Chapitre 6 (version alternative)

La géométrie dynamique avec GeoGebra

Tout comme la calculatrice TI-84Plus, le logiciel *GeoGebra*[1] a été conçu spécifiquement à des fins éducatives. C'est donc dire que la documentation et les divers exemples accompagnant ce logiciel sont particulièrement bien adaptés pour les professeurs de mathématiques, et nous vous recommandons de vous y référer au besoin[2]. Dans ce chapitre, nous nous contenterons de présenter quelques-unes des possibilités de *GeoGebra*, en les situant parfois dans une perspective plus globale.

6.1 Vue d'ensemble : figure, construction et dessin

La fenêtre obtenue à l'ouverture de *GeoGebra* contient un certain nombre de zones: selon les choix faits dans le menu « Affichage », nous pourrions faire apparaître ou non un système d'axes, une grille, une « Fenêtre Algèbre », un tableur, un champ de saisie, etc (voir la figure 6.1.1).

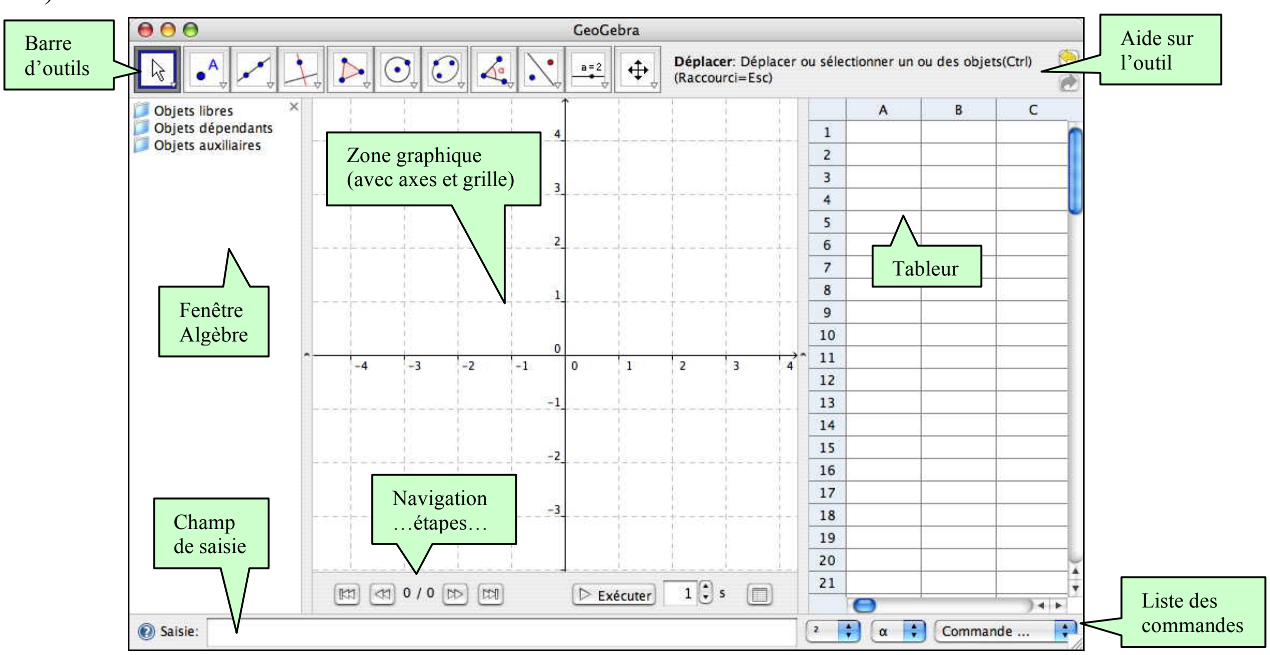
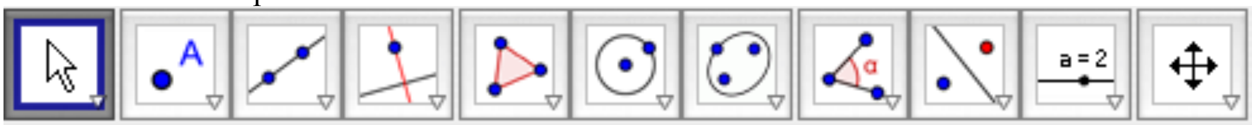


Figure 6.1.1 La fenêtre de GeoGebra et ses différentes zones.

[1] Dans ce chapitre, nous utiliserons la version 3.2.44 de *GeoGebra*, mais tout ce que nous ferons devrait rester valable dans les versions ultérieures.

[2] *GeoGebra* est un logiciel **libre**, disponible à l'adresse http://www.geogebra.org/. On pourra aussi trouver documentation et exemples à cette même adresse.

Nous constatons la présence d'une barre d'outils :



chaque icône cache en fait une liste d'outils, qu'on peut faire apparaître par un clic sur le petit triangle en bas à droite de l'icône. (Pour plus de détails, voir la figure 6.1.2). Soulignons que le choix d'un outil donné fait apparaître l'aide associée à cet outil dans la zone d'aide.

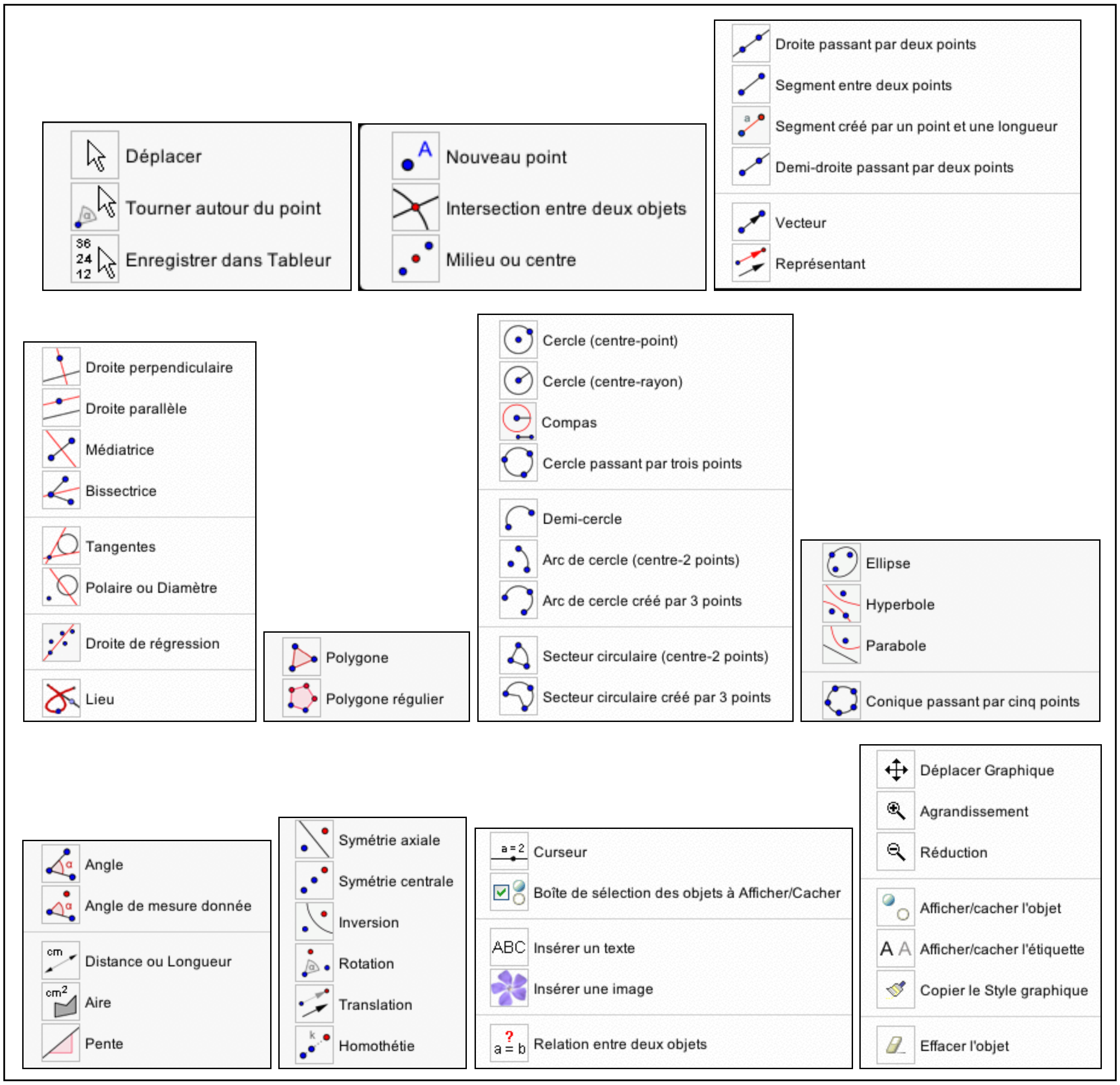


Figure 6.1.2 Courte description, colonne par colonne, des outils de GeoGebra.

Pour commencer à nous familiariser avec *GeoGebra*, construisons une figure simple : un rectangle. En passant en revue les outils disponibles, nous constatons l'absence d'un outil pour tracer directement un rectangle. En fait, *GeoGebra* effectue des constructions géométriques s'inspirant des outils classiques que sont la règle et le compas. Dans ce contexte, une façon de tracer un rectangle sera de suivre les étapes suivantes :

- Après avoir choisi l'outil « Segment entre deux points », nous désignons par un clic la première extrémité, repositionnons le curseur-souris, puis désignons par un second clic la deuxième extrémité. Selon la configuration (voir le menu « Options » ➤ « Etiquetage »), GeoGebra assignera automatiquement un nom à chaque objet créé. Si nous désirons modifier ce nom, il suffira de taper le nom voulu au clavier immédiatement après la création de l'objet : dans notre cas, nous nommerons *A* et *B* les deux extrémités du segment. Notons au passage le fonctionnement de *GeoGebra* : un segment est créé suite à deux clics (désignant deux points) et non via un mouvement de « glisser » de la souris.
- Si désiré, l'outil « Déplacer » nous permet de sélectionner et de déplacer les divers éléments de notre figure (points, segment, noms). On revient tellement souvent à cet outil qu'il existe un raccourci clavier pour y accéder : la touche « escape ».

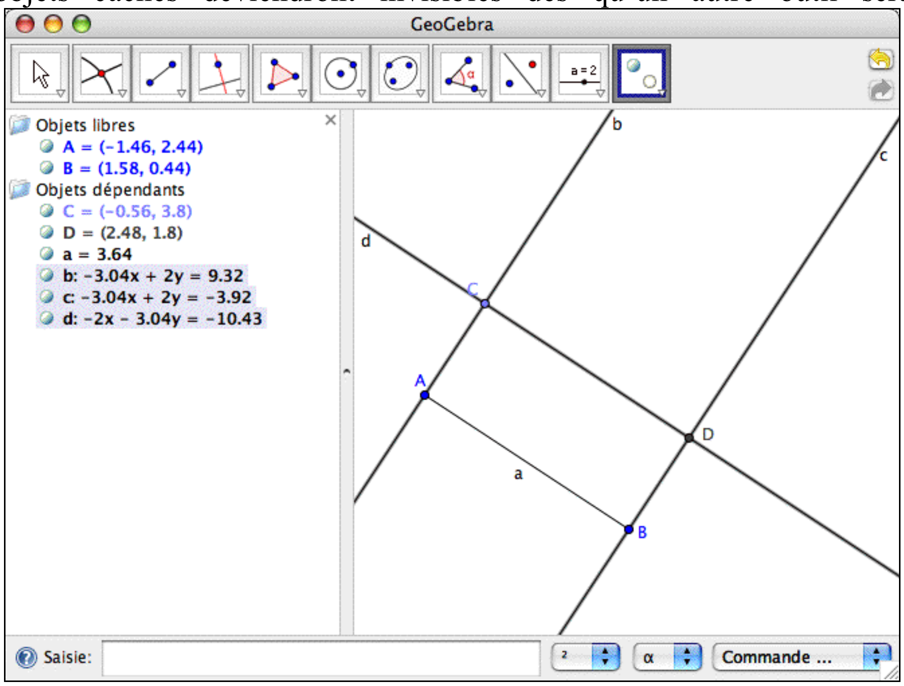
Pour corriger une erreur

On peut toujours annuler la dernière action en invoquant le menu « Éditer » ➤ « Annuler ». Si désiré, on peut utiliser plusieurs fois cette commande, et même se raviser (« Éditer » ➤ « Refaire »).

On peut aussi détruire un ou plusieurs objets en les englobant dans un rectangle défini avec l'outil « Déplacer » pour ensuite invoquer le menu « Editer » ➤ « Effacer ». Notons que, dans ce dernier cas, tous les objets *dépendant* de l'objet détruit seront aussi détruits. (On dit qu'un objet *dépend* d'un autre si sa construction fait appel, directement ou indirectement, à cet autre objet. Ainsi une perpendiculaire à un segment par un point *P* donné *dépend* non seulement du point *P* et du segment, mais aussi des extrémités de ce segment. De plus, si *P* a été obtenu comme intersection de deux droites, la perpendiculaire dépendra aussi de ces deux droites.)

- Nous sélectionnons ensuite l'outil « Droite perpendiculaire ». Nous désignons d'abord par un clic l'extrémité *A* (notez que, lorsque nous sommes près du point *A*, le curseur *GeoGebra* change d'aspect et indique « Point A », confirmant ainsi que *GeoGebra* l'a bien identifié). Nous désignons ensuite par un second clic le segment *AB* (notez encore que, lorsque nous sommes près du segment, le curseur *GeoGebra* change d'aspect et indique « Segment a : Segment[AB] », confirmant ainsi que *GeoGebra* l'a bien identifié). La droite perpendiculaire désirée apparaît alors.
- Après avoir sélectionné l'outil « Nouveau point », nous nous approchons de la droite perpendiculaire que nous venons de créer (le curseur *GeoGebra* nous confirme « Droite b : Perpendiculaire à a passant par a ») et nous désignons par un clic un point *C* sur cette droite.
- En procédant comme précédemment, nous traçons ensuite la perpendiculaire au segment *AB* passant par le point *B*, ainsi que la perpendiculaire au segment *AC* passant par le point *C*.
- Après avoir sélectionné l'outil « Intersection entre deux objets », nous désignons successivement les deux droites perpendiculaires (confirmation) que nous venons de créer : nous nommons *D* le point d'intersection ainsi construit (si GeoGebra lui a donné un autre nom).
- Nous voulons maintenant cacher (et non détruire) les trois perpendiculaires que nous avons construites : il s'agit d'objets intermédiaires qui ont servi à créer deux des sommets du rectangle, mais que nous ne voulons pas voir dans la figure finale. Pour ce faire, nous choisissons l'outil « Afficher/cacher l'objet », puis nous désignons successivement par un clic les trois droites en question. Tant que cet outil reste sélectionné, les objets cachés

apparaissent encore, ce qui permet de les montrer à nouveau (toujours au moyen d'un clic) si désiré. Les objets cachés deviendront invisibles dès qu'un autre outil sera choisi.



Si la « Fenêtre Algèbre » est visible, les objets cachés apparaissent en surligné quand l'outil « Afficher/cacher l'objet » est sélectionné.

- Soulignons en passant que, quel que soit l'outil sélectionné, on peut désigner les objets tant en cliquant sur leur représentation graphique qu'en cliquant sur leur description dans la « Fenêtre Algèbre ». De plus, on peut afficher ou cacher un objet en cliquant sur la petite boule apparaissant à gauche de sa description dans la « Fenêtre Algèbre ».
- Pour terminer notre rectangle, il nous suffit alors de créer les segments *AC*, *BD* et *CD*.

Une fois notre rectangle terminé, on peut le transformer de diverses manières avec l'outil « Déplacer ». On peut saisir les points *A* et *B* et les déplacer librement dans le plan. On peut aussi saisir le point *C*, mais on ne peut le déplacer que le long de la perpendiculaire à *AB* passant par *A* : c'est qu'il a été créé avec cette contrainte. Par contre, on ne peut pas déplacer le point *D*, qui est totalement déterminé en tant qu'intersection de deux droites. De même, on constate que *AB* est le seul segment qu'on peut déplacer (parallèlement à lui-même) : c'est que ses extrémités sont toutes deux des points *libres* (créés sans contraintes, avec l'outil « Nouveau point »).

