- opérateur « := » pour définir une fonction.
- opérateur « = » indique une équation dans Maxima.
- opérateur « ! » factoriel d'un entier naturel, par exemple `5!` = $1 \times 2 \times 3 \times 4 \times 5 = 120$.
- opérateur « . » de multiplication de deux matrices.

3 Constantes

- `%pi` désigne $\pi \approx 3,14159$
- `%e` désigne $e = \exp(1) \approx 2,7183$
- `%i` est l'imaginaire pur de module 1, d'argument $\pi/2$
- `true` valeur "vrai"
- `false` valeur "faux"
- `inf` désigne $+\infty$
- `minf` désigne $-\infty$

- `%gamma` constante d'Euler-Mascheroni qui est la limite de la suite de terme général $\left(\sum_{k=1}^n \frac{1}{k}\right) - \ln n$

4 Nombres réels

a fonctions usuelles

- `abs(x)` valeur absolue de x
- `floor(x)` partie entière de x
- `sqrt(x)` racine carrée de x
- `sin(x)`, `cos(x)`, `tan(x)`
- `exp(x)`, `log(x)` *Attention* : `log` désigne la fonction **logarithme népérien**

b valeurs approchées

- `float(x)` fournit une valeur décimale approchée de x
- `bfloat(x)` donne une valeur approchée de x en notation scientifique
- `fpprec :20` fixe la précision de la valeur approchée donnée par `bfloat` (20 chiffres affichés au lieu de 16 par défaut)

c trigonométrie

- `acos(0.2)` donne la mesure en radian de l'angle géométrique ayant pour cosinus 0,2
- `trigexpand(a)` développe l'expression trigonométrique a en utilisant les formules d'addition de `cos` et `sin`. Par exemple, `trigexpand(cos(x+y))` renvoie $\cos x \cos y - \sin x \sin y$
- `trigreduce(a)` permet de linéariser un polynôme trigonométrique a . Par exemple, `trigreduce(sin(x)^3)` renvoie $\frac{3 \sin x - \sin(3x)}{4}$
- `trigsimp(a)` simplifie l'expression trigonométrique a en utilisant la relation $\cos^2 t + \sin^2 t = 1$ et en remplaçant $\tan t$ par $\frac{\sin t}{\cos t}$

- `load(ntrig)` Permet de charger le package permettant de calculer des lignes pour des valeurs de x non usuelles.

Exemple : Pour calculer $\cos \frac{\pi}{10}$ on fait :

`load(ntrig)`

`cos(%pi/10)` et on obtient $\frac{\sqrt{\sqrt{5}+5}}{2\sqrt{2}}$.

5 Arithmétique des entiers

Soit a et b deux entiers. Soit n et p deux entiers naturels.

- `divide(a,b)` division euclidienne de a par b . Le résultat est une liste dont le premier élément est le quotient et le second élément le reste
- `divisors(a)` ensemble des diviseurs positifs de a
- `divsum(a)` somme des diviseurs positifs de a
- `mod(a,b)` reste de la division de a par b
- `gcd(a,b)` pgcd de a et b
- `load(funcs) $ lcm(a,b)` ppcm de a et b
- `primep(p)` teste si p est premier
- `p :prev_prime(n)` donne le nombre premier p qui vient juste avant n , avec $p < n$
- `next_prime(n)` donne le nombre premier qui vient juste après n
- `factor(n)` décompose n en produit de facteurs premiers
- `ifactors(n)` décompose n en produit de facteurs premiers en affichant le résultat sous forme de liste
- `binomial(n,p)` est le coefficient binomial $\binom{n}{p}$
- `random(n)` renvoie un entier naturel, choisi au hasard entre 0 et $n-1$ lorsque $n \in \mathbb{N}^*$

6 Nombres complexes

Soit z un nombre complexe.

- `%i` désigne le complexe i

- `realpart(z)` partie réelle de z
- `imagpart(z)` partie imaginaire de z
- `conjugate(z)` conjugué de z
- `abs(z)` module de z
- `carg(z)` argument de z (dans $] -\pi, \pi]$)
- `rectform(z)` écrit z sous forme algébrique
- `polarform(z)` écrit z sous forme exponentielle

7 Calcul algébrique

Soit P et Q deux polynômes.

