André Boileau et Maurice Garançon

Outils informatiques pour les enseignants de mathématiques

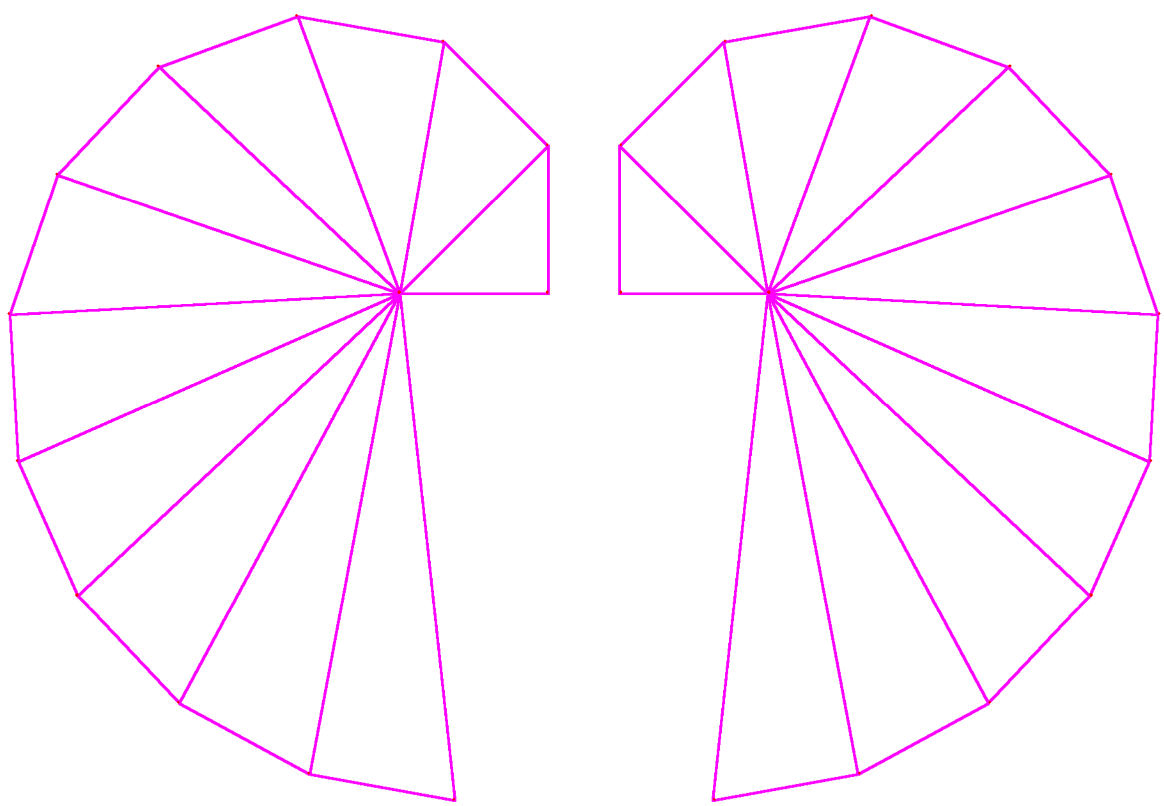


Table des matières

Avant-propos

Il reste bien peu de gens pour soutenir que la technologie n'a pas sa place dans l'enseignement des mathématiques. Mais quelles sont les connaissances et les habiletés technologiques requises pour un enseignant de niveau secondaire? Abordons cette question en racontant un fait vécu...

Nous assistions à un atelier destiné à des professeurs de mathématiques du secondaire. L'animateur, mû par un enthousiasme contagieux, voulait convaincre les participants que *Cabri-géomètre*, un logiciel que nous décrirons dans ce livre, permettait de faire des activités très intéressantes et motivantes pour les élèves, et ce sans demander une grande expertise de la part du professeur. Il avait donc entrepris de prouver ses dires en faisant vivre à son audience des activités de manipulations concrètes assez spectaculaires. Il circulait donc d'une équipe à l'autre, dépannant les personnes en difficulté, et expliquant à d'autres certains comportements assez mystérieux du logiciel utilisé. À la fin de l'atelier, tout le monde avait l'air ravi de l'expérience.

Mais en fait, sans le vouloir, il avait démontré le contraire de ce qu'il prétendait. Sans ses diagnostics experts, plusieurs équipes seraient restées bloquées et n'auraient pu obtenir les beaux résultats qui les avaient charmés. Sans sa connaissance de certains aspects du fonctionnement interne du logiciel, d'autres participants n'auraient pas compris pourquoi ils ne pouvaient obtenir le type d'interaction qu'ils visaient et auraient persisté dans leurs tentatives infructueuses plutôt que d'emprunter des voies alternatives. En fait, les experts oublient parfois la somme des expériences qui leur ont été nécessaires pour acquérir l'aisance dont ils font maintenant preuve et qui semble si naturelle à ceux qui les observent.

Au fil des années, nous avons pu constater que les professeurs qui réussissent le mieux à intégrer les TIC (Technologies d'Information et de Communication) dans leur enseignement sont souvent ceux qui les utilisent pour eux-mêmes. De la même façon qu'il semble judicieux d'acquérir des compétences en mathématiques avant de songer à les enseigner, il nous apparaît naturel de chercher à utiliser soi-même les TIC avant de songer à les utiliser avec ses élèves. Bien entendu, la démarche ne saurait être strictement chronologique car, même une fois qu'on aura commencé à utiliser les TIC dans ses classes, il restera nécessaire (et, souhaitons-le, agréable) de continuer à élargir ses compétences personnelles en ce domaine.

C'est la démarche que nous vous proposons dans ce livre. Il est destiné principalement à des professeurs de mathématiques de niveau secondaire, qu'ils soient en formation initiale ou en exercice. Mais il pourrait aussi être utile plus généralement à quiconque veut utiliser les TIC en mathématiques. Son contenu a été expérimenté, depuis quelques années, dans trois cours d'un programme de formation initiale d'enseignants en mathématiques au secondaire dispensé à l'Université du Québec à Montréal.

L'apprentissage de savoir-faire pratiques occupe la plus grande part de cet ouvrage. Nous avons choisi de susciter ces apprentissages par le biais de la réalisation concrète d'exemples choisis pour mettre en évidence certaines caractéristiques (mathématiques ou technologiques) que nous jugeons pertinentes. Soulignons que ces exemples ont été conçus pour intéresser le lecteur et non ses éventuels élèves, bien qu'il ne soit évidemment pas exclu qu'on puisse les adapter pour ces derniers.

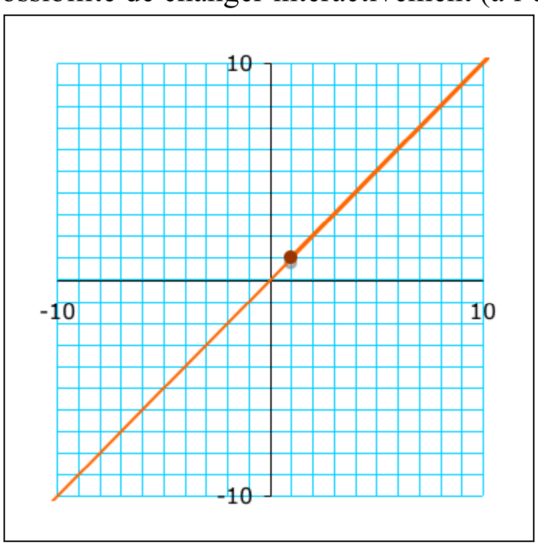
Nous ne négligeons pas pour autant les aspects plus théoriques. En effet, ce livre cherche aussi à répondre à la question suivante : *Que doit-on connaître du fonctionnement interne de ces logiciels afin de pouvoir les utiliser avec profit?* Il ne faut pas perdre de vue qu'un professeur doit non seulement décrire, mais aussi pouvoir expliquer.

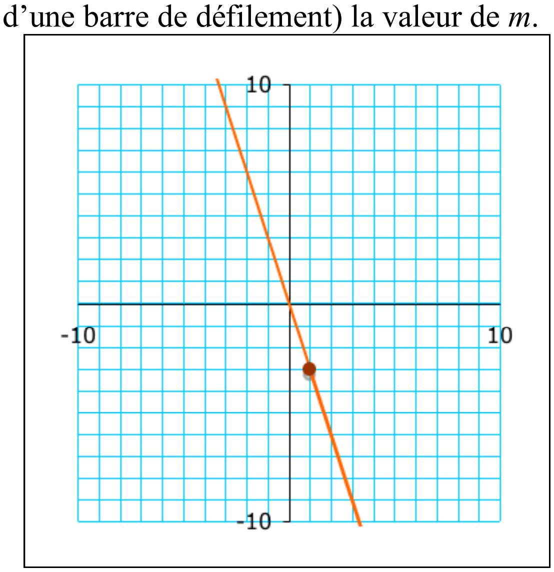
Par contre, bien qu'il puisse nous arriver à l'occasion d'évoquer des situations d'enseignement, soulignons encore que ce livre n'a pas de visées spécifiquement pédagogiques. Il vise un apprentissage des TIC par des professeurs de mathématiques dans des contextes susceptibles de les intéresser eux, mais n'aborde pas directement le problème de la transposition en classe : celle-ci devra faire l'objet d'une réflexion parallèle ou subséquente.