Il est important de remarquer que toutes les manipulations permises par *GeoGebra* respectent la construction que nous avons utilisée pour définir notre figure : les perpendiculaires restent

Distinction entre *figure*, *construction* et *dessin*

Dans le contexte d'utilisation de *GeoGebra*, il importe de bien distinguer entre **figure**, **construction** et **dessin**. La **figure** est l'objet géométrique que nous voulons représenter, un *rectangle* dans l'exemple ci-dessus. La **construction** est l'ensemble des étapes nécessaires pour décrire notre **figure** dans *GeoGebra*. Et le **dessin** est la représentation graphique sur l'écran produite par *GeoGebra*. En se rappelant ce que nous avons vu à la section 1.3, on peut aussi dire que la **construction** est un ensemble d'instructions décrivant un type particulier de *graphique vectoriel*[1], tandis que le **dessin** en constitue une représentation *matricielle*.

perpendiculaires, les intersections restent des intersections, les extrémités restent des extrémités, etc. En conséquence, bien que nous puissions modifier notre dessin, il représente toujours un rectangle, puisque la construction utilisée visait cette contrainte.
Notons que *GeoGebra* nous permet aussi de tracer le **dessin** d'un rectangle, en utilisant 4 segments horizontaux et verticaux (obtenus en utilisant la *touche option/alt*) qui forment visuellement un rectangle, mais qui ne sont pas perpendiculaires par construction. D'ailleurs, on peut facilement déformer ce « rectangle » en un quadrilatère quelconque (voir figure 6.1.3).

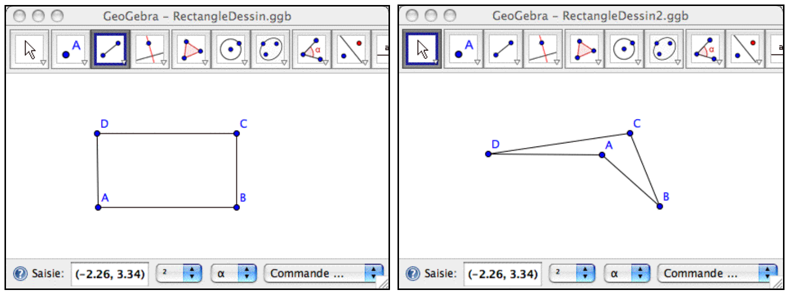


Figure 6.1.3 Le « dessin d'un rectangle » dans *GeoGebra* peut être aisément déformé.

Avant de passer à la section suivante, nous vous suggérons de vous familiariser un peu avec *GeoGebra* en travaillant certains exercices.

6.2 Mathématiques expérimentales avec GeoGebra

Nous allons maintenant utiliser *GeoGebra* pour faire une étude expérimentale du phénomène suivant : on place un point *D* à l'intérieur d'un triangle équilatéral *ABC*, et on étudie la somme des distances de *D* aux trois côtés du triangle (voir figure 6.2.4). Ce faisant, nous pourrons constater que les outils de *GeoGebra* ne se limitent pas à la seule géométrie *synthétique* : on peut aussi mesurer des longueurs et des angles, puis utiliser le *champ de saisie* pour faire des calculs à partir des nombres ainsi obtenus.

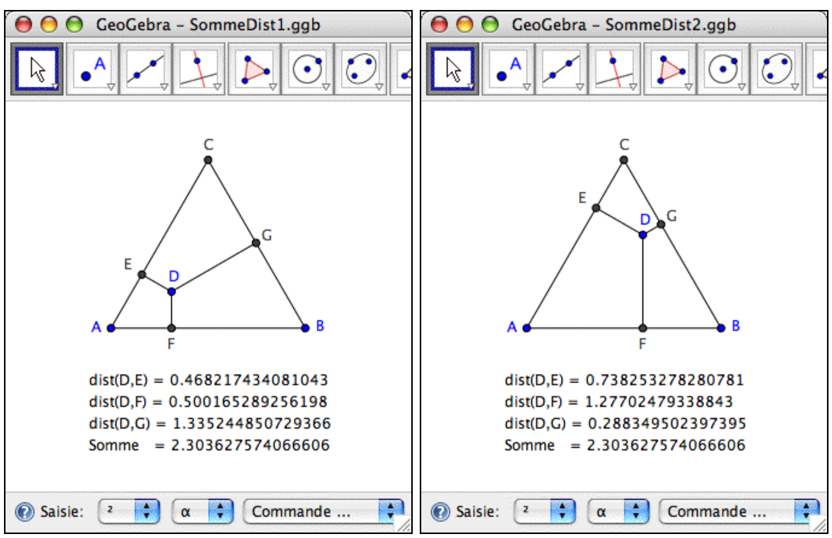


Figure 6.2.4 Deux positions du point *D* à l'intérieur du triangle équilatéral *ABC* :
la somme des distances de *D* aux trois côtés du triangle semble constante.

Commençons par décrire une construction pour réaliser dans *GeoGebra* la figure que nous venons de décrire.

- Nous construisons en premier le segment *AB*.
- Le point *C* est obtenu comme une intersection de deux cercles (outil « Cercle (centre-point) ») : celui de centre *A* et passant par *B*, et celui de centre *B* et passant par *A*. Par la suite, on cachera ces deux cercles et on tracera les segments joignant *C* à *A* et à *B*.
- On placera ensuite un point *D* à l'intérieur du triangle. Soulignons ici qu'il ne s'agit pas d'une contrainte pour le point *D* : au départ, on positionne le point *D* à l'intérieur du triangle, mais on pourra le déplacer par la suite à l'extérieur de celui-ci.
- Pour chacun des côtés du triangle *ABC*, on mène la perpendiculaire à ce côté et passant par le point *D*. Par la suite, on trouve le *pied de la perpendiculaire* (c'est-à-dire l'intersection de la perpendiculaire et du côté). On peut alors cacher la perpendiculaire, puis tracer le segment reliant *D* au pied de la perpendiculaire. On obtient ainsi les segments *DE*, *DF* et *DG*.
- On pourrait mesurer la longueur des segments *DE*, *DF* et *DG* l'aide de l'outil « Distance ou Longueur », mais on va plutôt procéder en utilisant le *champ de saisie*. Si on tape « longDE=Distance[D,E] » dans le champ de saisie, on définit un nombre qui apparaîtra dans la « Fenêtre Algèbre » mais pas dans la zone graphique. Pour l'afficher dans celle-ci, on utilisera l'outil « Insérer un texte » : on cliquera à l'endroit où l'on veut afficher notre texte, puis on tapera dans la fenêtre de dialogue le texte désiré, soit « "dist(D,E) = "+longDE ». Ce texte comporte deux parties jointes par un « + » : une première (entre guillemets) qui sera affichée telle quelle, et une seconde qui réfère à une expression connue de *GeoGebra* et dont *la valeur* sera affichée. On procédera de la même façon pour les deux autres longueurs.
  Important: les commandes du champ de saisie distinguent entre majuscules et minuscules.

> **Modification des préférences de *GeoGebra***
>
> Vous avez peut-être remarqué que tous les nombres affichés dans notre exemple comportent 15 décimales. En fait, nous pouvons dicter nos préférences en la matière à *GeoGebra*: il suffit de choisir le menu « Options » ➤ « Arrondi » puis de choisir l'affichage voulu. Nous reviendrons au besoin sur les autres paramètres que nous pouvons modifier, mais rien ne vous empêche d'y jeter dès à présent un petit coup d'oeil.

- Il ne nous reste plus qu'à faire la somme de ces trois longueurs : on utilisera l'outil « Insérer un texte » avec le texte « "Somme = "+(longDE+ longDF+ longDG) ».

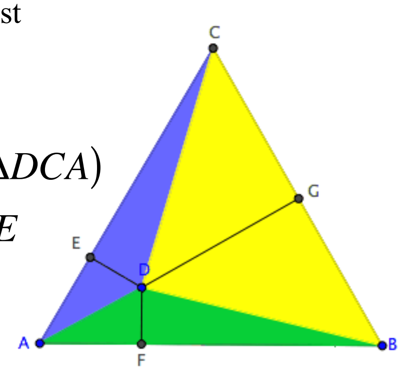
Précisons ici que nous avons choisi une construction parmi plusieurs autres : nous aurions aussi pu, par exemple, réaliser le triangle équilatéral *ABC* avec l'outil « Polygone régulier ».
Maintenant que la construction est terminée, nous pouvons procéder à des expérimentations. En faisant varier la position du point *D*, nous pouvons faire les observations suivantes :

- En déplaçant le point *D* à l'intérieur du triangle *ABC*, on constate que les 3 distances affichées sont automatiquement mises à jour. Elles disparaissent quand elles ne sont plus définies.
- Tant que le point *D* reste à l'intérieur du triangle *ABC*, la somme des distances aux côtés semble constante. On constate bien parfois une légère variation de la dernière décimale (la moins significative), mais on peut raisonnablement l'attribuer à la précision limitée des calculs sur des nombres en virgule flottante.
- Quand le point *D* sort du triangle, deux cas peuvent se présenter : quand les perpendiculaires issues de *D* tombent bien sur les trois côtés, la somme est définie et elle varie clairement;

quand les perpendiculaires issues de *D* ne tombent pas toutes sur les côtés, la somme n'est pas définie.

Concentrons nous sur la situation où *D* reste à l'intérieur du triangle. Le fait de pouvoir promener le point *D* à notre guise tout en constatant que la somme des distances demeure (à toutes fins pratiques) inchangée nous procure une **forte conviction** que le résultat en question est **toujours vérifié**. Cependant, comme dans à peu près toutes les **vérifications expérimentales**, nous rencontrons ici des problèmes de

- *Généralité* : une vérification exhaustive devrait porter sur une infinité de points, alors que seules les positions correspondant à un pixel de l'écran pourront être considérées par *GeoGebra*. Mais ce nombre de vérifications, tout fini qu'il soit, reste quand même très élevé quand on le compare au nombre de vérifications manuelles que nous pourrions faire (en un temps raisonnable) en l'absence de la technologie.
- *Certitude* : tous les calculs (coordonnées des points, calculs de distances, etc.) se font avec une précision limitée et les résultats affichés ne sont qu'approximatifs. Mais, dans ce cas précis, nous ne prévoyons aucune instabilité numérique pouvant nous induire en erreur.
- *Compréhension* : même si nos expérimentations nous procurent une quasi-certitude (une *certitude pratique*) de la véracité du résultat, elles ne nous font pas comprendre pourquoi ledit résultat est vrai. Pour atteindre une compréhension véritable (ainsi qu'une *certitude théorique*), il nous faudra faire appel à une argumentation/démonstration/preuve.
  Voici ce à quoi ça pourrait ressembler : comme le triangle *ABC* est équilatéral, les trois côtés *AB*, *BC* et *CA* ont une même longueur (que nous désignerons par *c*) et, quand le point *D* est à l'intérieur de ce triangle, on a



$$\begin{aligned} aire(\Delta ABC) &= aire(\Delta DAB) + aire(\Delta DBC) + aire(\Delta DCA) \\ &= \tfrac{1}{2} AB \times DF + \tfrac{1}{2} BC \times DG + \tfrac{1}{2} CA \times DE \\ &= \tfrac{1}{2} c \times DF + \tfrac{1}{2} c \times DG + \tfrac{1}{2} c \times DE \\ &= \tfrac{1}{2} c(DF + DG + DE) \end{aligned}$$

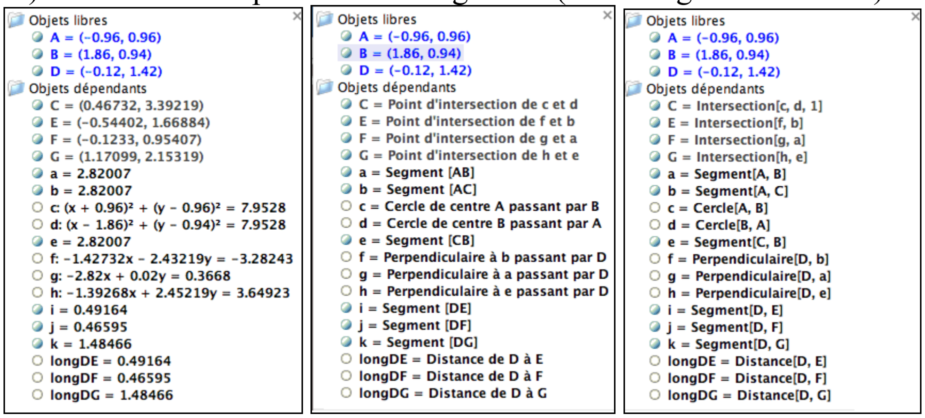
  ce qui implique que $DE + DF + DG = \dfrac{2aire(\Delta ABC)}{c}$, qui est une valeur ne dépendant pas de la position du point *D*. En fait, on démontre que *DE* + *DF* + *DG* vaut la hauteur du triangle. (Comment ?)

La double nature de *GeoGebra*

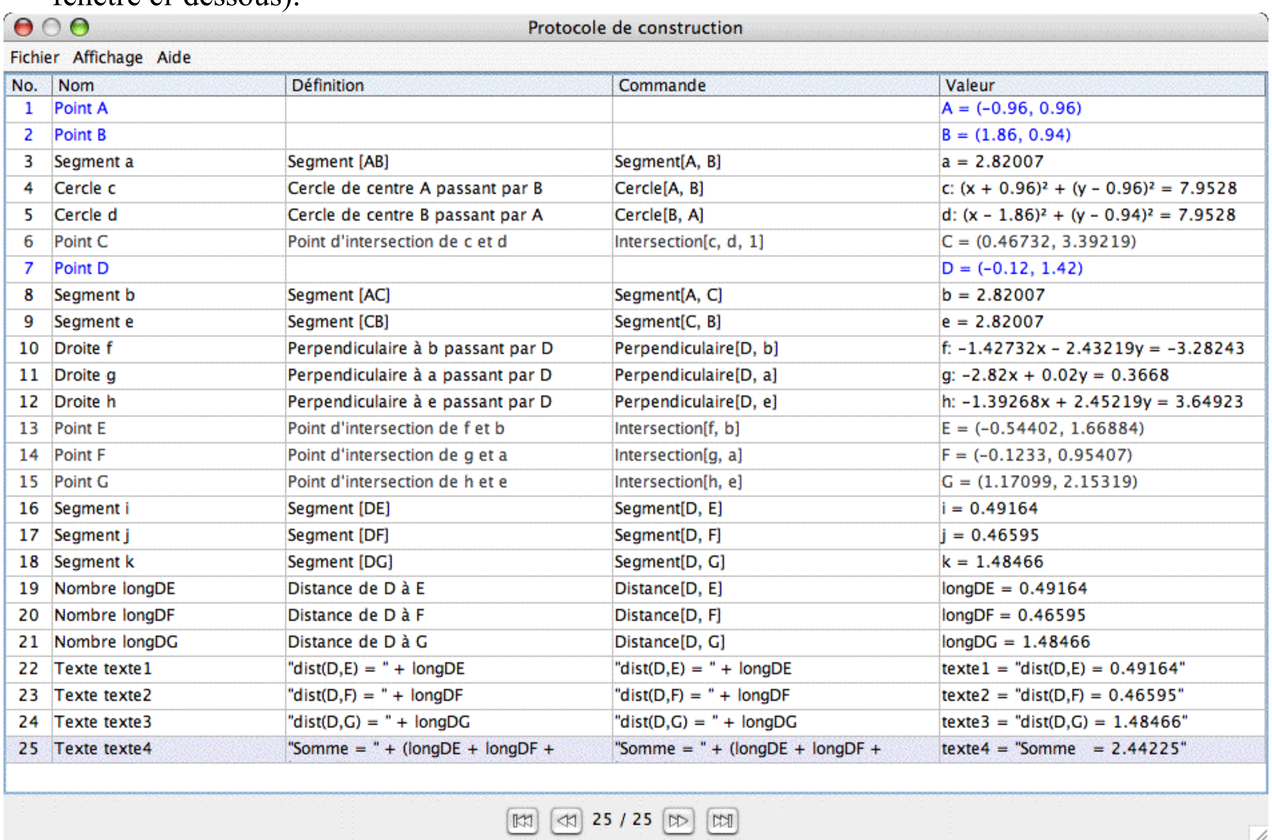
La figure que nous venons de construire est issue d'une communication *gestuelle* avec *GeoGebra* : elle a été produite par une suite de choix, effectués principalement avec la souris, dans des menus et des fenêtres de dialogue. *GeoGebra* nous permet aussi une communication *textuelle* avec lui, au moyen d'une suite de commandes que nous pouvons taper dans le *champ de saisie*. À priori, cela peut sembler une façon plus compliquée de communiquer, qui nécessite la connaissance des codes pour les diverses commandes, mais nous verrons plus tard qu'il est parfois avantageux de procéder de la sorte, notamment quand nous utiliserons des expressions mathématiques, des conditions ou des commandes itératives pour décrire nos figures.