- `expand(P)` développe P
- `factor(P)` factorise P
- `gfactor(P)` factorise P dans l'ensemble \mathbb{C}
- `divide(P,Q,x)` calcule le quotient et le reste de la division de P par Q . Le résultat est une liste dont le premier élément est le quotient et le second élément le reste
- `partfrac(P/Q,x)` décompose la fonction rationnelle P/Q (de la variable x) en éléments simples
- `ratsimp(expr)` simplifie l'expression `expr` (en écrivant tout sur le même dénominateur)
- `subst(1/z,x,expr)` remplace x par $1/z$ dans l'expression `expr`

8 Fonctions numériques

a définir une fonction

- `f(x) :=x^2+2*x-3`
- `define(f(x),x^2+2*x-3)`
- `f :lambda([x],x^2+2*x-3)`

b limites, tangentes et asymptotes

- `limit(sin(x)/x,x,0)` limite en 0
- `limit(1/x,x,0,plus)` limite à droite en 0
- `limit(1/x,x,0,minus)` limite à gauche en 0

- `limit(x*exp(x),x,minf)` limite en $-\infty$
- `taylor(f(x),x,a,1)` permet d'obtenir l'équation réduite de la tangente à \mathcal{C}_f au point $A(a, f(a))$
- `taylor(sqrt(1+x^2),x,inf,2)` permet d'obtenir le développement asymptotique à 2 termes de $x \mapsto \sqrt{1+x^2}$ en $+\infty$

c dérivation

- `diff(f(x),x)` calcule la dérivée $f'(x)$
- `diff(f(x),x,2)` calcule $f''(x)$, dérivée seconde

d courbes représentatives

Pour afficher les courbes \mathcal{C}_f et \mathcal{C}_g sur le même graphique, dans la fenêtre $[x_1, x_2] \times [y_1, y_2]$, on entre :

- `plot2d([f(x),g(x)], [x,x1,x2], [y,y1,y2])`

e intégrales

- `integrate(f(x),x)` calcule une primitive de la fonction f
- `integrate(f(x),x,a,b)` calcule l'intégrale $\int_a^b f(x) dx$
- `romberg(1/log(x),x,2,3)` fournit une approximation de l'intégrale $\int_2^3 \frac{1}{\ln x} dx$

9 Suites

a Définition

- `u[n] :=1/n` est la suite définie par son terme général $u_n = \frac{1}{n}$
- `u[0] :1` on définit u_0 .
- `u[n] :=1/1+u[n-1]` est la suite définie par récurrence $u_{n+1} = \frac{1}{1+u_n}$
On ne peut pas écrire `u[n+1] :=1/1+u[n]`

b Calcul sur les suites

- `makelist(u[k], k, 0, 5)` ; Affiche les 6 premiers termes de la suite u_n .
- `sum(1/u[n],n,0,20)` pour calculer $S = u_0 + u_1 + \dots + u_{20}$.
- `limit(u[n],n,inf)` ; pour calculer la convergence de la suite u_n lorsque u_n est définie par sa formule générale.

10 Équations

a résolution d'équations

Résolution exacte dans l'ensemble \mathbb{C} des complexes :

- `solve(x^2+x=1,x)`
Résolution approchée dans \mathbb{R} :
- `find_root(x^5=1+x,x,1,2)` solution dans $[1, 2]$

b systèmes linéaires

Pour résoudre le système $\begin{cases} 3x + 2y = 1 \\ x - y = 2 \end{cases}$

- `S1 : [3*x+2*y=1,x-y=2]`
- `solve(S1, [x,y])`

c équations différentielles

Pour résoudre l'équation différentielle $y'' + w^2 y = \sin x$, on définit d'abord l'équation :

- `eqn : 'diff(y,x,2)+w^2*y=sin(x)`
On la résout :
- `sol :ode2(eqn,y,x)`
Pour trouver la solution satisfaisant aux conditions initiales $y(0) = 1$ et $y'(0) = -1$, on entre :
- `ic2(sol,x=0,y=1,diff(y,x)=-1)`
Pour trouver la solution satisfaisant aux conditions $y(0) = 1$ et $y(1) = 0$, on entre :
- `bc2(sol,x=0,y=1,x=1,y=0)`
- `rhs(sol)` saisit le membre de droite de l'égalité `sol` obtenue ci-dessus.