Nous souhaitons aussi que l'enseignant soit capable de créer au besoin ses propres ressources technologiques, pour qu'il ne soit pas limité à utiliser celles des autres. Non pas qu'il doive, tel un Robinson sur son île, développer tous les outils dont il aura besoin : ce serait se priver de réalisations merveilleuses, sans compter les investissements considérables en termes d'imagination, de talent et de temps qui seraient exigés. Mais combien de fois rencontrons-nous des professeurs cherchant longuement un logiciel produisant une représentation particulière d'un phénomène mathématique, pour finalement se contenter d'une solution boiteuse, alors qu'ils auraient facilement pu obtenir exactement ce qu'ils voulaient à l'aide d'un logiciel outil.

Prenons l'exemple de ce professeur voulant illustrer qu'il est parfois plus pertinent d'interpréter une variation de la pente d'une droite comme une dilatation verticale plutôt qu'une rotation. Après avoir cherché longtemps sur Internet, et essayé en vain plusieurs applets sur le sujet, il a fini par produire une feuille de calcul *Excel* (dont vous voyez deux états reproduits ci-dessous) dotée des caractéristiques qu'il recherchait :

- Représentation d'une droite de pente *m* avec mise en évidence de son point d'abscisse 1.
- Possibilité de changer interactivement (à l'aide d'une barre de défilement) la valeur de *m*.





C’est une autonomie de ce type qui est visée par ce livre. Elle est basée sur une bonne connaissance d’une brochette de logiciels outils convenablement choisis. La table des matières de ce livre vous révèlera les logiciels qui ont été retenus, mais laissez-nous vous exposer les raisons de ces choix. Pour qu’un logiciel soit sélectionné, il fallait que ses utilisations possibles soient variées et pertinentes dans l’enseignement des mathématiques, qu’il soit assez versatile pour être utilisé dans plusieurs contextes, et que son fonctionnement soit assez riche pour nécessiter un apprentissage spécifique. À cela se sont ajoutés deux critères plus pratiques : ces logiciels devaient être facilement accessibles dans les écoles secondaires québécoises, et aussi capables de fonctionner tant sous Windows que sur Macintosh[1].

Une remarque en passant : la place donnée dans ce livre aux divers logiciels outils ne reflète pas nécessairement l’importance que nous leur prêtons dans l’enseignement des mathématiques. Ainsi, le fait qu’il y ait deux chapitres sur *Excel* et un seul sur *Cabri* reflète tout simplement qu’il est plus facile d’apprendre à utiliser ce dernier que le premier.

Pour notre part, nous avons supposé que notre lecteur n’était ni un *nul*, ni un *idiot*. Bien que nous ayons rarement utilisé des connaissances mathématiques dépassant celles enseignées au secondaire, nous supposons que celui-ci est capable de suivre un raisonnement et faire preuve d’une certaine autonomie. Il aura déjà une expérience minimale dans la manipulation des ordinateurs et de leurs logiciels de base, lui permettant par exemple d’écrire un texte simple avec un programme de traitement de texte, de renommer le fichier ainsi obtenu, de le recopier sur un bâtonnet de mémoire USB, etc.

On entend parfois dire à la blague que, si un ordinateur ou un logiciel est en vente, c’est qu’il est déjà dépassé. Cette boutade met en lumière l’évolution rapide et continuelle de tout ce qui touche l’informatique, tant dans ses aspects matériels que logiciels. Ce livre n’échappe pas à cette dure réalité : au moment où vous le lirez, de nouvelles versions des logiciels décrits ici seront peut-être disponibles et les divers modes d’emplois devront peut-être subir des corrections plus ou moins importantes. Déjà, lors de son écriture, nous avons eu recours à des versions bêta[2] de logiciels pour essayer d’être le plus à jour possible. Mais il serait utopique de penser que nous avons réussi totalement et de façon durable. C’est une des raisons de l’existence d’un site Web associé à notre ouvrage, dont l’adresse est

http://www.math.uqam.ca/LivreLogicielsOutils/

et où vous pourrez trouver d’éventuelles corrections et divers compléments. Ce site mettra aussi à votre disposition les versions électroniques de tous les exemples décrits dans le livre, ainsi que des solutions de certains exercices et de certains projets. Vous trouverez aussi les adresses électroniques des auteurs, pour le cas où vous voudriez leur faire parvenir vos commentaires ou vos suggestions.

La rédaction d’un tel ouvrage n’aurait pu se faire en vase clos. Nous avons bénéficié de fructueux échanges avec nos collègues du Département de mathématiques de l’Université du Québec à Montréal, en particulier Jean-Baptiste Lapalme, Alain Taurisson, Carolyn Kieran, Benoît Côté,

---

[1] Ici encore on peut prévoir que *Linux* en viendra à occuper une place moins marginale que présentement. Quand cela sera possible, nous évoquerons des adaptations possibles de nos démarches pour des logiciels fonctionnant sous ce système d’exploitation.

[2] Voir l’article « Version d’un logiciel » sur le site http://fr.wikipedia.org/wiki/Accueil.

Louis Charbonneau et Fernando Hitt. Au fil des ans, nous avons été enrichis par divers textes et conférences de nombreux chercheurs et enseignants désireux d'apporter leur contribution sur la problématique de l'utilisation des TIC dans l'enseignement des mathématiques.

En terminant, nous voulons souligner l'apport de Madame Mélanie Tremblay et de Monsieur Christian Boissinotte, qui ont lu diverses versions préliminaires de portions de ce livre, et dont les commentaires judicieux ont contribué à l'améliorer. Qu'ils en soient chaleureusement remerciés. Nous ne voulons pas passer sous silence la contribution des étudiants qui ont suivi des cours donnés avec des versions très préliminaires de ce livre : leurs suggestions nous ont été précieuses. Nous espérons que les lecteurs de la présente version nous feront part de leurs réactions, et des diverses façons de continuer à l'améliorer.

Notes importantes

Page Web du livre

Le lecteur est invité à consulter la page Web suivante :

http://www.math.uqam.ca/LivreLogicielsOutils/

Il y trouvera des liens vers les sites Web mentionnés dans ce livre, les fichiers correspondant à tous les exemples, et les solutions de plusieurs exercices. Au besoin, corrections et compléments seront ajoutés.

Clic-droit sur Macintosh

Quand nous demanderons de faire un « clic-droit », c'est-à-dire un clic avec le bouton droit de la souris, les utilisateurs de Macintosh avec une souris à un seul bouton pourront obtenir un résultat équivalent : il suffira de faire un clic en maintenant enfoncée la touche « contrôle » du clavier. Notez en passant que les utilisateurs d'ordinateurs Macintosh peuvent utiliser sans problèmes des souris à deux boutons et roulette de défilement.

Extensions de fichiers

Les utilisateurs de *Windows* ajoutent (implicitement ou explicitement) des extensions aux noms de fichiers (voir quelques exemples dans le tableau ci-dessous). Par exemple, un texte nommé « Essai » sera automatiquement enregistré sous le nom « Essai.doc » par *Word 2003* pour *Windows*. Bien que ceci ne soit pas strictement nécessaire pour les utilisateurs *Macintosh*, nous suggérons à ceux-ci de toujours inclure explicitement ces extensions à la fin du nom des fichiers qu'ils créent : la version *Windows* du logiciel correspondant saura alors les reconnaître et les utiliser.

Extension	Logiciel	Description
.doc	*Word*	Fichier de texte lisible par *Microsoft Word*.
.xls	*Excel*	Classeur lisible par *Microsoft Excel*.
.docx	*Word 2007*	Fichier de texte lisible par *Microsoft Word 2007* seulement. Nous déconseillons pour l'instant l'utilisation de ce format incompatible avec les versions antérieures.
.xlsx	*Excel 2007*	Classeur lisible par *Microsoft Excel 2007* seulement. Nous déconseillons pour l'instant l'utilisation de ce format incompatible avec les versions antérieures.
.8xP .8xk	*TI-Connect*	Fichier contenant un programme (.8xP) ou une application (.8xk) pour une calculatrice *TI-83Plus* ou *TI-84Plus*.
.fig	*Cabri*	Fichier des figures produites par *Cabri-géomètre*.
.mac	*Cabri*	Fichier des macros produites par *Cabri-géomètre*.
.htm .html	(*Plusieurs logiciels*)	Fichier contenant la description d'une page Web, lisible par tout navigateur Web (*Internet Explorer*, *Safari*, *Firefox*), et éditable avec *Kompozer*.
.gif .png .jpeg	(*Plusieurs logiciels*)	Fichier contenant un graphique de type « Graphics Interchange Format », « Portable Network Graphics » ou « Joint Photographic Experts Group ». Ces types de graphiques sont utilisables sur le Web.
.pdf	(*Plusieurs logiciels*)	Fichier contenant un document de type « Portable Document Format ». Ils peuvent être créés au moyen d'une « impression virtuelle », et sont utilisables sur le Web.