On peut voir des indices de la possibilité d'une communication *textuelle* avec *GeoGebra* en remarquant

- la présence du *champ de saisie*,
- la possibilité d'afficher la « Fenêtre Algèbre » sous trois formes (*Valeur*, *Définition* et *Commande*) via le menu « Options » ➤ « Algèbre » (voir la figure ci-dessous)



- et le menu « Affichage » ➤ « Protocole de construction ... » qui nous donne, dans l'ordre, la liste des commandes qu'on aurait pu utiliser pour obtenir *textuellement* notre figure (voir la fenêtre ci-dessous).



Protocole de construction

Fichier Affichage Aide

No.	Nom	Définition	Commande	Valeur
1	Point A			A = (-0.96, 0.96)
2	Point B			B = (1.86, 0.94)
3	Segment a	Segment [AB]	Segment[A, B]	a = 2.82007
4	Cercle c	Cercle de centre A passant par B	Cercle[A, B]	c: (x + 0.96)² + (y - 0.96)² = 7.9528
5	Cercle d	Cercle de centre B passant par A	Cercle[B, A]	d: (x - 1.86)² + (y - 0.94)² = 7.9528
6	Point C	Point d'intersection de c et d	Intersection[c, d, 1]	C = (0.46732, 3.39219)
7	Point D			D = (-0.12, 1.42)
8	Segment b	Segment [AC]	Segment[A, C]	b = 2.82007
9	Segment e	Segment [CB]	Segment[C, B]	e = 2.82007
10	Droite f	Perpendiculaire à b passant par D	Perpendiculaire[D, b]	f: -1.42732x - 2.43219y = -3.28243
11	Droite g	Perpendiculaire à a passant par D	Perpendiculaire[D, a]	g: -2.82x + 0.02y = 0.3668
12	Droite h	Perpendiculaire à e passant par D	Perpendiculaire[D, e]	h: -1.39268x + 2.45219y = 3.64923
13	Point E	Point d'intersection de f et b	Intersection[f, b]	E = (-0.54402, 1.66884)
14	Point F	Point d'intersection de g et a	Intersection[g, a]	F = (-0.1233, 0.95407)
15	Point G	Point d'intersection de h et e	Intersection[h, e]	G = (1.17099, 2.15319)
16	Segment i	Segment [DE]	Segment[D, E]	i = 0.49164
17	Segment j	Segment [DF]	Segment[D, F]	j = 0.46595
18	Segment k	Segment [DG]	Segment[D, G]	k = 1.48466
19	Nombre longDE	Distance de D à E	Distance[D, E]	longDE = 0.49164
20	Nombre longDF	Distance de D à F	Distance[D, F]	longDF = 0.46595
21	Nombre longDG	Distance de D à G	Distance[D, G]	longDG = 1.48466
22	Texte texte1	"dist(D,E) = " + longDE	"dist(D,E) = " + longDE	texte1 = "dist(D,E) = 0.49164"
23	Texte texte2	"dist(D,F) = " + longDF	"dist(D,F) = " + longDF	texte2 = "dist(D,F) = 0.46595"
24	Texte texte3	"dist(D,G) = " + longDG	"dist(D,G) = " + longDG	texte3 = "dist(D,G) = 1.48466"
25	Texte texte4	"Somme = " + (longDE + longDF +	"Somme = " + (longDE + longDF +	texte4 = "Somme = 2.44225"

25 / 25

Dans ce cas de la figure ci-dessus, il suffirait donc de taper successivement, dans le *champ de saisie* d'une nouvelle fenêtre, les commandes suivantes pour obtenir une figure identique (modulo la visibilité des éléments) à celle que nous avons obtenue *gestuellement* :

(0.96,0.96)
(1.86,0.94)
Segment[A,B]
Cercle[A,B]
…

Notons au passage quelques caractéristiques du *champ de saisie*[3] pouvant nous faciliter la tâche. Quand on commence à taper une commande, *GeoGebra* nous suggère parfois une complétion possible, que nous pouvons accepter (via la touche return) ou non (en continuant à taper). Ainsi, après avoir tapé « Cer », *GeoGebra* a affiché « Cercle[] ») : après avoir accepté (via la touche *return*) la suggestion, *GeoGebra* a placé le curseur entre les crochets pour nous faciliter la poursuite de l'entrée de la commande. Via le menu « Affichage » ➤ « Liste des commandes », on peut aussi afficher un menu local nous permettant de recopier dans le *champ de saisie* toute commande sélectionnée, nous évitant de la taper et (dans une moindre mesure) d'avoir à la retenir par cœur. Une dernière remarque sur le sujet : on aurait pu tout aussi bien taper « A=(0.96,0.96) », « b= Segment[A,B] » et « c= Cercle[A,B] » si on avait voulu choisir nous-même les noms des objets au lieu de laisser *GeoGebra* les nommer automatiquement pour nous. C'est ainsi, par exemple, que nous devrons utiliser la commande « longDE=Distance[D,E] » pour que cette distance soit nommée comme nous l'avons voulu dans notre construction initiale.

Avant de passer à la section suivante, nous vous suggérons d'acquérir plus d'expérience avec *GeoGebra* en travaillant certains exercices.

« GeoGebra nous dispense de faire des preuves »

C'est du moins une affirmation que nous entendons parfois lors de discussions avec des futurs maîtres en mathématiques. Bien sûr, on doit reconnaître que *GeoGebra* permet à des individus encore fort peu habiles à produire des démonstrations de se prononcer avec conviction et de façon généralement exacte sur la véracité d'énoncés en géométrie élémentaire. Mais il ne faut pas oublier que l'enseignement des mathématiques doit viser non seulement des connaissances, mais peut-être plus encore une compréhension (« pourquoi cet énoncé est-il toujours vrai ») que ne peuvent apporter même les vérifications expérimentales les plus exhaustives.

6.3 La création de macro-constructions

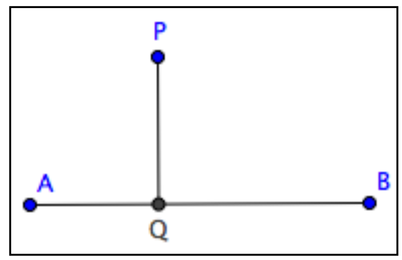
Les menus et la barre d'outils de *GeoGebra* mettent à notre disposition un grand nombre de possibilités. À l'aide du menu « Outils » ➤ « Barre d'outils personnalisée ... », on peut aussi disposer autrement ou même faire disparaître certains des outils disponibles. (Pour plus de détails sur la façon de procéder, nous vous référons au manuel de référence de *GeoGebra*.) Mais une des possibilités les plus intéressantes est sans contredit l'ajout de nouveaux outils, en créant des **macro-constructions**, souvent aussi désignées plus simplement par le terme **macros**.

[3] Pour ceux qui connaissent *Cabri*, un autre logiciel de géométrie dynamique, mentionnons que le champ de saisie de *GeoGebra* est beaucoup plus versatile que la calculatrice de *Cabri*. En fait, la communication avec *Cabri* se fait de façon essentiellement *gestuelle*.

La macro « Segment perpendiculaire

À la section précédente, nous avons dû construire trois fois un segment perpendiculaire à un autre segment, issu d'un point hors dudit segment. La tâche aurait été plus simple si *GeoGebra* avait comporté un outil réalisant précisément cette construction, mais ce n'est pas le cas. Nous allons maintenant voir comment nous pouvons ajouter un tel outil à *GeoGebra* : on pourra par la suite utiliser celui-ci tant pour refaire plus simplement la construction de la section précédente que pour réaliser de nouvelles figures.

Avant de créer notre macro, nous devons réaliser dans *GeoGebra* la construction qui servira à la définir. Dans notre cas, nous pouvons procéder de la façon suivante :



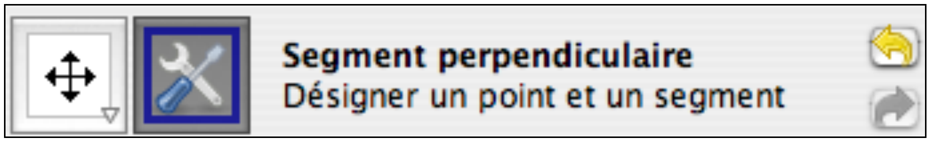
- Nous créons tout d'abord un point P et un segment AB. **Note** : pour renommer un objet ou cacher son étiquette, il suffit de faire un clic droit sur l'objet (ou encore sur sa description dans la « Fenêtre Algèbre »), et de faire le choix approprié dans le menu local qui apparaît alors.
- En utilisant l'outil « Droite perpendiculaire », nous traçons la perpendiculaire au segment AB passant par le point P : en déplaçant, si nécessaire, le point P, nous nous assurons que cette perpendiculaire coupe le segment AB en un point Q.
- Nous cachons ensuite la droite perpendiculaire et nous traçons le segment PQ.

Nous sommes maintenant prêts à définir notre nouvel outil, que nous appellerons « Segment perpendiculaire ». Cette définition se déroulera en trois étapes :

(1) On choisit le menu « Outils » ➤ « Créer un nouvel outil », ce qui crée une fenêtre de dialogue avec l'onglet « Objets finaux » sélectionné (voir figure 6.3.5 à gauche). On désigne tous les objets que l'on désire voir construits par la macro-construction : dans notre cas, il s'agit du segment PQ. Notons en passant que *GeoGebra* devra construire la perpendiculaire pour obtenir le nouveau triangle, mais celle-ci ne doit pas être désignée comme objet final si nous voulons qu'elle reste invisible. Soulignons ici que la désignation des objets peut se faire de trois façons : un clic sur l'objet, un clic sur la désignation de l'objet dans la « Fenêtre Algèbre », ou un choix dans le menu local. On peut changer l'ordre des objets ou enlever certains des objets via les boutons de droite. Ceci est valable tant pour les objets initiaux de l'étape suivante que pour les objets finaux de la présente étape.

(2) On clique sur l'onglet « Objets initiaux », puis on désigne tous les objets nécessaires pour produire notre construction : dans notre cas, il s'agit du point P et du segment AB, que nous désignons dans cet ordre (voir figure 6.3.5 au centre). Ces objets initiaux devront être suffisants pour permettre la construction des objets finaux de l'étape précédente. (Par exemple, nous aurions pu utiliser les points P, A et B comme objets initiaux, mais nous n'aurions pu nous contenter de désigner seulement deux de ces points).

(3) On clique sur l'onglet « Nom et icône ». On remplit d'abord les trois champs de texte (voir figure 6.3.5 à droite) : le nom (« Segment perpendiculaire »), le nom de la commande textuelle correspondante (« SegmentPerpendiculaire ») et le texte de l'aide qui apparaîtra quand l'outil sera sélectionné.



Nous terminons par un clic sur le bouton « Fin ». *GeoGebra* vérifie alors si les objets initiaux sont suffisants pour construire les objets finaux, et nous avertit en cas de problème. Par contre, si tout se passe bien, la macro est ajoutée à notre barre d'outils (du moins si nous n'avons pas décoché la case correspondante).

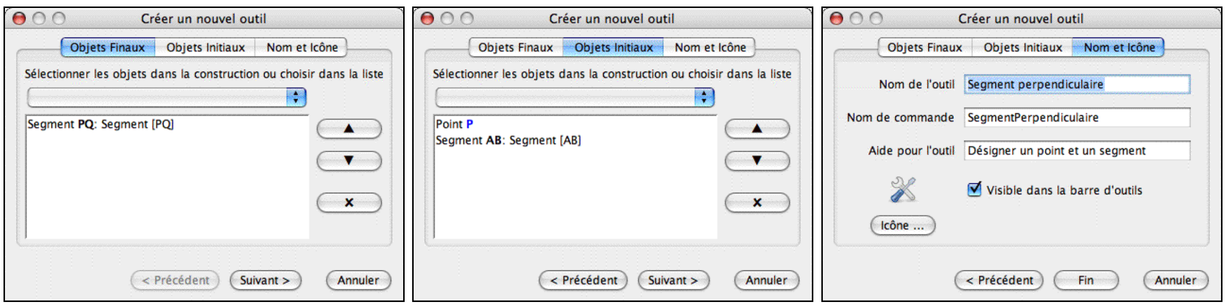


Figure 6.3.5 Fenêtre de dialogue pour définir notre macro.

Nos macros seront enregistrées automatiquement dans nos figures. Nous pouvons aussi les enregistrer indépendamment (via le menu « Outils » ➤ « Gérer les outils … »), tel qu'illustré à la figure 6.3.6, de façon à pouvoir les ramener au besoin. Elles sont aussi sauvegardées automatiquement quand utilise le menu « Options » ➤ « Sauvegarder la configuration »[4].

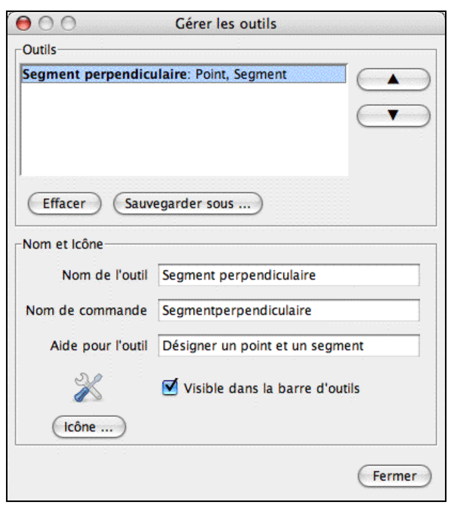


Figure 6.3.6 Gestion des macros.

[4] Pour ceux qui connaissent le logiciel *Cabri*, mentionnons que les macros de *GeoGebra* sont moins versatiles : on ne peut les surcharger (voir figure 6.5 de la page 207 du manuel) et on ne peut désigner les objets initiaux dans un ordre différent (quand il n'y a aucune ambiguïté) lorsqu'on utilise nos macros.

La macro « Étape de la spirale »

Imaginons maintenant, que l'on veuille créer la figure 6.3.7. Il s'agit d'une portion de la spirale de Pythagore, nommée ainsi parce des applications successives du théorème de Pythagore permettent de calculer les longueurs des hypoténuses des triangles rectangles qui la constituent : si les deux petits côtés d'un triangle rectangle (les *cathètes*) mesurent 1 et $\sqrt{n}$, alors l'hypoténuse mesurera $\sqrt{n+1}$.

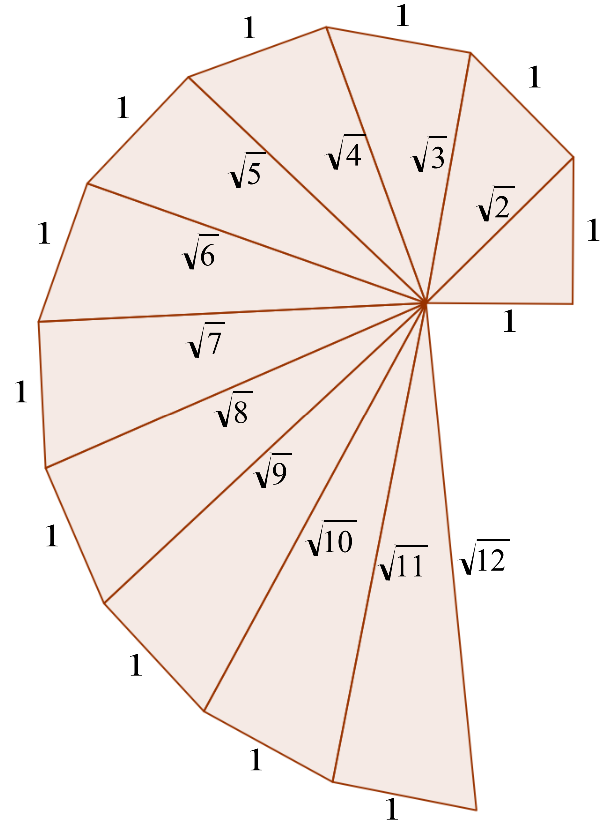


Figure 6.3.7 Portion de la spirale de Pythagore.
La construction a été faite dans *GeoGebra*, et les annotations ont été ajoutées dans *Word* en utilisant l'outil « Ajouter une étiquette » (de *LangageGraphique*) et l'éditeur d'équations.