11 Listes

Une liste est un type de données, qui tient compte de l'ordre, accepte les répétitions d'éléments et est délimité par les caractères [et]. Voici quelques fonctions importantes concernant les listes :

- `L :makelist(k^2,k,0,9)` permet de créer la liste des carrés des 10 premiers naturels, k prenant toutes les valeurs entières de 0 jusqu'à 9.
- `L[2] :5` remplace le 2ème élément de la liste L par 5.
- `length(L)` donne le nombre d'éléments de la liste L .
- `first(L)` ; `second(L)` ; `last(L)` renvoient respectivement le premier, le second, le dernier élément de L .
- `member(x,L)` vaut `true` si x appartient à la liste L (`false` sinon).
- `append([a,1,3],[2,7])` regroupe les deux listes en une seule liste $[a,1,3,2,7]$.
- `join(l,m)` crée une nouvelle liste constituée des éléments des listes l et m , intercalés. La liste obtenue est $[l[1],m[1],l[2],m[2],l[3],m[3],\dots]$.
- `sort(L)` permet de ranger les éléments de la liste L par ordre croissant.
- `map(f,L)` permet d'appliquer la fonction f à tous les éléments de la liste L .

12 Sommation et produit

a somme finie

- `sum(1/k^2,k,1,10)` calcule la somme des inverses des carrés des entiers compris entre 1 et 10.

b produit fini

- `product(sqrt(k),k,1,10)` calcule le produit des racines carrées des entiers compris entre 1 et 10.

c somme infinie

On peut montrer que la suite $(u_n)_{n \in \mathbb{N}^*}$ de terme général $u_n = \sum_{k=1}^n \frac{1}{k^2}$ est convergente. Sa limite est notée $\sum_{k=0}^{+\infty} \frac{1}{k^2}$. On peut demander sa valeur exacte comme suit :

- `load(simplify_sum) $ sum(1/k^2,k,1,inf) $ simplify_sum(%)`

13 Vecteurs

- `u :[a,b,c]` Définit les coordonnées du vecteur u .
- `u+v` Renvoie les coordonnées de $u+v$
- `u.v` Renvoie le produit scalaire de u et de v (le point est précédé et suivi d'un espace)
- `load(vect)` Permet de charger le package pour calculer le produit vectoriel de deux vecteurs.
- `express(u ~ v)` : Renvoie le produit vectoriel de u et v .

14 Programmation

a syntaxe d'une procédure

```
nom(paramètres en entrée) := block( [variables locales],
<instruction 1>, <instruction 2>, ...
/* -----Commentaire----- */
)$
```

Voici un exemple simple de procédure qui additionne deux nombres.

- `somme(a,b) :=block([c], c :a+b, return(c))`

b structure conditionnelle

- `if (condition)
then (<instruction1> , <instruction2>)
else (<instruction3> , <instruction4>)`

c structures itératives

Boucle **For** et affichage de la table de 7 :

- `for k from 1 thru 10 do
(print("7 fois",k,"égale",7*k))`
Boucle **While** et affichage de la table de 7 :

- `k :1 $ while k<11 do
(print("7 fois",k,"égale",7*k) , k :k+1)`

15 Matrices

Soit B une matrice de taille 3×3 .

On définit la matrice $A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & -9 \end{pmatrix}$ ligne par ligne

de la façon suivante :

- `A :matrix([1,2,3],[4,5,6],[7,8,9])`
- `A+B` somme des matrices A et B
- `3*A` produit de la matrice A par le réel 3
- `A.B` produit des matrices A et B
- `A^^3` matrice A élevée à la puissance 3
- `invert(A)` inverse A^{-1} de la matrice A