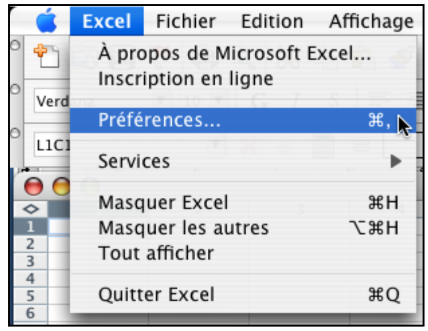
Conventions utilisées dans ce livre

Pour que ce livre reste d'une taille raisonnable, nous avons choisi de décrire les manipulations à effectuer de façon abrégée, mais cependant suffisante pour s'y retrouver.

Voici un exemple d'une description abrégée illustrant les conventions utilisées :

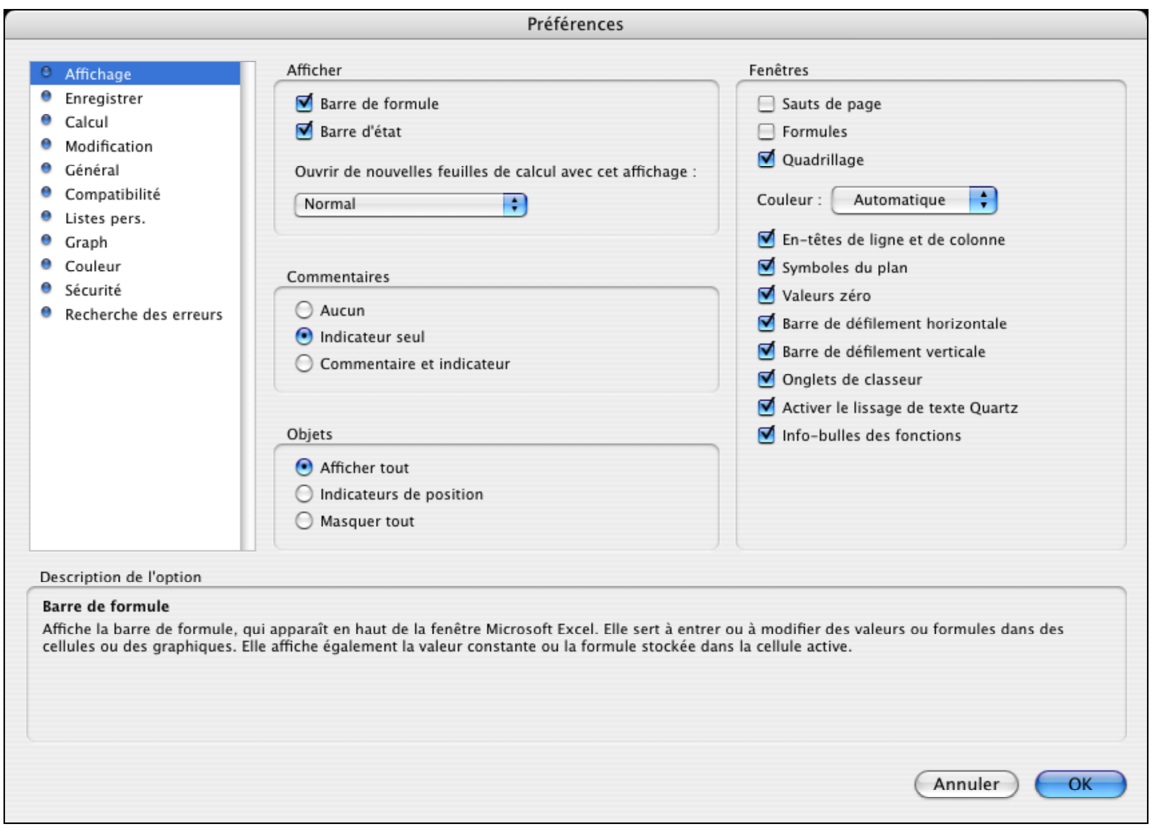
> à partir du menu « **Excel** », faire successivement les choix suivants :
> « Préférences... » ➤ « Général » ➤ cocher « Style de référence L1C1 ».

Explicitons maintenant en détails cette description abrégée, dans le cas d'*Excel 2004* sur *Macintosh*.

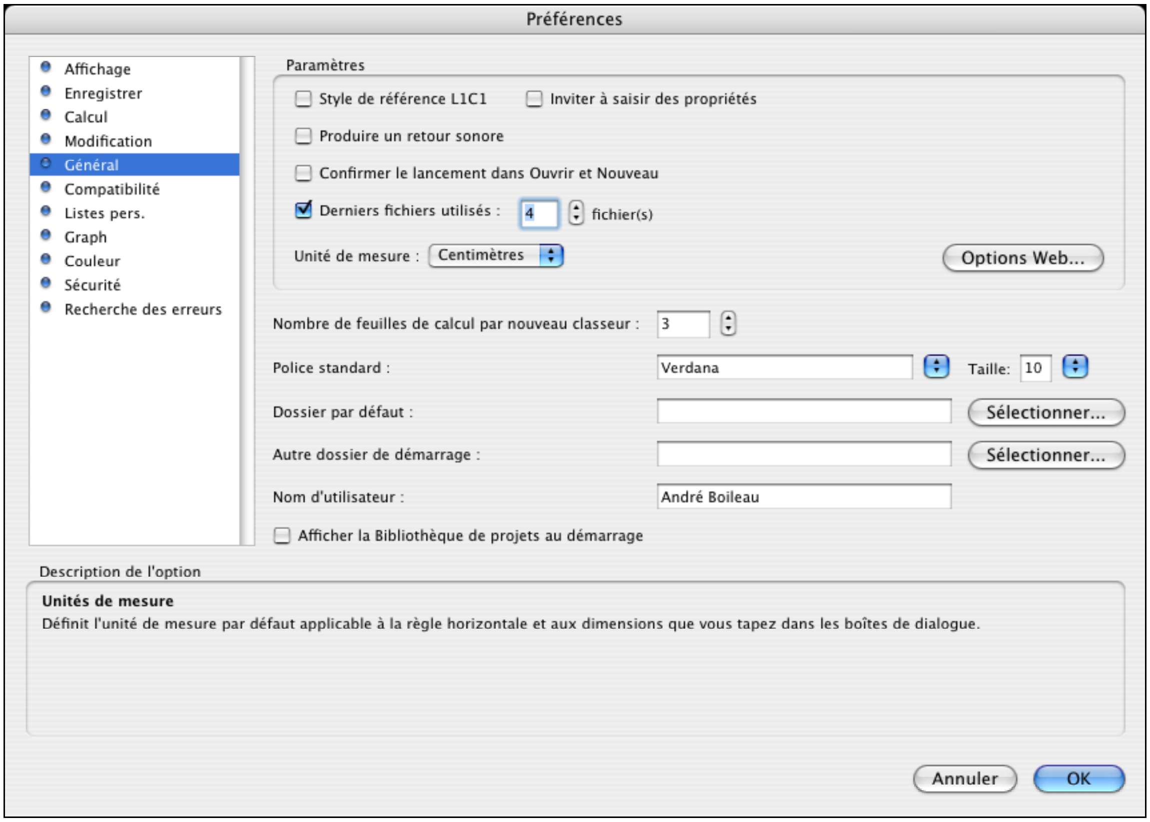


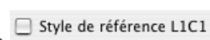
1. Dans un premier temps, il faut s'assurer que l'application *Excel* est bien lancée et disponible. C'est le cas lorsque le menu « Excel » apparaît immédiatement à droite du menu pomme . Dans le menu « Excel » de l'application, choisir l'item « Préférences... », tel qu'illustré ci-contre.

2. Une fenêtre de dialogue apparaît alors, tel qu'illustré ci-dessous. À la gauche de la fenêtre apparaît un rectangle comportant une liste d'items, dont un qui se nomme « Général » : amener le pointeur de la souris au-dessus de cet item et cliquez sur celui-ci.

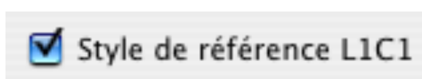


3. Le contenu de la fenêtre se transforme alors, comme illustré ci-dessous.

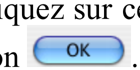


Dans le rectangle de droite de la fenêtre se trouve un item 

comportant une case à cocher  et un nom « Style de référence L1C1 ». Si l'item est déjà coché et dans l'état ci-dessous



vous pouvez cliquer sur le bouton 

. Sinon, amenez le pointeur de la souris au-dessus de cette case à cocher et cliquez sur celle-ci pour cocher l'item. Confirmer ensuite votre choix par un clic sur le bouton 

.