La construction d'une telle figure ne pose pas de difficultés particulières sinon qu'elle promet d'être longue et répétitive. On souhaiterait disposer d'un outil (temporaire parce que spécifique à la présente construction) nous permettant de construire un nouveau triangle à partir d'un triangle donné, tel qu'illustré à la figure 6.3.8. (Note : pour obtenir un nom avec indice dans *GeoGebra*, il suffit de faire précéder l'indice du caractère « souligné ». Ainsi « B_2 » a été obtenu via « B_2 ».)

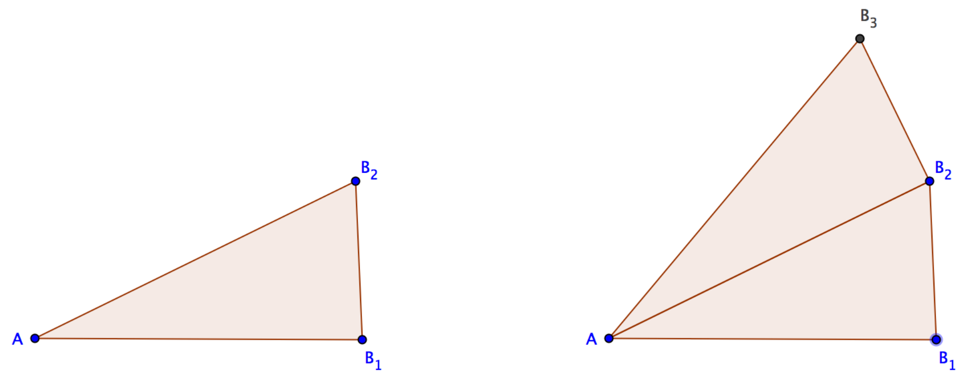


Figure 6.3.8 À gauche, le triangle initial. À droite, le nouveau triangle a été ajouté.

La figure 6.3.9 illustre les détails de la construction, qui ont été cachés par la suite : le point B_3 est obtenu comme intersection de la perpendiculaire par B_2 au côté AB_2 et du cercle centré en B_2 qui passe par B_1. Notons en passant que cette droite et ce cercle ont deux points d'intersection, et que nous devons choisir celui qui est approprié dans les circonstances. Il ne nous resterait plus qu'à relier les points A, B_2 et B_3 en un triangle (via l'outil « Polygone »).

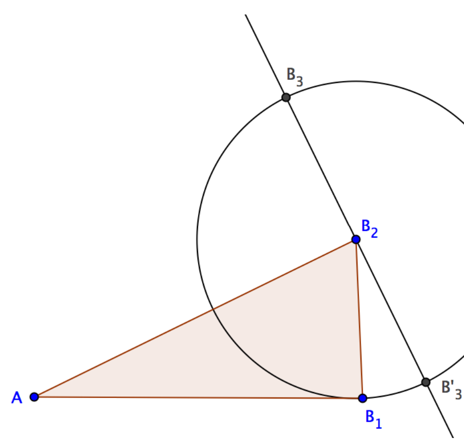


Figure 6.3.9 Construction du point B_3 à partir du triangle AB_1B_2.

Soulignons aussi que notre construction du triangle AB_2B_3 dépend non seulement du triangle de départ (AB_1B_2) mais aussi des sommets choisis pour élever la perpendiculaire (B_2), désigner le côté par rapport auquel la perpendiculaire sera élevée (A et B_2), servir de centre (encore B_2), et déterminer un point sur le cercle (B_1). Convenons de définir tous les triangles de notre construction en désignant ses sommets dans l'ordre suivant : d'abord le centre de la spirale (A), puis le sommet de l'angle droit (B_1 pour le premier triangle, B_2 pour le second), et enfin le dernier sommet (B_2 pour le premier triangle, B_3 pour le second).

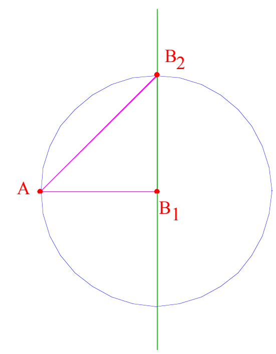
Supposons donc que nous ayons réalisé dans *GeoGebra* la construction illustrée dans la portion de droite de la figure 6.3.8 et que les deux triangles en présence aient été définis comme décrit au paragraphe précédent. Nous sommes maintenant prêts à définir une *macro* qui simplifiera la construction (d'une portion) de la spirale de Pythagore. Comme précédemment, cette définition se déroulera en trois étapes :

(1) On choisit le menu « Outils » ➤ « Créer un nouvel outil … », puis on désigne le point B_3 et le triangle AB_2B_3 comme seuls objets finaux. Notons en passant que *GeoGebra* devra construire la perpendiculaire et le cercle de la figure 6.3.9 pour obtenir le nouveau triangle, mais ceux-ci ne doivent pas être désignés comme objets finaux si nous voulons qu'ils restent invisibles.

(2) On clique sur l'onglet « Objets initiaux », puis on constate que *GeoGebra* nous suggère les sommets A, B_1 et B_2 (dans cet ordre) comme objets initiaux : nous acceptons.

(3) On clique sur l'onglet « Nom et Icône », et on donne alors le nom « Étape spirale » à notre macro.

Au terme des trois étapes précédentes, nous pouvons vérifier qu'un nouvel outil « Étape de la spirale » a bien été ajouté au menu « Macros ». Nous pouvons maintenant construire facilement notre spirale de Pythagore comme suit :



- On construit d'abord un triangle rectangle isocèle comme ci-contre, puis on cache les objets intermédiaires (cercle et perpendiculaire). Rappelons que ce triangle initial doit être défini en désignant ses trois sommets dans un ordre précis : centre de la spirale (A), sommet de l'angle droit (B_1) et troisième sommet (B_2).
- On choisit enfin l'outil « Étape de la spirale » et l'on désigne les trois sommets du triangle dans un ordre précis : centre de la spirale (A), sommet de l'angle droit (B_1) et troisième sommet (B_2) : un nouveau triangle est ajouté.
- On continue de la sorte autant de fois que désiré : un clic sur chacun des sommets du dernier triangle construit (dans le bon ordre) produira un nouveau triangle de la spirale.
- On peut alors choisir l'outil « Déplacer » et modifier notre triangle de départ, pour constater comment notre spirale s'adaptera dynamiquement à ces changements.

Tout semble donc aller pour le mieux. Mais une surprise nous attend quand nous tentons d'appliquer la macro au triangle de gauche de la figure 6.3.10. Que se passe-t-il ? D'autres essais nous portent à émettre l'hypothèse suivante : le problème survient quand le triangle AB_1B_2 est parcouru dans le sens des aiguilles d'une montre, à l'inverse du cas de figure où la macro a été définie. Mais pourquoi en est-il ainsi ?

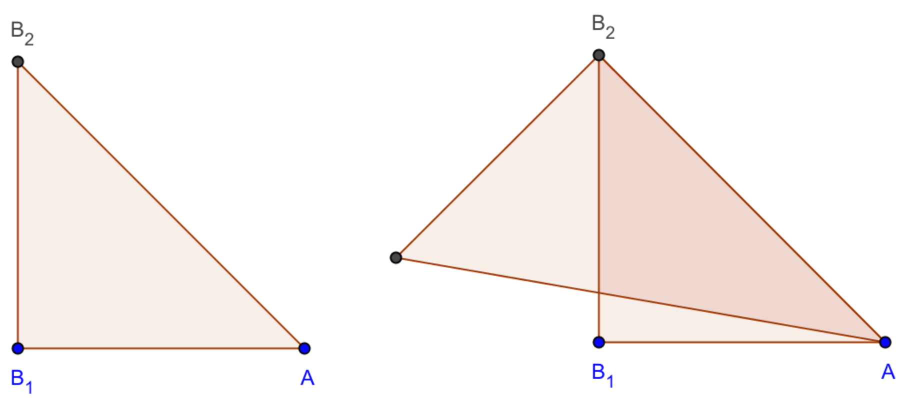


Figure 6.3.10 Quand on tente d'appliquer la macro « Étape de la spirale » à la figure de gauche, on obtient le résultat à droite.

Pour comprendre ce qui se passe, il nous faut revenir à la figure 6.3.9. Rappelons-nous que le point B_3 a été choisi **visuellement** parmi les deux points d'intersection comme celui qui était le plus approprié dans les circonstances. Or, dans un logiciel permettant de modifier dynamiquement les figures, ces circonstances varient et nous constatons que nos choix aussi devraient s'adapter : **le point B_3 devrait non pas être choisi mais bien construit**. Mais comment ? Un moment de réflexion nous amère à énoncer le critère suivant : les points B_1 et B_3 doivent se retrouver dans des demi-plans distincts, tels que déterminés par la droite passant par A et B_2 ou, de manière équivalente, les segments AB_2 et B_1B_3 doivent se couper en un point F.

Il nous suffit maintenant d'imaginer une construction qui nous assurera de trouver le point d'intersection se situant dans le bon demi-plan. Une solution possible est illustrée dans la figure 6.3.11. On peut la décrire comme suit :

- Soient B'_3 et B''_3 les points d'intersection de la perpendiculaire et du cercle. Nous construisons le segment $B_1B''_3$ et trouvons son point d'intersection F avec le segment AB_2.
- Si F existe, alors nous définissons B_3 comme étant B''_3, sinon nous le définissons comme étant B'_3 : on voit que B_3 sera toujours le point voulu.
- Mais comment définir B_3 dans GeoGebra? Il faudra procéder *textuellement*[5]!
  - On peut faire la construction de F ci dessus et taper dans le *champ de saisie* :

    B_3=Si[EstDéfini[F],B"_3,B'_3]

    Notons que ceci suppose de choisir B''_3 de telle sorte que le point F soit **défini dans notre cas de figure**.
  - On peut aussi éviter de faire la construction, et taper dans le *champ de saisie* : B_3=Si[EstDéfini[Intersection[Segment[A,B_2],Segment[B_1,B"_3]]],B"_3,B'_3]
- On utilisera alors le point B_3 et le triangle AB_2B_3 comme objets finaux de notre macro.

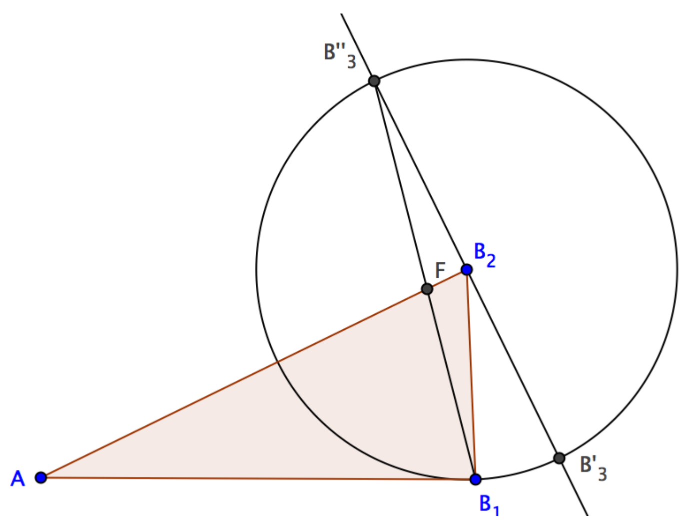


Figure 6.3.11 Construction du point d'intersection approprié.

[5] Soulignons ici la présence du **Si** dans *GeoGebra*, instruction qui facilite grandement la création de constructions conditionnelles. Pour ceux qui connaissent le logiciel *Cabri*, mentionnons que les constructions conditionnelles sont beaucoup moins naturelles dans *Cabri* et ont nécessité le développement de ce qu'on a appelé des géométries *logique* et *booléenne*.

Après avoir effacé notre macro précédente (via le menu « Outils » ➤ « Gérer les outils … »), qui était incorrecte, nous utilisons la construction ci-dessus pour définir une nouvelle macro (que nous nommerons aussi « Étape spirale ») Nous pourrons l'utiliser pour tracer une spirale de Pythagore à partir de n'importe quel triangle rectangle isocèle, quelle que soit l'orientation reçue lors de sa définition[6] : la figure 6.3.12 en donne un exemple.

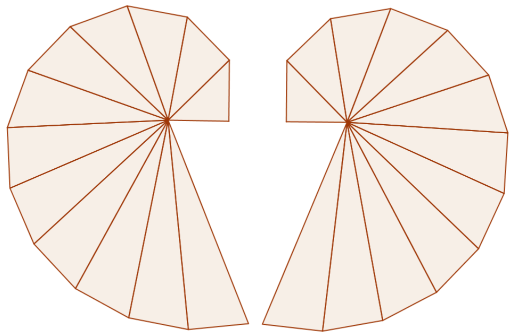
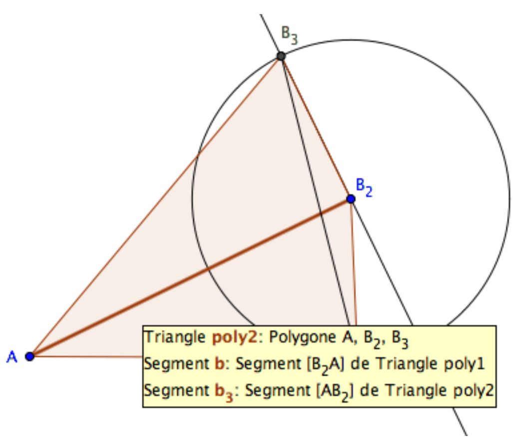
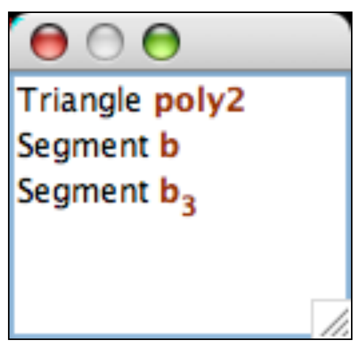


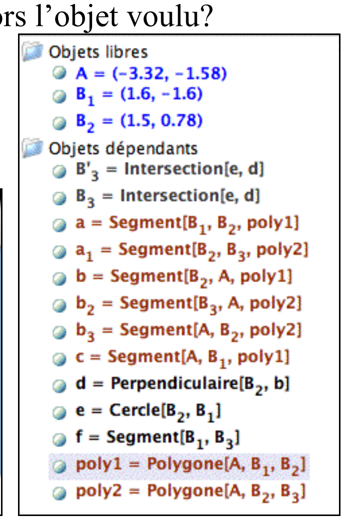
Figure 6.3.12 Deux portions symétriques de spirales de Pythagore.

La levée des ambiguïtés dans *GeoGebra*

Normalement, dans *GeoGebra*, quand le pointeur de la souris survole un objet, il identifie celui-ci en affichant son nom dans un rectangle. Mais il peut arriver qu'il y ait ambiguïté, lorsque plusieurs objets sont superposés à la position du pointeur : ainsi, dans la figure 6.3.11, quand nous avons voulu désigner le segment AB_2, *GeoGebra* nous a indiqué qu'il y avait plusieurs objets sous notre pointeur (voir à gauche ci-dessous). Comment désigner alors l'objet voulu?







Il y a au moins deux façons de procéder.

- La première façon consiste à cliquer sans changer la position du pointeur : GeoGebra affiche alors une fenêtre donnant la liste des noms des objets en question (au milieu ci-dessus) et il nous suffit de cliquer sur le nom l'objet que nous voulons désigner.
- La second façon consiste à cliquer sur son nom dans la « Fenêtre Algèbre » (à droite ci-dessus), nom qui est significatif si l'option « Définition » ou « Commande » a été choisie via le menu « Options » ➤ « Algèbre ».

[6] Soulignons seulement qu'il faudra désigner le sommet de l'angle droit en second dans la liste des sommets, lors de la définition du triangle de base.

6.4 Initiation aux lieux géométriques

GeoGebra nous permet aussi de créer des **lieux géométriques**. Illustrons ceci en considérant la **parabole** définie non pas comme une section de cône mais plutôt comme l'ensemble des points *P* du plan qui sont à égale distance d'un point *F* appelé **foyer** et d'une droite *d* appelée **directrice** (voir figure 6.4.13).

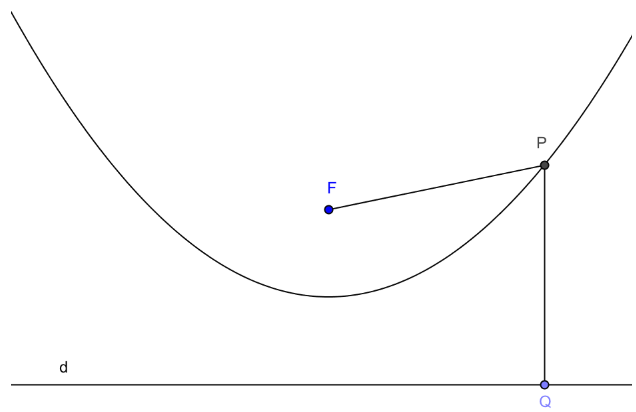


Figure 6.4.13 Définition d'une parabole en tant que lieu des points *P* à égale distance du foyer *F* et de la directrice *d*.

Bien que *GeoGebra* dispose déjà d'un outil pour tracer une parabole de foyer et de directrice donnés, il nous semble intéresant de voir comment nous aurions pu ajouter nous-même cet outil s'il n'avait pas été disponible.

Dans un premier temps, nous allons essayer de transformer notre *description* de la parabole en *construction*. Étant donné un point *Q* sur la directrice, comment construire le point *P* qui lui correspond (sur la perpendiculaire à la directrice élevée en *Q*) ? Comme *P* doit se trouver à égale distance de *F* et de *Q*, *P* sera aussi sur la médiatrice du segment *FQ*, et donc à l'intersection de la médiatrice et de la perpendiculaire : ceci suffit à construire *P*.

Pour tout point *Q* sur la directrice, nous savons donc comment construire le point *P* correspondant sur la parabole. On peut donc imaginer placer plusieurs points *Q* sur la directrice et leur appliquer à tous la même construction (peut-être en se servant d'une macro) pour obtenir les points *P* correspondants. Mais *GeoGebra* dispose de deux outils pour rendre tout ceci plus simple.

Le premier outil de simplification est la *trace*. Un clic droit sur un objet nous permet d'activer ou de désactiver l'item « Trace activée » de son menu local. Tous les objets dont la trace est activée laisseront une trace quand on effectuera des déplacements avec l'outil « Déplacer». Par exemple, si on demande de laisser une trace du point *P* et que, par la suite, on déplace le point *Q* sur la droite *d*, on obtiendra un dessin pouvant ressembler à l'un de ceux de la figure 6.4.14. Noter que l'on peut passer et repasser plusieurs fois pour laisser une trace comportant plus de points. Soulignons que tous les objets dont la trace est activée continueront de laisser une trace tant et aussi longtemps que leur trace n'aura pas été désactivée. Notons que les traces en question n'ont aucune permanence : elles disparaissent automatiquement lors de certains événements (par exemple lorsqu'on redimentionne la fenêtre). On peut aussi effacer toutes les traces obtenues en utilisant le menu « Affichage » ➤ « Rafraîchir l'affichage ».

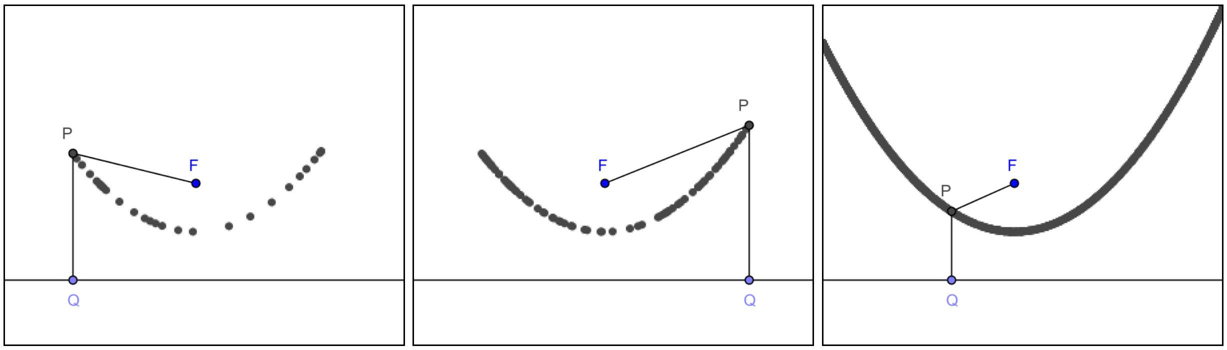


Figure 6.4.14 La trace du point *P* de la parabole quand on déplace le point *Q* sur la directrice. En repassant tout en ralentissant, on peut multiplier le nombre de points tracés.

Le second outil de simplification est le « Lieu ». On devra tout d'abord désigner un objet dont *GeoGebra* laissera une sorte de trace automatique (un *lieu*), puis un point sur un objet que *GeoGebra* déplacera (automatiquement, là aussi) pour obtenir le lieu en question. C'est de cette façon que nous avons obtenu la figure 6.4.13.

Le grand avantage d'un lieu sur une trace est que, contrairement aux traces, les lieux sont des objets géométriques à part entière : ils s'adaptent si on déplace des objets dont ils dépendent (la directrice, par exemple), et on peut construire des points sur des lieux. Dans la présente version de *GeoGebra*, on ne peut cependant pas obtenir les intersections de lieux avec d'autres objets.

Dans notre cas, par exemple, on peut maintenant définir une macro qui, étant donnés un foyer et une directrice, construira la parabole associée (en tant que lieu dans *GeoGebra*). Dans les exercices, nous verrons comment construire les autres coniques (ellipses et hyperboles) en tant que lieux géométriques.

6.5 Géométrie analytique avec GeoGebra

Jusqu'à présent, nous avons utilisé *GeoGebra* dans un contexte de **géométrie synthétique** : les seuls nombres qui interviennent apparaissent dans des contextes de mesure (de longueurs, d'angles, etc.). Mais *GeoGebra* nous permet aussi de travailler dans un contexte de **géométrie analytique**, utilisant coordonnées et équations. En fait, *GeoGebra* fait implicitement référence à un système d'axes permettant d'afficher les coordonnées et les équations des divers objets (dans la « Fenêtre Algèbre », lorsque la représentation « Options » ➤ « Algèbre » ➤ « Valeur » est choisie) : on peut d'ailleurs faire apparaître et disparaître les axes en cochant ou décochant l'item « Affichage » ➤ « Axes »[7].

[7] *GeoGebra* ne comporte qu'un seul système d'axes. Bien qu'il serait parfois utile d'en avoir plusieurs, cela évite les problèmes de consistance entre les coordonnées et les distances qu'on retrouvait dans *Cabri*.

On peut aussi changer notre système d'axes de plusieurs façons, notamment

- par un clic droit dans la zone graphique, suivi du choix de l'item « Graphique… » dans le menu local obtenu,
- par l'utilisation des outils « Déplacer Graphique », « Agrandissement » ou « Réduction »,
- par un glisser souris avec la touche majuscule enfoncée (translation),
- ou en utilisant la roulette présente sur plusieurs souris (zoom).

Voyons maintenant comment obtenir des graphes cartésiens de fonctions dans *GeoGebra*. On pourrait obtenir de tels graphes comme des *lieux géométriques* (en plaçant un point P sur l'axe des abscisses, en calculant le point correspondant F du graphe à partir de P, puis on créant le lieu des points F quand P varie sur l'axe des x). La résultat obtenu est illustré à la figure 6.5.15.

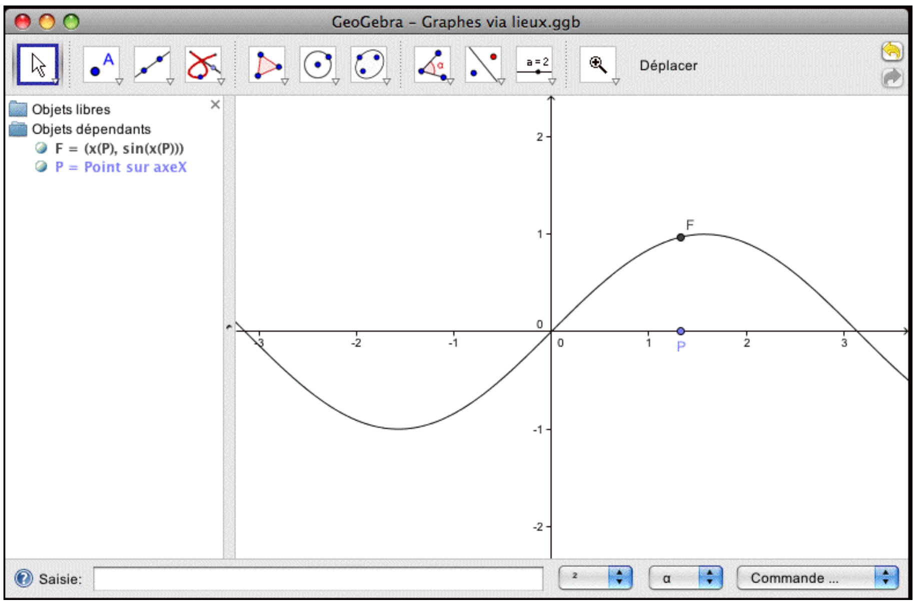


Figure 6.5.15 Le graphe de la fonction sinus, obtenu comme lieu géométrique. Notez l'utilisation de l'expression x(*P*) pour obtenir l'abscisse du point *P*.

Mais *GeoGebra* nous permet de procéder beaucoup plus simplement. Voyons, par exemple, comment représenter la famille de fonctions $a\sin\big(b(x-h)\big)+k$, tel qu'illustré à la figure 6.5.16.

- À l'aide de l'outil « Curseur », on crée quatre curseurs (sortes de barres de défilement) pour représenter nos quatre paramètres a, b, h et k. Pour chaque curseur, on clique d'abord dans la zone graphique pour spécifier sa position, puis on donne les renseignements voulus dans la fenêtre de dialogue qui apparaît ensuite : nom, intervalle (onglet « Intervalle ») et largeur de la barre de défilement (onglet « Curseur »).
- On tape ensuite la définition générale d'une fonction de la famille dans le champ de saisie :

  f(x)=a*sin(b*(x-h))+k

  Tout ira bien car la variable x est bien reconnue, et les quatres paramètres ont été définis.

C'est terminé! On obtient le graphe d'une fonction de la famille, qu'on peut faire varier en modifiant les valeurs de nos quatre curseurs.

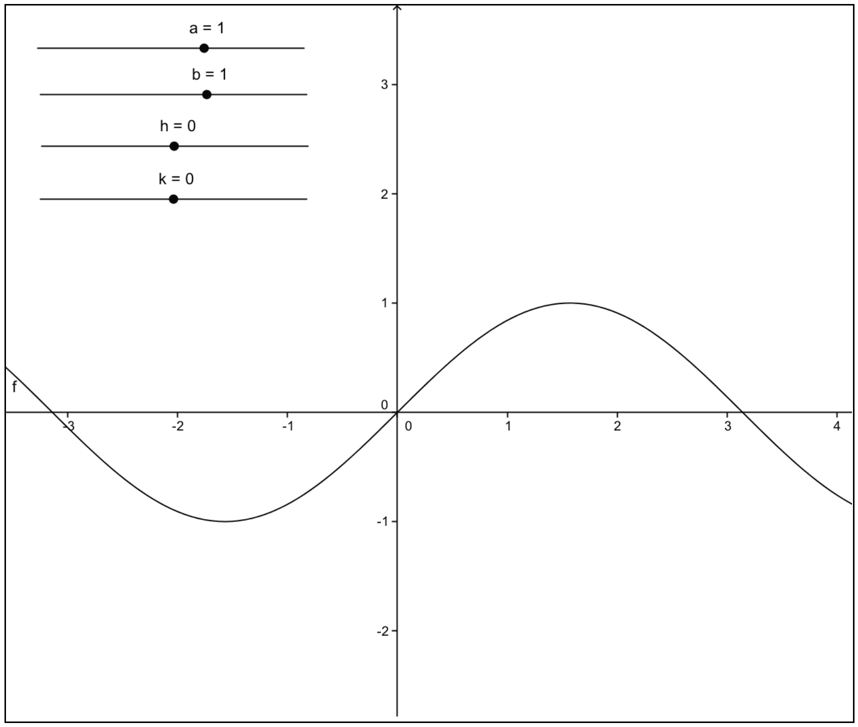


Figure 6.5.16 Représentation de la famille de fonctions *a* sin(*b*(*x*-*h*)) + *k*.

Il arrive parfois qu'une fonction est définie non pas par une expression algébrique mais bien par un processus de fabrication. Les fonctions *cosinus* et *sinus*, par exemple, sont parfois définies via un enroulement de l'axe des x sur le cercle unité centré à l'origine. On imagine qu'on amène l'abscisse 0 de l'axe des x sur le point (1,0) du cercle et qu'on enroule l'axe sur le cercle. Chaque point x de l'axe est amené, de cette façon, à correspondre au point (cos x , sin x) du cercle. Voyons maintenant comment réaliser ceci dans *GeoGebra*. (Le résultat final apparaît à la figure 6.5.17.)

- On fait tout d'abord apparaître les axes, si ce n'est déjà fait.
  On peut aussi modifier les graduations des axes (partant d'un clic droit dans la zone graphique) pour souligner l'importance de π dans notre définition.
- On place ensuite un point P sur l'axe des x.
- Dans le champ de saisie, on définit les points O=(0,0) et U=(1,0), ainsi que le nombre *abscisseP*=x(P).
- À l'aide de l'outil « Cercle (centre-point) », on définit le cercle centré en O et passant par U.
- On veut maintenant enrouler l'axe des abscisses autour du cercle, et trouver ainsi le point P' où aboutira le point P. Pour ce faire, on choisit l'outil « Report de mesure »[8], puis on désigne successivement le nombre *abscisseP*, le centre O et le point U. Un point (que nous nommerons P') apparaît alors.
  Une remarque en passant : bien que l'outil se nomme « Report de mesure », il tient compte du

[8] L'outil « Report de mesure » est inclus dans les fichiers fournis avec le présent chapitre. Sa définition fera aussi l'objet d'un exercice.

signe du nombre utilisé comme paramètre. Dans notre cas, l'enroulement se fait dans le sens anti-horaire quand le nombre est positif, et dans le sens horaire sinon.

- On construit ensuite le point *S* comme l'intersection de la perpendiculaire à l'axe des abscisses passant par *P* et de la perpendiculaire à l'axe des ordonnées passant par *P'*. Le lieu du point *S* quand le point *P* parcourt l'axe des abscisses sera le graphe de la fonction sinus. Si on le désire, on peut cacher les objets intermédiaires de notre construction.

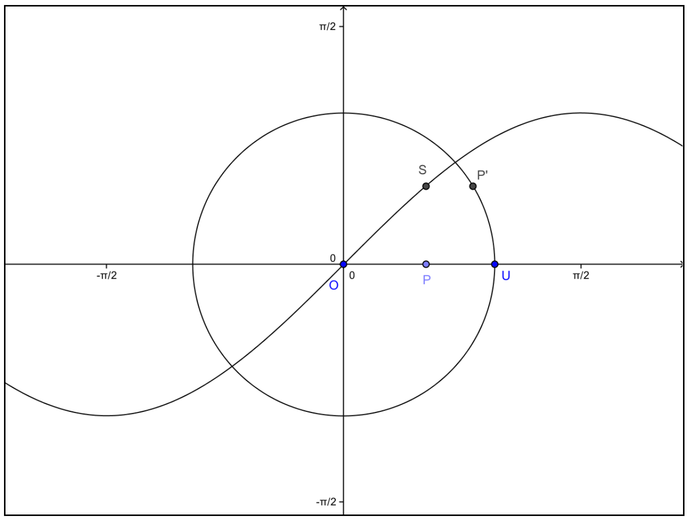


Figure 6.5.17 Le graphe de la fonction sinus, obtenu via une fonction d'enroulement.