On imagine sans peine l'augmentation de la taille de ce livre si toutes les manipulations étaient décrites de façon exhaustive (les trois étapes ci-dessus) plutôt qu'abrégée (la formulation originale). D'autant plus que ces manipulations diffèrent parfois d'une version du logiciel à l'autre : la description ci-dessus est correcte pour la version *Macintosh* d'*Excel 2004*, mais il faudrait la modifier quelque peu pour la version *Windows* d'*Excel 2003*, ainsi que pour la version *Windows* d'*Excel 2007* (voir à ce sujet l'encadré de la page 55).

Remarque à propos de la couleur

Ce manuel est imprimé en noir et blanc. Quand nous faisons référence à certains éléments colorés, il faut comprendre que nous décrivons ce qu'on obtient à l'écran de l'ordinateur, et non ce qu'on peut voir dans les figures imprimées. Le lecteur pourra retrouver les couleurs en ouvrant, sur son propre ordinateur, le fichier exemple correspondant.

Remarque à propos *Microsoft Office 2008* pour *Macintosh*

Dans le cadre de notre ouvrage, nous déconseillons l'utilisation de *Microsoft Office 2008* pour *Macintosh*, car *Microsoft* a décidé de ne pas y inclure *VBA*, son langage de programmation, sur lequel sont basés plusieurs exemples de ce livre. Devant le tollé de protestations qui a suivi cette annonce, *Microsoft* a annoncé que le *VBA* serait de retour dans la prochaine version principale de la suite *Office* pour *Macintosh*. En attendant ce jour, nous vous suggérons d'utiliser *Microsoft Office 2004* (ou une version plus ancienne) sur *Macintosh*. Soulignons cependant que ceci ne s'applique qu'aux ordinateurs *Macintosh* : sous *Windows*, il n'y a pas de problème à utiliser *Office 2007* (ou toute autre version), car *VBA* y est inclus.

Chapitre 1

Word et les textes mathématiques

Dans ce chapitre, nous allons voir comment utiliser *Microsoft Word* pour créer des textes mathématiques[1]. Nous supposerons que le lecteur est déjà capable d'utiliser *Word* pour écrire des textes simples, comportant quelques éléments de mise en forme : choix de la police de caractères ainsi que sa taille et son style, choix de l'alignement des paragraphes, insertion d'un saut de page ou d'une note en bas de page, etc.

Nous porterons spécifiquement notre attention sur la création et l'incorporation d'équations et de graphiques dans nos textes. Nous nous intéresserons plus particulièrement aux graphiques représentant des objets mathématiques et pour lesquels nous rechercherons une fidélité et une précision maximale.

Pour faciliter notre étude, nous allons installer dans *Word* la barre d'outils *Progiciels* et la collection de macros *LangageGraphique* (qui servira au chapitre suivant). Nous pouvons trouver sur la page Web de ce livre ou à l'adresse suivante

http://www.math.uqam.ca/_boileau/LangageGraphique.html

des fichiers permettant d'installer ces deux éléments ainsi qu'un guide décrivant comment les installer et les utiliser.

Au terme du processus d'installation, nous obtenons la barre d'outils *Progiciels*[2] comportant les icônes suivantes (version *Macintosh* à gauche et *Windows* à droite) :

De gauche à droite, nous retrouvons des icônes-boutons permettant

- d'ouvrir l'éditeur d'équations de *Word*
- de faire apparaître ou de cacher la barre des outils de dessin de *Word*
- d'ouvrir l'éditeur de programmes *Visual Basic for Applications*
- de créer un dessin « *LangageGraphique* » dans un nouveau document *Word*
- d'insérer une zone de texte (transparente et sans bordures).

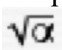
---

[1] Il existe aussi des logiciels plus puissants, mais plus complexes, pour produire des textes mathématiques : TeX, LaTeX et AMS-LaTeX par exemple. Nous sommes cependant d'avis que, pour des enseignants au niveau secondaire, *Word* comporte assez de puissance, tout en demeurant beaucoup plus simple d'utilisation.

[2] Elle demeure toujours visible et accessible, sauf pour *Word 2007*, où l'on doit cliquer sur l'onglet « Compléments » pour la révéler.

Dans ce premier chapitre, nous allons utiliser les deux premières icônes, alors que l'usage des trois dernières sera étudié au chapitre suivant.

1.1 Utilisation de l'éditeur d'équations

Supposons que l'on veuille insérer une formule mathématique dans un document *Word*. Il faut d'abord cliquer dans le texte, à l'endroit où l'on veut effectuer cette insertion, puis cliquer sur l'icône  permettant d'ouvrir l'éditeur d'équations[1]. Une barre d'outils (*Windows*) ou une fenêtre comportant cette barre d'outils (*Macintosh*) apparaît alors[2].

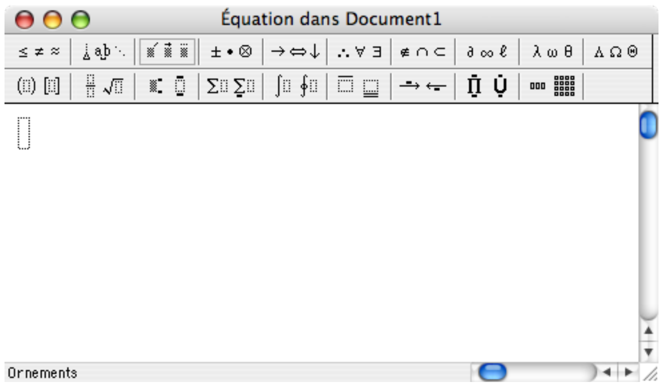
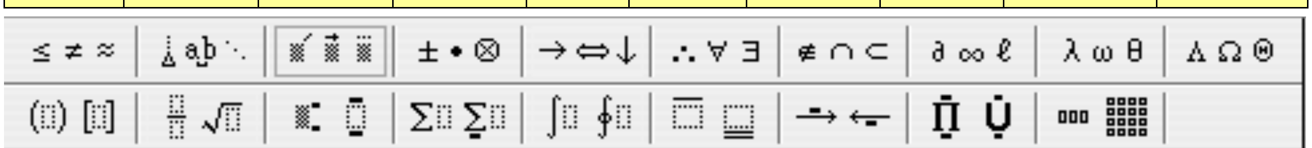


Figure 1.1.1 Fenêtre permettant la définition d'une formule sur plate-forme Macintosh.

Jetons un bref coup d'œil sur la barre d'outils de l'éditeur d'équations. Elle est composée de deux lignes, elles-mêmes constituées de boutons sur lesquels on peut cliquer pour faire apparaître des palettes. Ces boutons (et les palettes du même nom qui leur correspondent) sont identifiés à la figure 1.1.2.

Relations	Espaces et points de suspension	Ornements	Opérateurs	Flèches	Logiques	Ensembles	Symboles divers	Caractères grecs (minuscules)	Caractères grecs (majuscules)



Délimiteurs	Fractions Radicaux	Indices Exposants	Sommations	Intégrales	Barres	Flèches étiquetées	Produits et ensembles	Matrices

Figure 1.1.2 Identification des boutons de la palette de l'éditeur d'équations.

[1] Bien qu'il serait plus juste de parler d'un **éditeur de formules**, nous avons choisi de conserver l'appellation utilisée dans la documentation de *Word*. Signalons en passant qu'il existe un tout nouvel éditeur d'équations dans la version 2007 de *Word* pour *Windows* : nous ne l'utiliserons pas, par souci de compatibilité avec les versions précédentes.

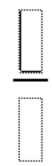
[2] Il est aussi possible de travailler avec une fenêtre dans la version *Windows*. Pour plus de détails, voir http://www.hotline-pc.org/word.htm ou http://support.microsoft.com/kb/318797/fr.

Notons que les palettes de la ligne du haut permettent d'insérer des caractères spéciaux, parfois à des positions spécifiques (par exemple : le symbole ± ou une flèche $\vec{v}$ au-dessus d'un autre symbole), tandis que les palettes de la ligne du bas permettent d'insérer des modèles à compléter (par exemple : une fraction, une racine, un indice ou un exposant).

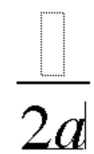
Passons maintenant à un premier exemple : supposons qu'on veuille écrire la formule

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

avec cet éditeur. Par un clic sur le bouton « Fractions-Radicaux », on fait apparaître la palette correspondante puis on choisit le modèle « Fraction ». On obtient ainsi le modèle ci-contre constitué d'une barre de fractions et de deux zones rectangulaires (délimitées par des pointillés tant qu'elles sont vides) correspondant au numérateur et au dénominateur. Remarquons aussi que le curseur clignotant est composé de deux barres : une barre horizontale, qui correspond à la base de la zone rectangulaire courante, et une barre verticale dont la hauteur correspond aussi à la zone rectangulaire courante et dont la position correspond au point où se fera l'insertion.