Le lecteur pourra, en guise d'exercice, construire semblablement le graphe de la fonction cosinus.

6.6 Aller plus loin avec GeoGebra

GeoGebra possède certaines commandes permettant de définir des figures par programmation[9]. Nous avons vu précédemment qu'il était possible de faire des constructions conditionnelles (avec l'instruction **Si**). Nous allons maintenant voir qu'il est aussi possible de faire des constructions itérées, et ce de deux façons différentes : avec l'instruction **Séquence** et en utilisant le *tableur* intégré. Pour illustrer notre propos, nous utiliserons la construction décrite à la figure 6.6.18.

Voyons tout d'abord comment procéder en utilisant l'instruction **Séquence** :

- On crée tout d'abord les deux points *A* et *B*, ainsi que le curseur *n* (qui prendra des valeurs de 2 à 50 avec incrément de 1).
- On définit ensuite le nombre *d* comme étant la distance entre les points *A* et *B*.
- On définit alors le nombre $r = d\sin\left(\frac{\pi}{n}\right)$. Ce nombre sera le rayon de nos petits cercles. (Pourquoi ? Indice : les centres des petits cercles seront les sommets d'un polygone régulier à *n* côtés.)
- On crée ensuite le cercle *C* de centre *B* et de rayon *r*, puis on cache ce cercle.
- Dans le champ de saisie, on tape enfin

Séquence[Rotation[C,k*2π/n,A],k,0,n-1]

[9] Pour les personnes familières avec *Cabri*, disons simplement que *GeoGebra* offre une bien meilleure performance dans ce domaine.

Examinons de plus près cette dernière commande. Jetons d'abord un coup d'œil sur la commande **Rotation**, dont la forme générale est

Rotation[*objet*, *angle*, *centre*].

Dans notre cas particulier, il s'agira de faire subir à notre cercle *C* une rotation d'un angle $k\frac{2\pi}{n}$ autour du point *A*. Mais quel est ce nombre *k* qui apparaît ici? Il provient de l'instruction **Séquence**, qui le fait varier (dans notre cas particulier) de 0 à *n*-1. La forme générale de cette instruction (avec une variable nommée *k*)

Séquence[*objet_k*, k, *valeur_initiale_de_k*, *valeur_finale_de_k*]

ressemble en fait à une instruction **For**, telle que rencontrée dans *VBA* (voir les sections 2.6 et 4.4), qui prendrait la forme suivante :

For k = *valeur_initiale_de_k* To *valeur_finale_de_k*
Créer l'objet_k
Next k

Le résultat final de notre instruction composée correspond donc bien à la figure 6.6.18.

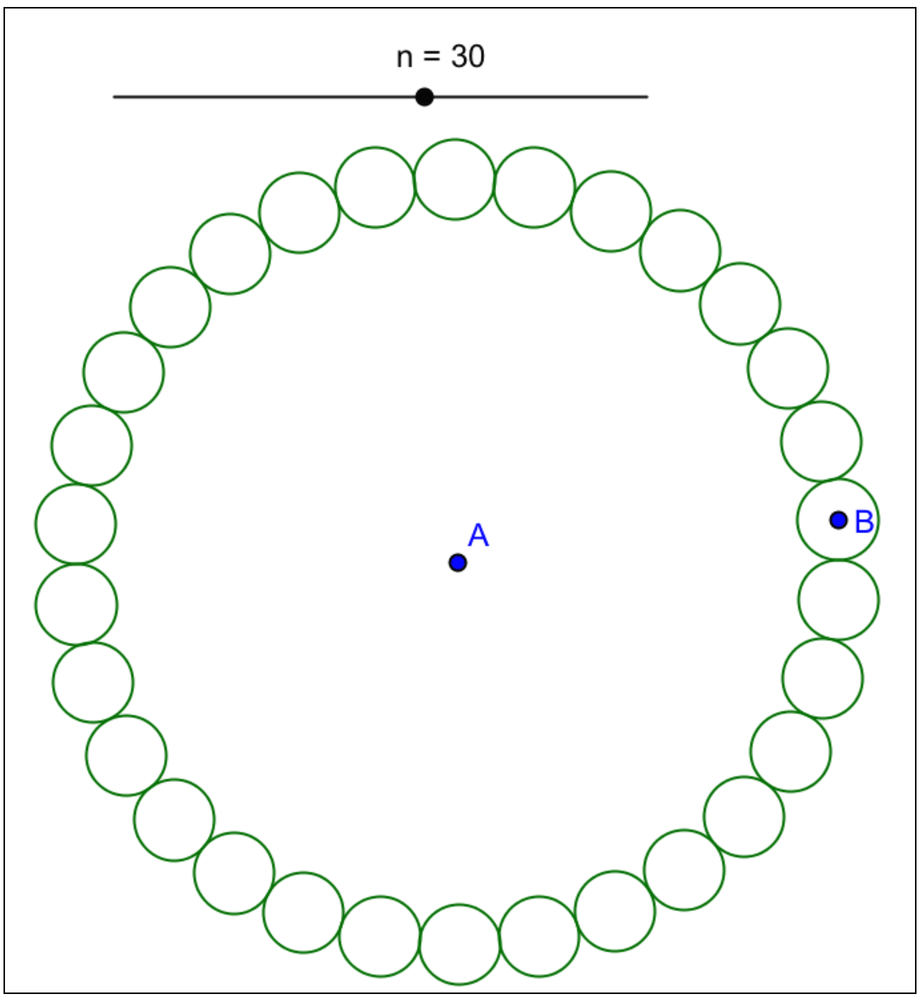


Figure 6.6.18 Figure consistant en *n* cercles tangents de rayons égaux dont les centres sont tous sur un cercle de centre *A* et passant par *B*.

Voyons maintenant comment utiliser le *tableur* pour faire la figure 6.6.18. On commence, comme précédemment, par créer les points *A* et *B* ainsi que le curseur *n*, par définir les nombres *d* et *r*, et par créer le cercle *C* (qui restera invisible). Puis on fait apparaître, si nécessaire, le *tableur* (via le menu « Affichage » ➤ « Tableur »). La figure 6.6.19 montre ce que l'on obtiendra à la fin, et nous permet de faire les observations suivantes :

- Ce tableur utilise la notation A1. La notation L1C1 n'est tout simplement pas disponible.
- Le tableur de la figure 6.6.19 affiche les *valeurs* contenues dans ses cellules. (Pour choisir d'afficher les *valeurs* et non les *formules*, utiliser le menu « Options » ➤ « Algèbre » ➤ « Valeur ».) On constate dans notre exemple que les cellules peuvent contenir des nombres, mais aussi des cercles. En fait, une cellule peut contenir tout objet connu de *GeoGebra*.
- Tout objet géométrique contenu dans une cellule du *tableur* est automatiquement affiché, et son nom est précisément le nom de sa cellule. (Bien entendu, on pourra, si on le désire, cacher l'objet ou son étiquette dans la zone graphique.)
  Réciproquement, si on crée un objet dont le nom correspond à celui d'une cellule, alors cet objet sera automatiquement placé dans cette cellule.
- Dans notre exemple, on peut voir dans la zone graphique des cercles nommés *B2*, … , *B23* correspondant aux équations placées dans les cellules correspondantes. Si on veut cacher tous ces noms, il suffira de faire un clic droit dans le tableur, de sélectionner l'item « Propriétés… » dans le menu local, de sélectionner les items *B2* à *B23* dans la liste de gauche, et de décocher la case « Afficher l'étiquette » de l'onglet « Basique ».

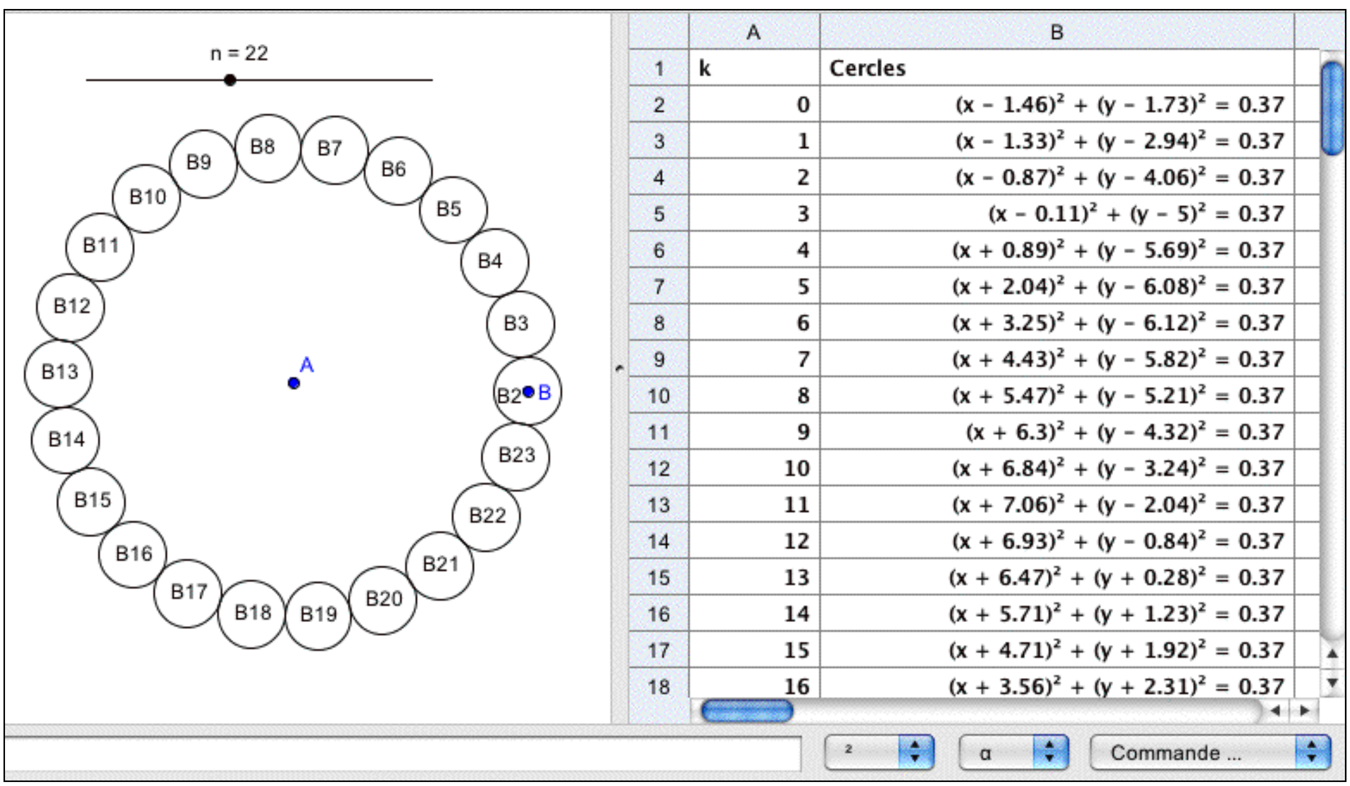


Figure 6.6.19 Les cellules du tableur de GeoGebra peuvent contenir des nombres (colonne A) ou des objets géométriques (cercles dans la colonne B).

La figure 6.6.20 nous montre les formules ayant servi à placer les valeurs dans les colonnes *A* et *B*. Voici comment on a procédé.

- Pour placer le texte « k » dans la cellule A1, on a tapé **A1="k"** dans le champ de saisie. De même avec **B1="Cercles".**
- Pour placer le nombre 0 dans la cellule A2, on a tapé **A2=0** dans le champ de saisie.
- On a ensuite tapé **A3=A2+1** dans le champ de saisie. Rappelons-nous qu'il s'agit ici d'une référence *relative* : on demande d'additionner 1 à la cellule immédiatement en haut.
- On a alors pu recopier vers le bas : on a sélectionné (par un clic) la cellule A3, puis on a saisi avec la souris le petit carré en bas à droite de cette cellule pour le glisser jusqu'à la cellule A51. Bien entendu la recopie s'est faite en tenant compte des références relatives.
- Pour terminer, on a tapé **B2=Si[A2<n, Rotation[C, A2*2*pi/n, A]]** dans le champ de saisie et on a recopié vers le bas jusqu'à la cellule B51. Comparons au passage avec notre approche utilisant l'instruction **Séquence** : la variable *k* (variant entre 0 et *n*-1) est remplacée par un bloc de cellules A2…A51 dont on ne retient (via l'instruction **Si**) que les *n* premières (de 0 à *n*-1).

  Notons au passage que, bien que les objets *n* et *C* ne soient pas définis dans le tableur, ils peuvent être utilisés dans celui-ci car ils sont connus de *GeoGebra*.

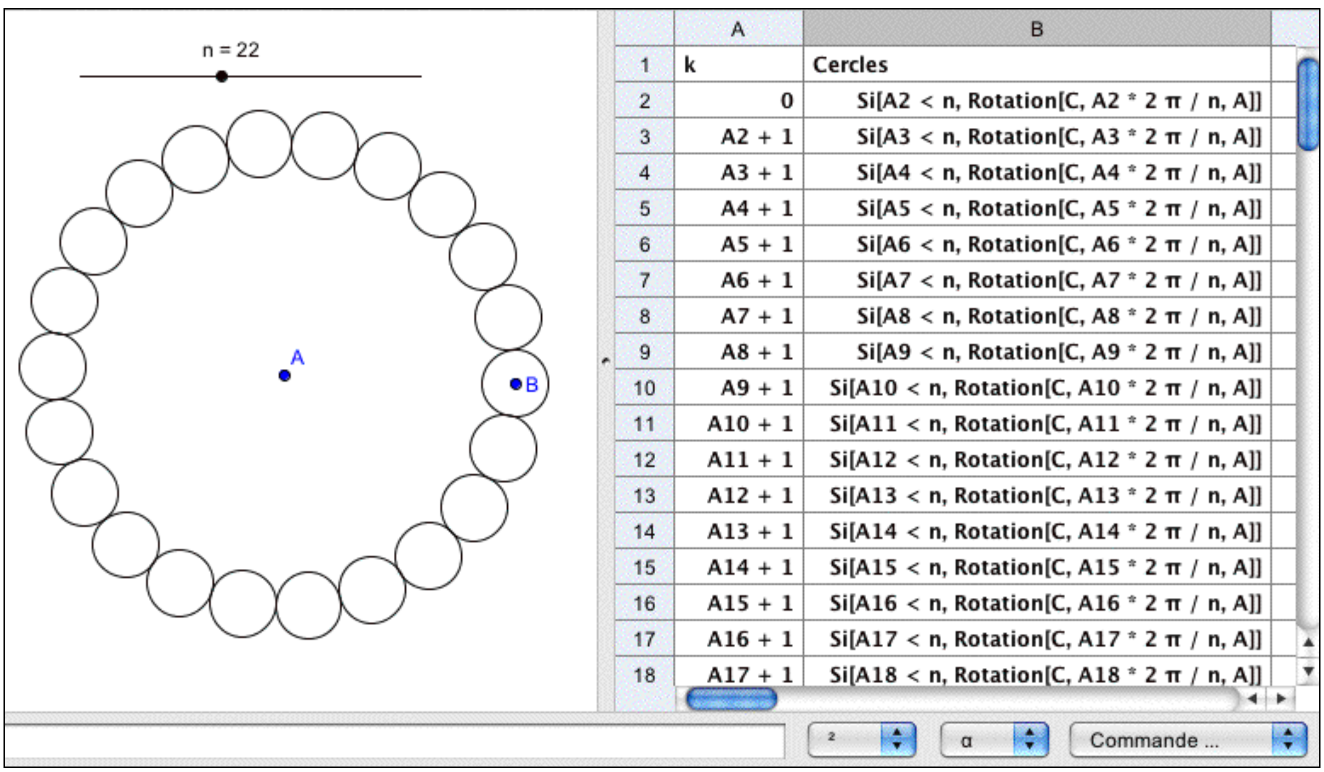


Figure 6.6.20 Vue du tableur montrant les formules définissant le contenu des cellules.

Ici se termine notre brève exploration de *GeoGebra*. Si vous voulez la poursuivre, le site Web de *GeoGebra* contient plusieurs ressources pour vous assister.

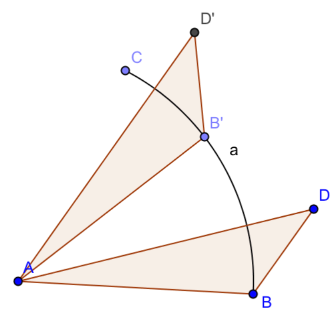
6.7 Exercices

1- *Construire un carré de différentes façons.*
Utilisez GeoGebra pour construire un carré étant donné :
   a) Un côté.
   b) Une diagonale.
   c) Le centre et un point du cercle circonscrit.
   d) Le centre et un point du cercle inscrit.
   e) En utilisant seulement les outils : Nouveau point, Segment entre deux points, Cercle (centre-point), Intersection entre deux objets.

2- *Construire un parallélogramme*
Étant donnés 3 points A, B et C non colinéaires, construisez un parallélogramme ABCD.
Que se passe-t-il lorsque A, B et C deviennent colinéaires? Si le point D existe encore bravo! Sinon modifiez votre construction pour que le point D continue à exister, (évidemment toujours avec AB = CD et BC = AD).

3- *Construire les tangentes à un cercle*
Étant donnés un cercle et un point *P* hors de ce cercle, construire les deux tangentes à ce cercle passant par *P*.

4- *Construire le transformé d'un triangle par une rotation.*
   - À partir de deux points A et B, construisez un arc *a* d'extrémités B et C, où C est un point sur le cercle de centre A et passant par B.
   - Construisez ensuite un triangle ABD où D est un nouveau point.
   - À partir d'un point B', mobile sur l'arc BC, construisez le triangle AB'D' qui est l'image du triangle ABD par la rotation de centre A qui amène B sur B'.



5- *Faire une hypothèse et la vérifier expérimentalement.*
Avec GeoGebra :
   - Construisez un quadrilatère ABCD.
   - Construisez les milieux E, F, G et H des côtés AB, BC, CD et DA respectivement.
   - Construisez le quadrilatère EFGH.

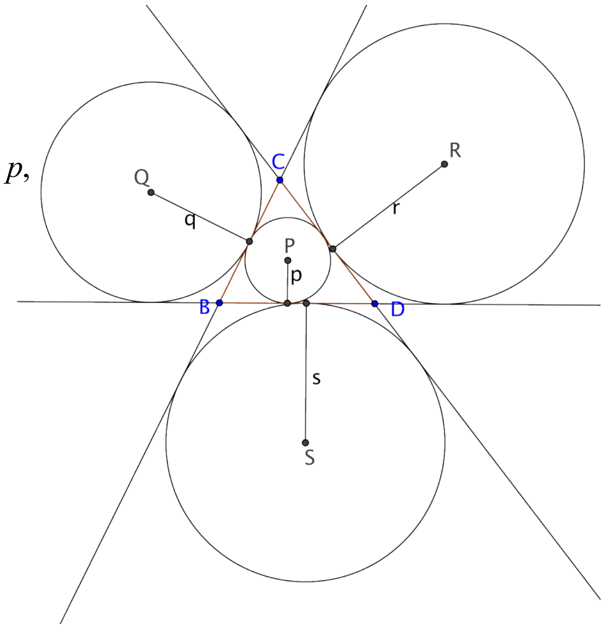
Faites une hypothèse sur la nature du quadrilatère EFGH et utilisez GeoGebra pour la vérifier.

6- *Vérifier expérimentalement une hypothèse.*
Un théorème de géométrie élémentaire affirme que :
« *Dans tout triangle le carré d'un côté opposé à un angle aigu est égal à la somme des carrés des deux autres côtés moins deux fois le produit de l'un deux par la projection de l'autre sur lui* »
Utilisez GeoGebra pour vérifier expérimentalement ce théorème.
Faites une hypothèse sur ce que serait un théorème analogue pour le côté opposé à un angle obtus et vérifiez expérimentalement votre hypothèse.

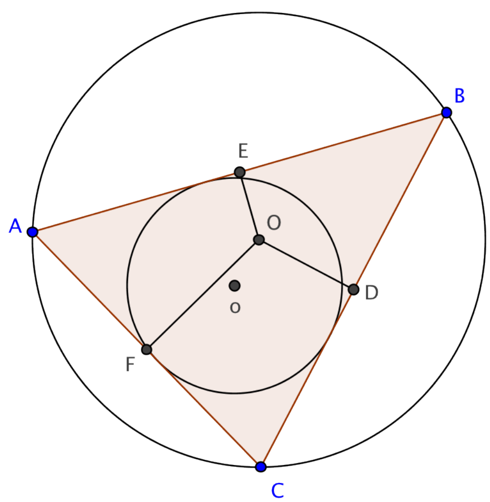
7- *Déterminer expérimentalement une relation.*

- Construisez un triangle BCD.
- Construisez le cercle, de centre P et de rayon *p*, inscrit dans le triangle BCD.
- Construisez les cercles, de centres Q, R, S et de rayons respectifs q, r, et s, ex-inscrits au triangle BCD.
- Mesurez l'aire A du triangle.
- Calculez M = $\sqrt{p.q.r.s}$

Quelle relation y a-t-il entre A et M ?



8- *Vérifier expérimentalement le « Théorème japonais » de Lazare Carnot.*

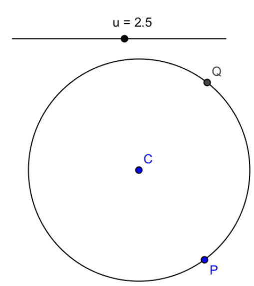


Étant donné un triangle ABC, son cercle circonscrit de centre O et de rayon R et son cercle inscrit de rayon r, on trace les segments OD, OE et OF perpendiculaires aux trois côtés. Un théorème de Lazare Carnot (dit Théorème japonais…) établit une relation du type :

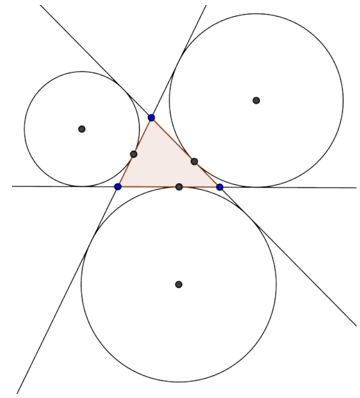
$$\pm \text{OD} \pm \text{OE} \pm \text{OF} = \text{R} + \text{r}$$

et donne une condition géométrique pour savoir où placer les signes plus et moins en fonction du cas de figure.
Sachant qu'il y a 0 où 1 signe moins, utilisez GeoGebra pour déterminer la condition qui permet de d'associer un cas de figure avec la relation correspondante.

9- *Création de la macro « Report de mesure »*
Créer une macro qui, étant donnés (dans l'ordre) un nombre *u* (pouvant être négatif), le centre *C* d'un cercle et un point *P* sur ce cercle, construit un point *Q* sur ledit cercle tel que la distance signée de l'arc entre *P* et *Q* est égale à *u*. (Par convention, on reporte une distance |*u*| dans le sens anti-horaire si *u* est positif, et dans le sens horaire si *u* est négatif.)

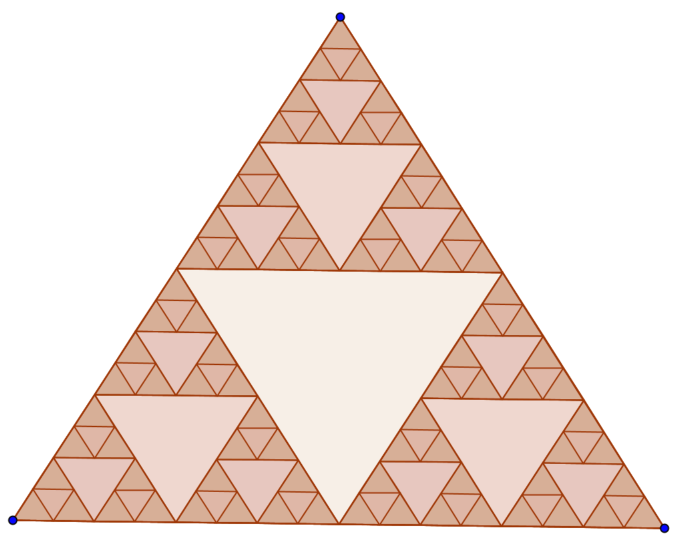


10- *Réaliser une macro-construction.*
Réalisez une macro-construction qui, à partir de 3 points non alignés, trace les trois cercles ex-inscrits au triangle, tel qu'illustré dans la figure ci-contre.
Remarquez que dans la figure on a tracé les prolongements des côtés pour montrer qu'ils sont bien tangents aux cercles, mais ce n'est pas une exigence de l'énnoncé



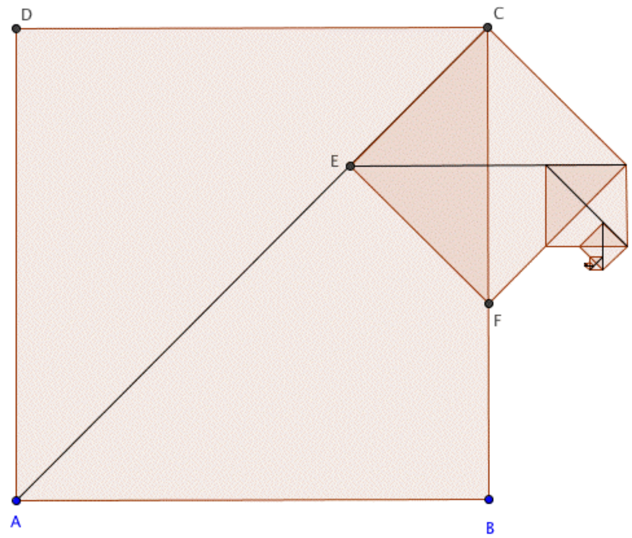
11- *Utiliser une boîte noire.*
Dans le contexte de *GeoGebra* on appelle « boîte noire » une figure dont une partie de la construction est cachée. Une telle figure est souvent utilisée en éducation, dans le but d'amener l'utilisateur à expérimenter avec la figure pour arriver à en déduire quelle est la construction cachée.
Une façon de cacher efficacement une construction c'est de la réaliser à l'aide d'une macro-construction. La construction réalisée par la macro ne peut alors être visualisée.
Vous trouverez un exemple d'une telle « boîte noire sur le site du livre, dans le dossier « BoîteNoire » des fichiers exemples du chapître.
Ouvrez la figure « Figure ». Elle montre un triangle ABC et un point P qui est construit à partir du triangle par la macro « Mystère ». Expérimentez avec le triangle pour en déduire une construction du point P.

12- *Itérer une construction à l'aide d'une macro-construction : le triangle de Sierpinski.*
On désire utiliser GeoGebra pour construire la figure suivante :



Étant donné un triangle ABC, réalisez une macro qui construit les triangles AMP, BMN et CNP où M, N et P sont les milieux de AB, BC et CA respectivement. Cette macro doit utiliser les sommets A, B, C comme objets initiaux.
Utilisez votre macro pour construire la figure ci-dessus.

13- *Itérer une construction à l'aide d'une macro-construction.*
La figure ci-dessous a été obtenue en itérant la construction suivante : à partir d'un carré ABCD, on trace la diagonale AC sur laquelle on reporte AE un segment congru à AB, puis on construit un nouveau carré de côté EC.



Réalisez tout d'abord une macro-construction qui, à partir des quatre sommets d'un carré, construit le carré suivant. Utilisez ensuite cette macro pour obtenir la figure ci-dessus

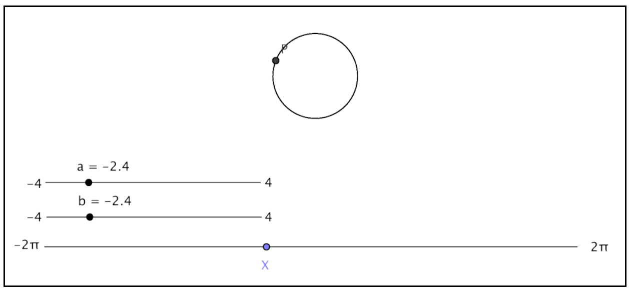
14- *Construire des coniques comme lieux de points.*

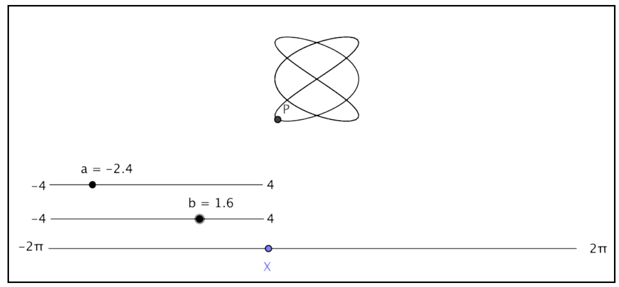
a) On peut définir une ellipse comme le lieu des points P tels que :
$$PF_1 + PF_2 = 2a$$
où F_1 et F_2 sont deux points distincts (les foyers) et a une constante positive ($F_1F_2 < 2a$).
F_1, F_2 et un segment (dont la longueur représente la constante 2a) étant donnés, utilisez la définition ci-dessus pour construire l'ellipse correspondante. Que se passe-t-il si $2a$ devient inférieur à F_1F_2 ?

b) On peut définir une hyperbole comme le lieu des points P tels que :
$$|PF_1 - PF_2| = 2a$$
où F_1 et F_2 sont deux points distincts (les foyers) et a une constante positive ($2a < F_1F_2$).
F_1, F_2 et un segment (dont la longueur représente la constante 2a) étant donnés, utilisez la définition ci-dessus pour construire l'hyperbole correspondante.

15- *Contruire des graphes comme lieux géométriques*
Vous avez vu, dans la section 6.5, comment construire le graphe de la fonction sinus via une fonction d'enroulement. Utilisez la même méthode pour tracer le graphe des fonctions cosinus et tangente.

16- *Contruire un lieu géométrique qui dépend de paramètres*
Construisez le lieu des points de coordonnées $(\cos(ax), \sin(bx))$, où x varie de -2π à 2π. Les paramètres *a* et *b* devront pouvoir varier entre -4 et 4.
Voici deux lieux obtenus pour différentes valeurs des paramètres a et b.





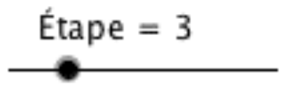
17- *Contruire un lieu géométrique qui dépend de paramètres.*
En vous basant sur les indications données dans la section 6.5 :

1. construisez le graphe de la fonction $a\sin(b(x-h))+k$. Les valeurs des paramètres *a*, *b*, *h*, *k* seront contrôlées par le déplacement de points sur des segments (par exemple $-2 \le a \le 2$, $-10 \le b \le 10$, $-2 \le h \le 2$, $-1 \le k \le 1$).
2. construisez le graphe de la fonction $\dfrac{ax+b}{cx+d}$ où, de la même façon, les valeurs des paramètres *a*, *b*, c, *d* seront contrôlées par le déplacement de points sur des segments.

6.8 Projets

1- *Construire une figure GeoGebra comportant plusieurs étapes.*

Quelques fois, lorsqu'on fournit une figure *GeoGebra* , on aimerait y inclure un mode d'emploi sous forme d'étapes successives. Ceci peut se réaliser en faisant appel à un curseur pouvant prendre des valeurs entières entre 1 et *n* (où *n* est le nombre d'étapes désiré).