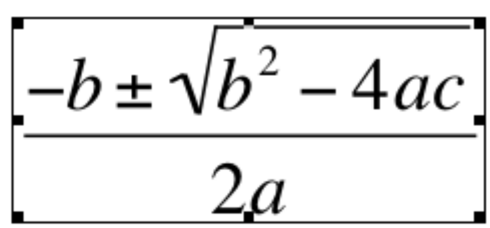
En général, on peut passer par toutes les zones rectangulaires d'une formule via une utilisation répété de la touche de tabulation TAB (ou Majuscule+TAB pour un déplacement en ordre inverse). On peut aussi aller à un point précis de la formule en utilisant les touches fléchées ou les clics souris. D'une façon ou d'une autre, allons maintenant dans la zone du dénominateur. On peut alors taper « 2a », et on peut constater que l'éditeur d'équations suit automatiquement[1] les conventions typographiques usuelles en mathématiques, selon lesquelles les variables sont écrites en italique. Il est intéressant de constater, par ailleurs, que si on tapait « sin(x) », l'éditeur rétablirait le style normal dès qu'il reconnaîtrait la fonction « sin ».



Passons maintenant à la zone du numérateur. On tape « -b », on insère le symbole « ± » via le bouton « Opérateurs », puis on choisit le modèle « Racine carrée » dans la palette « Fractions-Radicaux ». Notez que notre curseur se trouve maintenant dans la zone correspondant au radical. On tape alors « b », puis on choisit le modèle « Exposant » de la palette « Indices-Exposants » de façon à pouvoir taper l'exposant « 2 ». En utilisant la touche « TAB » (à moins que ce ne soit une touche fléchée ou un clic souris), on passe de la zone « Exposant » à la zone « Radical » pour finir de taper « -4ac ». Voici toutes les étapes parcourues pour arriver au résultat final :

$$\frac{-b}{2a} \qquad \frac{-b\pm}{2a} \qquad \frac{-b\pm\sqrt{}}{2a} \qquad \frac{-b\pm\sqrt{b}}{2a} \qquad \frac{-b\pm\sqrt{b^2}}{2a} \qquad \frac{-b\pm\sqrt{b^2-4ac}}{2a}$$

Reste maintenant à insérer cette formule dans notre document *Word* : il suffit de fermer la fenêtre de l'éditeur d'équations (ou de cliquer dans le texte, en dehors de l'équation, si on travaille directement dans le document *Word*, sans la fenêtre de l'éditeur d'équations). Si, plus



[1] Même l'espacement est géré automatiquement : la touche d'espacement est parfois inopérante, ce qui n'est pas toujours souhaitable. D'où la présence de la palette « Espaces et points de suspension ».

tard, nous voulons modifier cette formule, nous pouvons toujours l'ouvrir dans l'éditeur d'équations par un double-clic ou par un clic-droit suivi du choix de l'item « Objet Équation ➤ Ouvrir » dans le menu local. Notons aussi qu'on peut redimensionner une formule[1] en la sélectionnant (par un clic) et en modifiant ses dimensions à l'aide d'une des « poignées » qui apparaissent alors.

À partir de ces connaissances de base, nous invitons le lecteur à explorer de façon autonome les différentes possibilités offertes par les palettes de symboles et de modèles qui sont disponibles. Nous l'invitons à utiliser les **modèles pour les parenthèses, crochets ou accolades** car ces modèles, contrairement aux caractères correspondants obtenus via le clavier, verront leur taille s'adapter automatiquement au contexte de leur utilisation. Nous lui suggérons aussi de ne pas trop se laisser influencer par les noms donnés aux divers modèles, noms qui apparaissent quand on survole l'icône qui leur correspond, Par exemple, on peut utiliser le modèle de « Matrice à taille variable » (qui permet de définir sur-le-champ l'alignement des lignes et colonnes - voir figure 1.1.3) pour définir une expression telle que $f(x)=\begin{cases} x\sin\left(\frac{1}{x}\right) & si\ x\neq 0 \\ 0 & sinon \end{cases}$.

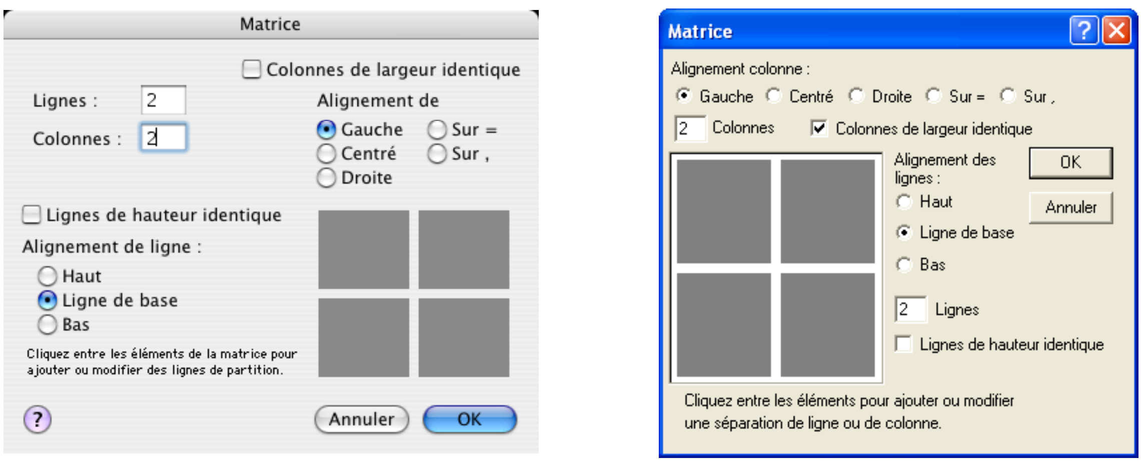


Figure 1.1.3 Fenêtre de dialogue accompagnant le choix du modèle « Matrice de taille variable » (Macintosh à gauche et Windows à droite).

Il peut arriver que la disposition choisie automatiquement par l'éditeur d'équations produise un résultat qui ne nous convienne pas totalement. Par exemple, dans la formule ci-contre, nous aimerions peut-être que le chiffre 3 soit de taille plus petite et disposé de façon à ne pas toucher le symbole du radical. Dans l'éditeur d'équations, on peut déplacer manuellement des blocs comme suit : on sélectionne tout d'abord le bloc visé (le chiffre 3 ici) puis on utilise de façon répétée des combinaisons de touches commande+flèche (*Macintosh*) ou contrôle+flèche (*Windows*). De même on peut modifier manuellement la taille de la police utilisée dans un bloc à l'aide de l'item « Autre... » du menu « Taille ».

$$\sqrt[3]{x}$$

$$\sqrt[3]{x}$$

[1] On verra plus loin d'autres façons de redimensionner formules et graphiques.

Le lecteur pourra explorer par lui-même les divers menus mis à sa disposition par l'éditeur d'équations. Nous nous contenterons de souligner ici

- le menu « Affichage » qui permet notamment de contrôler la taille de l'affichage et de redessiner la formule si jamais elle venait à être corrompue, souvent suite à des modifications (voir l'exemple ci-contre) $\sqrt[3]{x}$
- le menu « Format » qui permet notamment de contrôler la disposition d'un texte mathématique s'étendant sur plus d'une ligne
- le menu « Style » qui permet notamment de basculer entre le style automatique « Math » et le style « Texte » où l'utilisateur peut, entre autres choses, utiliser la barre d'espacement
- le menu « Aide » qui renseignera le lecteur, entre autres choses, sur les raccourcis-clavier disponibles : commande+F ou contrôle+F pour insérer le modèle « Fraction verticale », commande+R ou contrôle+R pour insérer le modèle « Racine carrée », commande+E ou contrôle+E pour insérer le modèle « Exposant », etc.

Disons finalement que *Word* dispose d'un outil plus ancien pour écrire des formules mathématiques : les champs « Eq ». Il s'agit là d'une manière vieillotte et moins pratique de procéder, mais vous en rencontrerez peut-être des exemples si vous héritez de textes mathématiques anciens composés dans *Word*. Illustrons le fonctionnement de cet outil en revenant à notre premier exemple : pour obtenir la formule

$$\frac{-b\pm\sqrt{b^2 -4ac}}{2a}$$

il faut utiliser le menu « Insertion » ➤ « Champ... » ➤ « Équations et formules » ➤ « Eq (Eq) » puis taper[1] « EQ \F(-b±\R(;b\S(2) -4ac) ;2a) » dans la zone de texte. Plutôt que d'avoir à apprendre comment composer et modifier de telles formules, il suffit de savoir qu'on peut les transformer et les ouvrir dans l'éditeur d'équations via un double-clic. Mais ça ne fonctionne pas toujours : par exemple, il m'est arrivé (dans certaines circonstances qui restent nébuleuses) que l'éditeur d'équation s'ouvre avec une zone vide après un double-clic sur la formule ci-dessus donnant les solutions d'une équation du second degré[2]. Il faut aussi savoir que le résultat obtenu n'est pas identique à l'original : un double-clic sur la formule $\sqrt[3]{\mathrm{x}}$ la transformera en $\sqrt[3]{x}$.

Terminons en disant que certaines versions françaises plus anciennes de *Word* avaient le problème suivant : quand on ramenait un fichier contenant des champs « Eq », certains points-virgules séparant les paramètres étaient transformés en virgules, provoquant des erreurs signalées par la mention « **Erreur !** ». La solution consistait alors à faire un clic-droit et « Basculer les codes de champs », puis de reconvertir manuellement ces virgules en points-virgules. Il semble que *Microsoft* ait corrigé ce problème dans les versions plus récentes de *Word*.

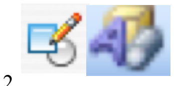
1.2 Les outils graphiques dans Word

Il est surprenant de constater les nombreuses possibilités mises à votre disposition par *Word* pour produire des dessins. Dans ce qui suit, nous allons décrire comment utiliser les outils de **dessin** de *Word* (disponibles dans toute la suite *Office*) pour créer certains graphiques utiles en

[1] Dans la version *Windows*, il faut d'abord cliquer sur le bouton « Codes de champs ».
[2] Dans ce cas, il semble que ce soit la présence du symbole « ± » qui causait problème.

mathématiques. Nous ne visons pas une description exhaustive de tous les outils disponibles : nous voulons seulement illustrer certaines des possibilités.

Pour accéder à la barre des outils de dessin de *Word*, on peut cliquer sur l'icône **Dessin** (dont les versions *Macintosh* et *Windows* apparaîssent ci-contre) de la barre d'outils *Progiciels*[1]. Une fois affichée, cette barre d'outils ressemble à la figure 1.2.4[2].



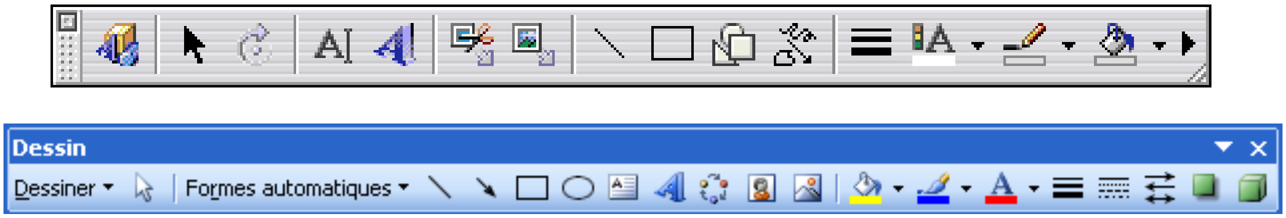


Figure 1.2.4 La barre d'outils Dessin, version Macintosh (en haut) et Windows (en bas)

À première vue, ça peut sembler assez inoffensif. Mais cette barre d'outils recèle des possibilités cachées, comme nous le montrent les figures 1.2.5 et 1.2.6.

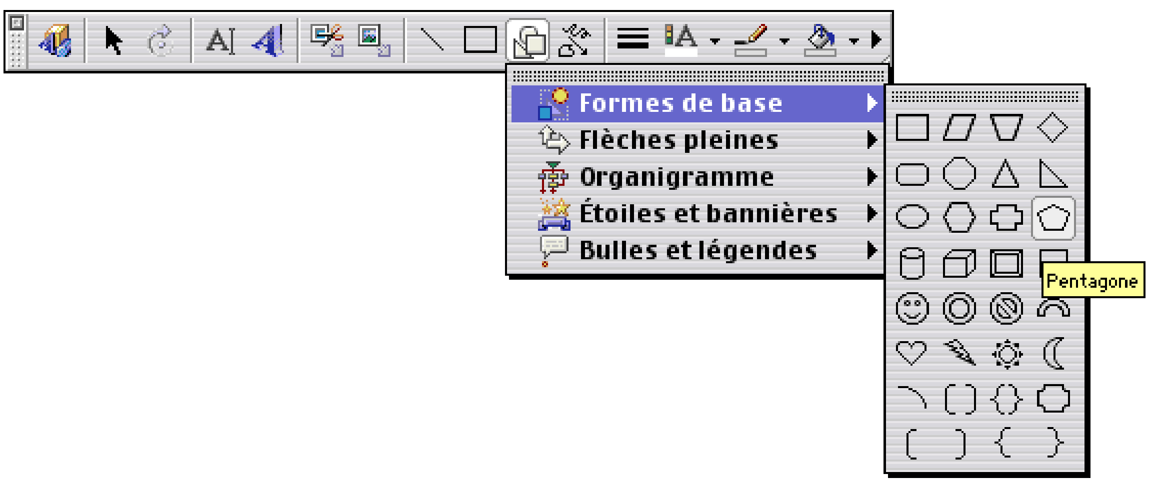


Figure 1.2.5 Quelques possibilités de la barre d'outils de dessin de *Word* (version Macintosh).

[1] Dans *Word 2007* pour *Windows*, il n'existe plus de barre d'outils de dessin proprement dite. L'icône **Dessin** de la barre « Progiciels » n'y est donc pas fonctionnelle. Dans l'appendice, à la fin de ce chapitre, nous donnons une façon de retrouver, dans cette version de *Word*, tous les outils de dessin mentionnés dans le texte.

[2] La disposition peut varier selon la version de *Word* utilisée, mais aussi selon la forme que vous donnerez à cette palette (grâce à la « poignée » de redimensionnement en bas à droite).

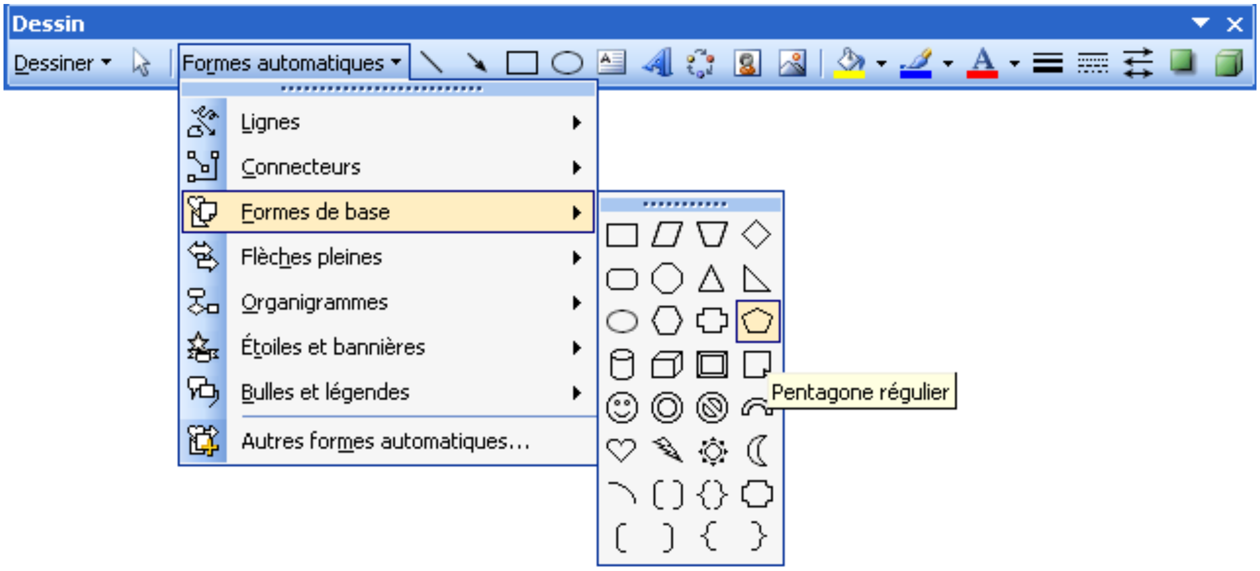


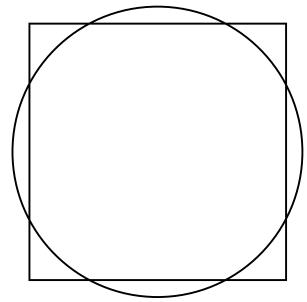
Figure 1.2.6 Quelques possibilités de la barre d'outils de dessin de Word (version Windows).

Plaçons-nous dans l'optique de quelqu'un qui veut produire des graphiques précis. Si nous voulons obtenir une forme (un rectangle, par exemple) qui apparaît dans les items de la barre d'outils **Dessin**, nous n'avons qu'à sélectionner (par un clic) l'icône correspondante et qu'à décrire ensuite **gestuellement** (par un « glisser » de souris) la position et les dimensions de cette forme. Notons en passant que nous pourrons toujours modifier ultérieurement (toujours gestuellement, par un « glisser » de souris) la position et les dimensions de notre forme.