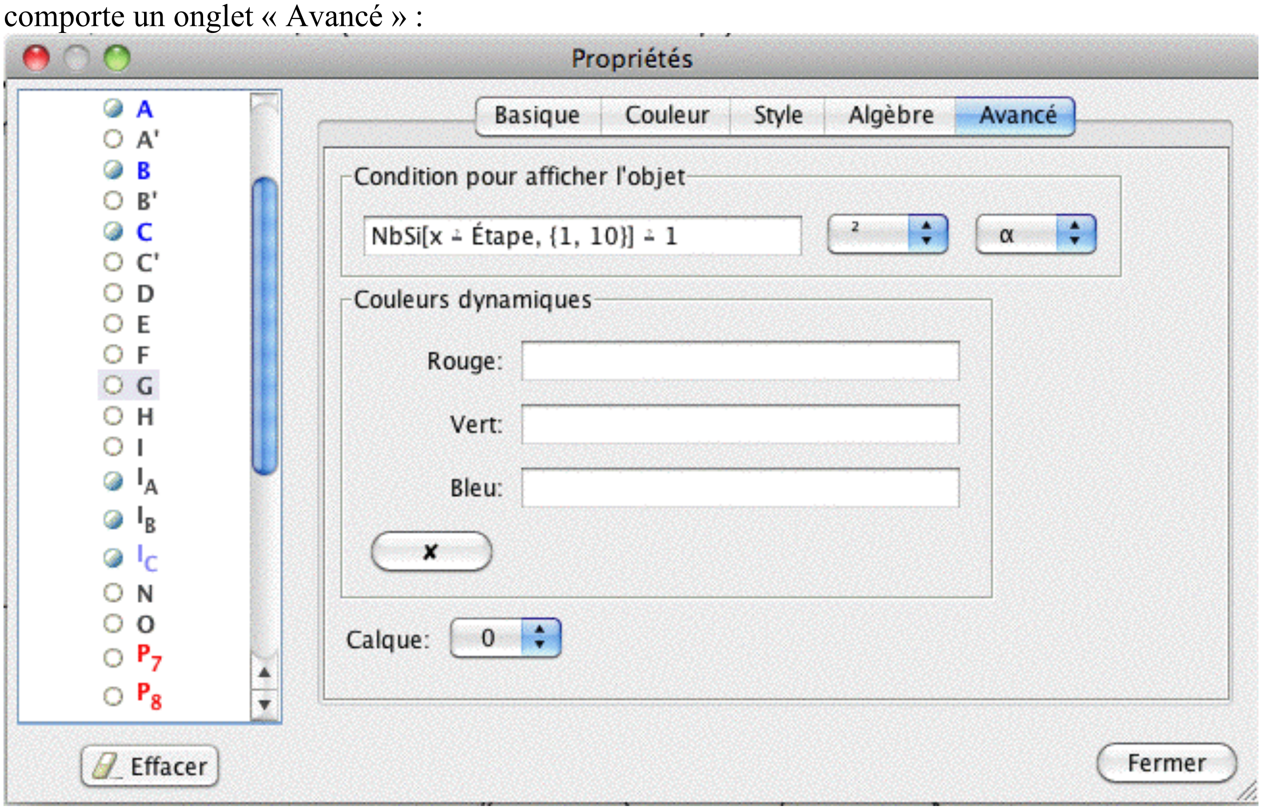


Selon les valeurs du curseur, certains objets pourront être affichés ou non, en particulier des textes pouvant donner des explications sur la figure affichée, décrire des actions à poser, ou fournir de l'aide à l'utilisateur. Pour un exemple d'une telle réalisation, voir

http://www.math.uqam.ca/~expresso/LivreLogicielsOutils/Projet1_GGB.html

Le principe

Un clic droit sur un objet de *GeoGebra* permet d'afficher sa fenêtre de propriétés, qui comporte un onglet « Avancé » :

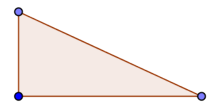


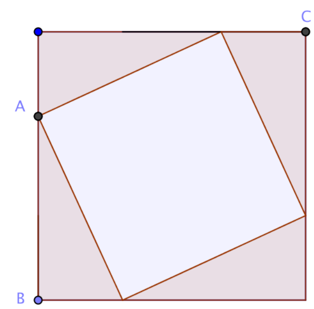
On voit ci-dessus que l'objet « G » sera affiché exactement aux étapes 1 et 10. (Pour plus de détails, notamment sur l'instruction « NbSi », veuillez consulter l'aide du logiciel.)

Le projet proposé ici est plutôt ouvert : il s'agit de réaliser un tel document *GeoGebra* « à étapes », selon un scénario de votre choix.

2- *Construire une démonstration « visuelle » du théorème de Pythagore.*

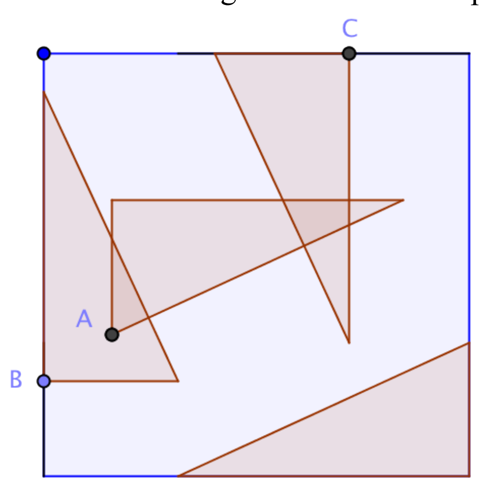
L'objectif de ce projet est de réaliser une figure *GeoGebra* illustrant une démonstration du théorème de Pythagore.

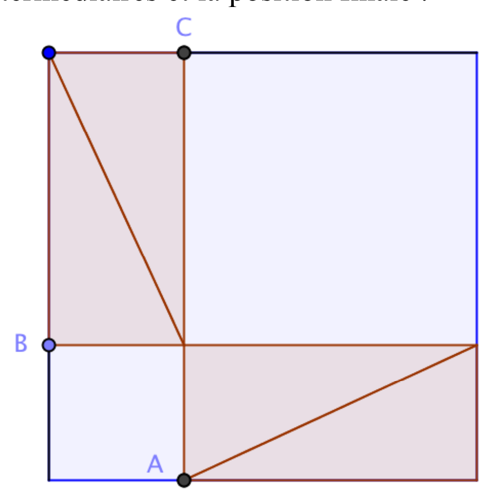




- Commencez par construire un triangle rectangle que vous pourrez déplacer en déplaçant le sommet de l'angle droit et dont vous pourrez modifier les dimensions en déplaçant les extrémités de l'hypothénuse.
- Construisez ensuite un carré dont le côté a pour mesure la somme des longueurs des côtés de l'angle droit du triangle.
- Finalement, par dessus le carré, construisez quatre triangles rectangles, comme sur la figure ci-contre, de sorte que :
    - le point C puisse être déplacé sur le côté horizontal du carré en entrainant avec lui le triangle dont il est le sommet de l'angle droit.
    - le point B puisse être déplacé sur le côté vertical gauche du carré en entrainant avec lui le triangle dont il est le sommet de l'angle droit.
    - le point A puisse être déplacé diagonalement en entrainant avec lui le triangle qui est dans le coin supérieur gauche.

Voici ci-dessous des figures illustrant des positions intermédiaires et la position finale :





Dans la figure initiale le carré central est construit sur l'hypothénuse du triangle et, lorsqu'on compare avec la figure finale, on voit que son aire doit être égale à la somme des aires des deux carrés construit sur les côtés de l'angle droit.

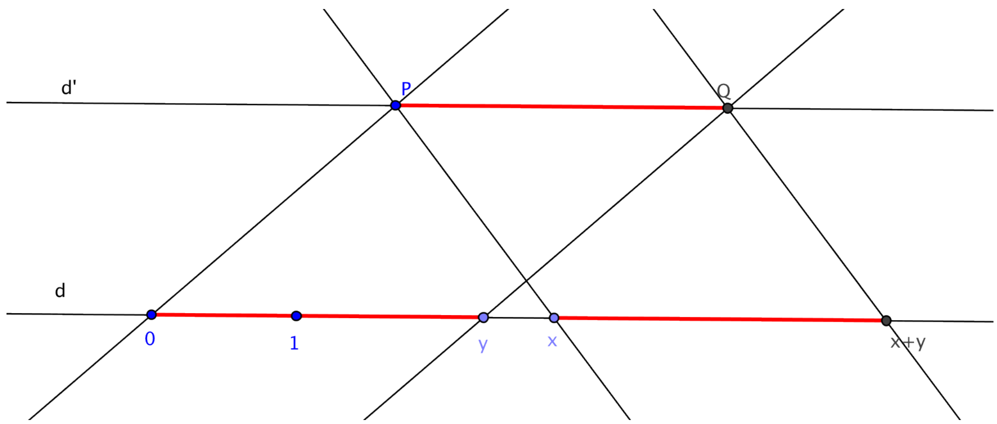
3- *Interpréter géométriquement les opérations arithmétiques.*

Notre point de départ sera une droite *d* sur laquelle se trouvent deux points distincts. Nous allons considérer que cette droite est une droite numérique, et que nos deux points distincts désignent l'origine (0) et l'unité (1).

L'addition

Étant donnés deux points x et y sur d, nous allons construire le point x+y sur *d* comme suit :

- Choisissons un point P hors de la droite *d*. Par P, nous menons une parallèle *d'* à *d*.
- Nous traçons ensuite successivement les droites suivantes :
  - la droite $\overleftrightarrow{0P}$ passant par les points 0 et P
  - la droite d_1 passant par y et parallèle à $\overleftrightarrow{0P}$
    Soit Q le point d'intersection de d_1 et de *d'*
  - la droite $\overleftrightarrow{Px}$ passant par P et x
  - la droite d_2 passant par Q et parallèle à $\overleftrightarrow{Px}$
    x+y sera le point d'intersection de d_2 et de *d*



Vérifiez expérimentalement que

- le point x+y est indépendant du choix du point P
- si x et y sont « positifs » (c.-à.-d. sur la demi-droite issue de 0 et passant par 1), alors distance(0,x) + distance(0,y) = distance(0,x+y)
- si x et y ne sont pas tous deux positifs, alors tout continue de bien se comporter.

La soustraction

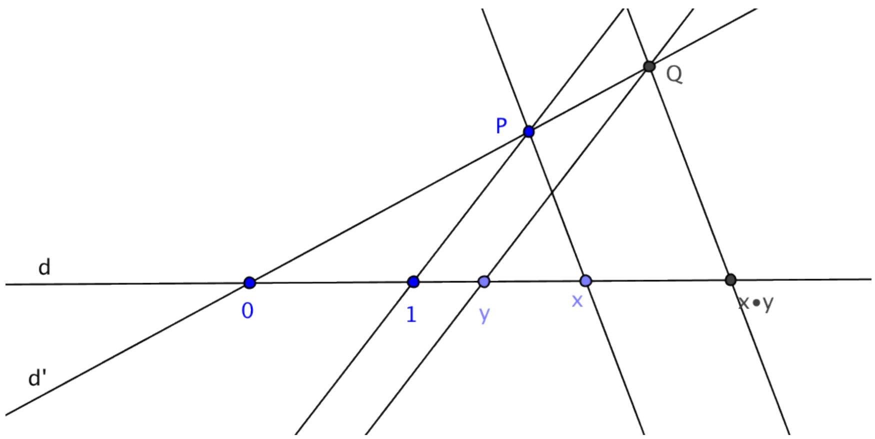
Au lieu de définir une soustraction, il est plus aisé de définir l'opposé -x de x (comme l'image de x par une symétrie de centre 0) et de poser

$$x - y = x + (-y).$$

La multiplication

Étant donnés deux points x et y sur *d*, nous allons construire le point x×y sur *d* comme suit :

- Choisissons un point P hors de la droite *d*. Soit *d'* la droite $\overleftrightarrow{0P}$.
- Nous traçons ensuite successivement les droites suivantes :
  - la droite $\overleftrightarrow{1P}$ passant par les points 1 et P
  - la droite d_1 passant par y et parallèle à $\overleftrightarrow{1P}$
    Soit Q le point d'intersection de d_1 et de *d'*
  - la droite $\overleftrightarrow{Px}$ passant par P et x
  - la droite d_2 passant par Q et parallèle à $\overleftrightarrow{Px}$
    x×y sera le point d'intersection de d_2 et de *d*



Vérifiez expérimentalement que

- le point x×y est indépendant du choix du point P
- si x et y sont « positifs » (c.-à.-d. sur la demi-droite issue de 0 et passant par 1), alors $\frac{\text{distance}(0,\ x)}{\text{distance}(0\ ,1)} \times \frac{\text{distance}(0\ ,y)}{\text{distance}(\ 0{,}1)} = \frac{\text{distance}\ (0{,}x \times y)}{\text{distance}(0\ ,1)}$
- si x et y ne sont pas tous deux positifs, alors la règle des signes est respectée. En particulier, on a : (-x) × (-y) = x×y.

La division

Au lieu de définir une division, il est plus aisé de définir l'inverse multiplicatif x^{-1} de x, et de poser

$$x \div y = x \times y^{-1}.$$

En vous inspirant de la définition de la multiplication ci-dessus, définissez l'inverse multiplicatif x^{-1} de x, et vérifiez ensuite expérimentalement que

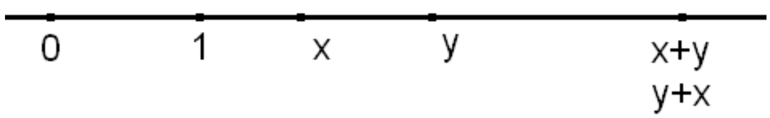
- $\frac{\text{distance}(0,\ x^{-1})}{\text{distance}\ (0{,}1)} = 1 \div \frac{\text{distance}(0,\ x)}{\text{distance}\ (0{,}1)}$
- $x \times x^{-1} = 1$.

Des macros
Définissez ensuite les macros suivantes :

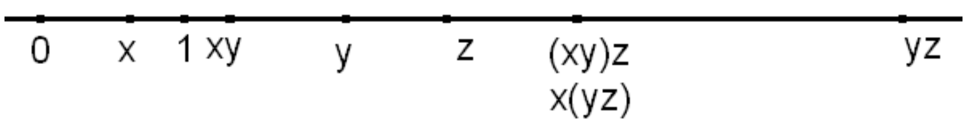
- Opposé : étant donnés l'origine 0, l'unité 1 et un point x, construit -x
- Inverse : étant donnés l'origine 0, l'unité 1, le point P et un point x, construit x^{-1}
- Somme : étant donnés l'origine 0, l'unité 1, le point P ainsi que deux points x et y, construit x+y
- Produit : étant donnés l'origine 0, l'unité 1, le point P ainsi que deux points x et y, construit $x \times y$

et utilisez-les pour vérifiez les propriétés suivantes :

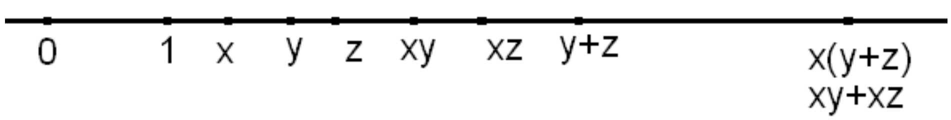
- la commutativité de l'addition : $x + y = y + x$



- l'associativité de la multiplication : $x \times (y \times z) = (x \times y) \times z$



- la distributivité de la multiplication sur l'addition : $x \times (y+z) = (x \times y) + (x \times z)$



- ainsi que toute autre propriété qu'il vous intéressera de vérifier.

4- *Construire le cercle des neuf points.*
Commencez par tracer un triangle ABC, puis :

- Construisez les milieux des côtés et G le point d'intersection des médianes.
- Construisez les pieds des hauteurs (sur les côtés ou leurs prolongements) et H le point d'intersection des hauteurs.

Vérifiez, expérimentalement, que les six points, milieux des côtés et pieds des hauteurs, sont situés sur un même cercle. On appelle ce cercle « cercle d'Euler ». Si ce n'est déjà fait, construisez le ainsi que son centre E.

- Construisez le cercle circonscrit au triangle et son centre O.

Énoncez une propriété géométrique liant les quatre points G, H, E et O.

- Construisez les cercles inscrit et exinscrits au triangle.

Quelle propriété lie ces quatre cercles au cercle d'Euler?

- Construisez les intersections des hauteurs avec le cercle d'Euler.

Quelle propriété ont ces points par rapport aux segment AH, BH et CH?
C'est à cause de cette dernière propriété que le cercle d'Euler est aussi appelé cercle des neuf points.

5- *Création d'une figure résultant d'une double itération*
Dans la section 6.6, nous avons créé des figures consistant en *n* cercles tangents disposés en cercle. Nous vous proposons maintenant d'emboîter *m* telles figures, de façon à obtenir des figures comme celle illustrée ci-dessous.