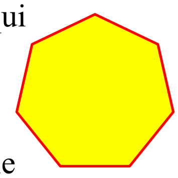
Même si la forme désirée ne correspond à aucun des items de la barre d'outils **Dessins**, tout n'est pas perdu. Si, par exemple, nous voulions tracer un carré, il suffirait d'indiquer à Word que nous désirons tracer un rectangle avec une contrainte : ceci se fait en maintenant enfoncée la touche **majuscule** pendant qu'on fait glisser la souris. De même, si nous voulions un triangle équilatéral, il suffirait de tracer un triangle isocèle avec contrainte (toujours indiquée par **majuscule**). Notons, toujours en passant, que cette même touche **majuscule** nous permet de redimensionner une forme avec la contrainte que les facteurs horizontal et vertical de changement d'échelle demeurent identiques (ce qui nous permet d'obtenir une **figure semblable** à la figure initiale).

Il y a une autre façon d'imposer une contrainte à une forme. Supposons que nous voulions obtenir un rectangle dont la base mesure deux fois la hauteur. On commence d'abord par tracer un rectangle quelconque. Puis, tandis qu'il est encore sélectionné, on fait apparaître le menu local correspondant (par un clic-droit) et on choisit l'item « Format de la forme..., » (*Macintosh*) ou « Mettre en forme l'objet » (*Windows*) suivi de l'onglet « Taille » : on peut alors indiquer **textuellement** (via le clavier) la hauteur et la largeur (par exemple 1 cm et 2 cm). Remarquons qu'on pourrait aussi spécifier ici l'angle de rotation à appliquer à cette forme.

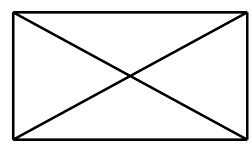
Considérons un autre exemple : supposons que nous voulions tracer simultanément un cercle et un carré de même aire. Si c est la mesure du côté du carré et r la mesure du rayon du cercle, on devra avoir $c^2 = \pi r^2$, et donc $c = \sqrt{\pi}\, r$. On peut donc tracer tout d'abord un cercle (une ellipse avec contrainte, via touche **majuscule**) et un rectangle quelconques. Puis on fixe (comme au paragraphe précédent) le rayon du cercle à 1cm ([1]) et le côté du carré à $\sqrt{\pi} \approx 1{,}772453851$ cm. Si on veut obtenir une figure comme ci-dessous où nos figures sont centrées et superposées, il faudra en outre



- spécifier qu'on ne veut aucun remplissage pour nos formes (via l'outil « pot de peinture » de la palette **Dessin**)
- donner le même centre à nos deux formes (via les commandes « Aligner le centre »[2] et « Aligner le milieu » appliquées quand les deux formes sont sélectionnées[3])
- regrouper nos deux formes en un seul objet (via la commande « Grouper », appliquée quand les deux formes sont sélectionnées)
- redimensionner le tout pour obtenir une figure semblable de la dimension voulue (ne pas oublier la touche **majuscule**).



Cependant, malgré toute cette versatilité, on se heurte rapidement à des figures qui sont, à toutes fins pratiques, impossibles à réaliser **exactement** dans *Word*. Il peut arriver qu'une forme ne puisse tout simplement pas être obtenue à partir des formes disponibles. Par exemple, si on veut faire le dessin d'un polygone régulier à 7 côtés (pour donner un exemple de polygone non constructible avec la règle et le compas).



Il est aussi très difficile en général de réaliser des figures qui impliquent un agencement précis de plusieurs formes simples. Par exemple, supposons que nous voulions tracer un rectangle avec ses deux diagonales. On peut essayer de tracer tout d'abord un rectangle, puis deux segments en essayant de choisir les sommets du rectangle comme points de départ et d'arrivée, pour terminer en regroupant les trois formes. Mais le choix des extrémités de ces « diagonales » n'est qu'approximatif[4], et on n'obtient donc pas la figure « exacte » recherchée.

On peut cependant déterminer **textuellement** non seulement les dimensions mais aussi les positions exactes de tous nos éléments graphiques.

---

[1] Pour obtenir un rayon de 1 cm, il faudra spécifier une hauteur et une largeur de 2 cm.

[2] Ces commandes sont disponibles à partir du 1er item de la barre d'outils **Dessin**. Dans la version *Windows*, les noms utilisés sont « Centrer » et « Aligner au milieu ».

[3] Rappelons qu'on peut sélectionner plusieurs formes (ou, plus généralement, plusieurs objets) en cliquant sur chacune tout en maintenant la touche **majuscule** enfoncée.

[4] Si nous sommes libres de choisir nous-mêmes les sommets du rectangle, on peut demander d'aligner les objets sur une grille, ce qui nous permettra de choisir facilement les extrémités des diagonales. Mais si notre rectangle est lui-même fruit d'une autre construction et a ses sommets en des endroits qui ne correspondent pas exactement à des pixels de l'écran, on ne pourra pas (avec la souris) spécifier que les sommets du rectangle et les extrémités des diagonales coïncident.

Pour déterminer exactement la position d'un élément graphique

Word 2004 pour Macintosh : après avoir sélectionné le graphique, on fait apparaître le menu local et on choisit l'item « Format de la forme... », suivi de l'onglet « Disposition » : après un clic sur le bouton « Avancée... » on choisit l'onglet « Positionnement de l'image », on peut alors spécifier **textuellement** (via le clavier) la position absolue (horizontale et verticale) du graphique sélectionné.

Word 2003 et 2007 pour Windows : après avoir sélectionné le graphique, on fait apparaître le menu local et on choisit l'item « Mettre en forme l'objet... », suivi de l'onglet « Habillage » : après un clic sur le bouton « Avancé... » on choisit l'onglet « Positionnement de l'image », on peut alors spécifier **textuellement** (via le clavier) la position absolue (horizontale et verticale) du graphique sélectionné.

De cette façon, on peut obtenir une description exacte[1] de chacune des formes constituant la figure désirée, mais il faut avouer que la lourdeur du processus nous fait payer un prix très élevé pour obtenir la précision que nous recherchons. Soulignons encore une fois que cette précision a été rendue possible par une description **textuelle** de toutes les dimensions et positions en présence. Nous verrons au chapitre suivant une autre approche possible face à cette recherche de précision, tout en ne quittant pas *Word*.

Dans certains cas cependant, si vous avez la chance de disposer de logiciels mathématiques (dont vous trouverez des références sur la page Web du livre) vous permettant de faire des graphiques

- logiciels de géométrie dynamique (tel *Cabri* ou *Cinderella*)
- traceurs de courbes (tel *Graphing Calculator*)
- tableurs (tel *Excel* ou *Calc*)
- logiciels de calcul symbolique (tel *Maple*, *Derive* ou *Mathematica*)
- environnements de programmation (tel *Logo*, *POV-Ray* ou *Expresso*)

il arrivera souvent que vous pourrez obtenir **plus simplement** des graphiques précis en choisissant l'outil le plus approprié.

1.3 Graphiques vectoriels et matriciels

On peut distinguer deux types de graphiques informatiques : les graphiques **vectoriels** (appelés **dessins** dans *Word*), et les graphiques **matriciels** (appelés **images** dans *Word* et « bitmaps » en anglais). Après une brève description, nous verrons quel type convient le mieux à la précision et à l'exactitude recherchée en mathématiques.

Commençons par les graphiques matriciels, qui sont les plus répandus : les logiciels de copie d'écran, les numériseurs de documents, les appareils photos et les caméras numériques produisent tous des graphiques matriciels. On peut se représenter un graphique de ce type comme un tableau

[1] Cette exactitude est cependant limitée par la précision des nombres qui sont entrés textuellement, précision qui ne peut toujours être absolue (exemple : les nombres irrationnels). Mais cette précision est, en général, bien suffisante pour que la figure obtenue soit visuellement exacte.

rectangulaire de carrés colorés, appelés pixels[1]. La figure 1.3.7 montre une image matricielle agrandie dix fois : on distingue clairement les pixels qui la composent.

Figure 1.3.7 L'icône de la barre d'outils « Dessin », agrandie 10 fois.

Il est moins facile de décrire les graphiques vectoriels, parce que leur définition interne ne nous est pas directement accessible : il s'agit en fait d'une collection d'objets géométriques (segments, polygones, cercles, courbes, etc.) spécifiés par leurs propriétés mathématiques (coordonnées, équations, etc.) et physiques (épaisseur, couleurs de contour et de remplissage, degré de transparence, etc.). Soulignons cependant que, pour visualiser un graphique vectoriel sur l'écran d'un ordinateur ou sur une feuille imprimée, on doit en faire une représentation matricielle. En effet, que ce soit dans le cas de l'écran d'un ordinateur (tube à rayons cathodiques ou écran à cristaux liquides) ou de la feuille d'une imprimante (laser ou à jet d'encre), les images sont obtenues en colorant des pixels disposés en grilles rectangulaires[2].

Essayons de rendre ceci plus concret. Les outils de dessin de *Word* produisent des graphiques vectoriels. Si nous les utilisons pour tracer un segment (avec l'outil « Trait »), nous pourrons par la suite changer la position de ses extrémités (via des « glisser » de la souris) : la description mathématique en mémoire sera modifiée tout en mettant à jour la représentation du segment sous forme de pixels. Par contraste, si nous travaillions sur une représentation uniquement matricielle de ce segment, nous pourrions modifier la position de groupes de pixels désignés par un outil de sélection (rectangle, lasso, etc.), mais l'alignement pourrait être perdu dans le processus (puisque la description de l'image par un tableau des pixels n'inclut aucune contrainte d'alignement).

---

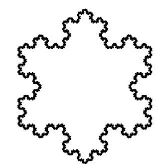
[1] Pour être plus complet, disons que, dans certains cas, les pixels ne sont pas carrés ou ne sont pas disposés selon une grille rectangulaire.

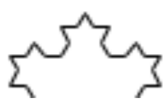
[2] Pour être plus complet, mentionnons deux exceptions : les tubes à rayons cathodiques pilotables vectoriellement et les tables traçantes ne fonctionnent pas de cette manière, et sont donc susceptibles de représenter plus fidèlement les graphiques vectoriels. Mais ces technologies deviennent de plus en plus rares, sans doute en raison de leur coût prohibitif.

De même, si on applique une transformation[1] (changement d'échelle, rotation, symétrie axiale, inversion, etc.) à un graphique, l'effet sera très différent selon que celui-ci est vectoriel ou matriciel. Prenons, par exemple, le segment suivant : —— . S'il s'agit d'un dessin vectoriel et que nous demandons de multiplier ses dimensions par un facteur 10, nous obtiendrons le segment ci-dessous :

Nous remarquons que son épaisseur est restée inchangée, et que seule sa longueur a été multipliée par 10. En fait, *Word* a pu calculer de nouvelles coordonnées répondant à notre demande : les représentations interne (coordonnées) et externe (pixels) ont été modifiées. Si, par contre, notre segment initial avait été un graphique matriciel, chaque pixel coloré aurait été remplacé par un bloc de 10×10 pixels de même couleur, pour donner le résultat ci-dessous :

Illustrons ceci à l'aide du graphique ci-contre (une des étapes menant au **Flocon de Koch**), sur lequel nous allons effectuer un changement d'échelle de facteur 10. Constatons l'effet produit sur un détail agrandi, selon qu'il est vectoriel (à gauche ci-dessous) ou matriciel (à droite ci-dessous) : on voit clairement dans ce cas la supériorité de la représentation vectorielle.





On vient de voir pourquoi les graphiques vectoriels produisent des résultats qui restent mathématiquement plus exacts quand on les transforme. Mais que faire si les graphiques sont obtenus autrement qu'avec les outils de dessin de *Word*, à l'aide de logiciels qui ne génèrent que des graphiques matriciels ? Dans les chapitres qui suivent, on verra des réponses adaptées aux divers logiciels étudiés. Mais nous pouvons déjà donner la piste suivante : on peut insérer[2] dans le document une image plus grande que nécessaire et réduire ses dimensions avec la barre d'outils « Image » de *Word*. L'image originale sera conservée par *Word*, qui pourra y avoir recours pour obtenir plus de précision.

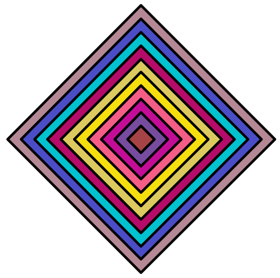
1.4 Placement des graphiques dans un texte

Que nos graphiques aient été réalisés dans *Word* ou dans un autre logiciel, qu'il s'agisse de graphiques vectoriels ou de graphiques matriciels, nous devons décider comment disposer ceux-ci dans notre texte. C'est précisément l'objet de la présente section. Et puisque *Word* considère les équations comme un type particulier de graphique, ce que nous dirons pour les graphiques s'appliquera aussi pour les équations.

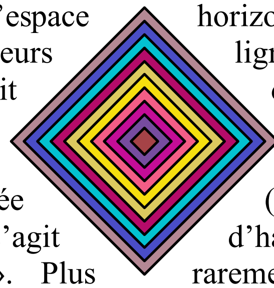
---

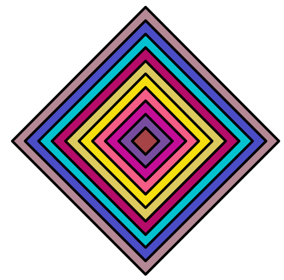
[1] Soulignons que ces transformations se font parfois de façon implicite. Par exemple, quand nous imprimons un document, la résolution de l'imprimante est souvent beaucoup plus grande que celle de l'écran : le pilote d'impression doit donc augmenter le nombre de pixels utilisés pour représenter un texte ou un graphique donné.

[2] Par copier-coller ou via le menu « Insertion » ➤ « Image » ➤ « À partir d'un fichier... ».



Avant de décrire comment procéder, voyons les divers résultats que nous pourrons obtenir. Le graphique pourra occuper tout l'espace horizontal, un peu comme si nous avions réservé plusieurs lignes à cet effet (voir ci-dessus) : pour *Word*, il s'agit d'un habillage « Haut et bas » du texte. On pourra aussi l'insérer dans un paragraphe, de sorte que le texte l'encadre de façon plus ou moins serrée (voir au centre et à droite) : pour *Word* il s'agit d'habillages de types « Rapproché » et « Encadré ». Plus rarement, on voudra placer le graphique « Derrière le texte » (en filigrane - voir ci-dessus à gauche) ou « Devant le texte » (pour le cacher).





Voyons maintenant comment arriver à de tels résultats. Il suffit de faire un clic-droit sur le graphique visé, et de choisir l'item « Format de l'image » dans le menu local qui apparaît alors. La fenêtre de la figure 1.4.8 apparaît alors, et on peut choisir le style d'habillage désiré par un clic sur l'icône correspondante. Remarquons que tous les styles disponibles ne sont pas proposés : pour les voir tous, il faut cliquer sur le bouton « Avancée... » et choisir l'onglet « Retour à la ligne » de la fenêtre de dialogue supplémentaire (voir figure 1.4.9). Veuillez noter que le style « Inséré dans le texte » n'est pas disponible pour les graphiques vectoriels (donc pour les **dessins** de *Word*), tandis que c'est le style adopté par défaut pour les équations et les graphiques matriciels : le style par défaut pour les graphiques vectoriels est plutôt « Haut et bas ». Notons au passage que nous disposons aussi de réglages pour contrôler de façon plus précise l'alignement et le positionnement des graphiques, ainsi que la distance du texte autour du graphique. Nous invitons le lecteur à expérimenter les diverses options, et à recourir à l'aide en ligne en cas de besoin, en faisant des recherches avec des mots clé comme « habillage » ou « filigrane ».

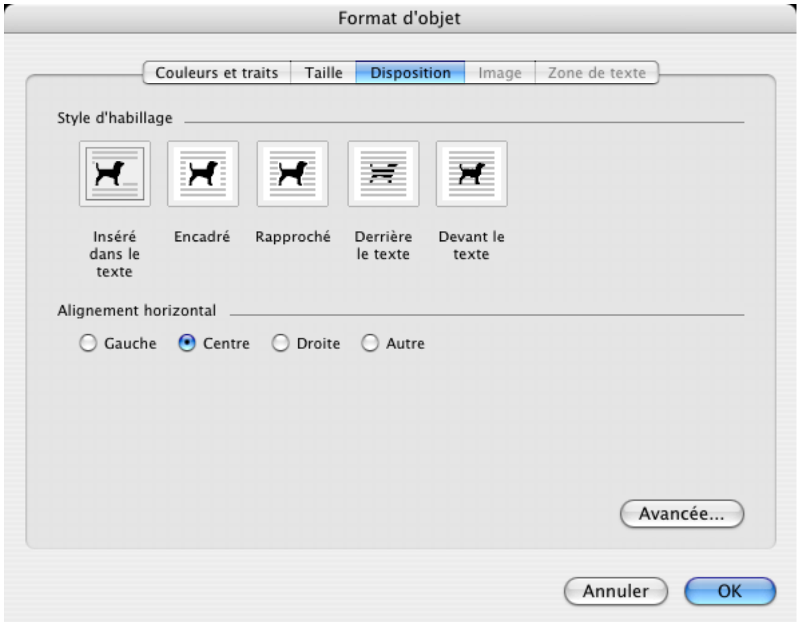


Figure 1.4.8 L'onglet « Disposition » de la fenêtre de dialogue amenée par le choix « Format de l'image » du menu local d'un graphique. (Noter que dans la version Windows « Disposition » est remplacé par « Habillage ».)

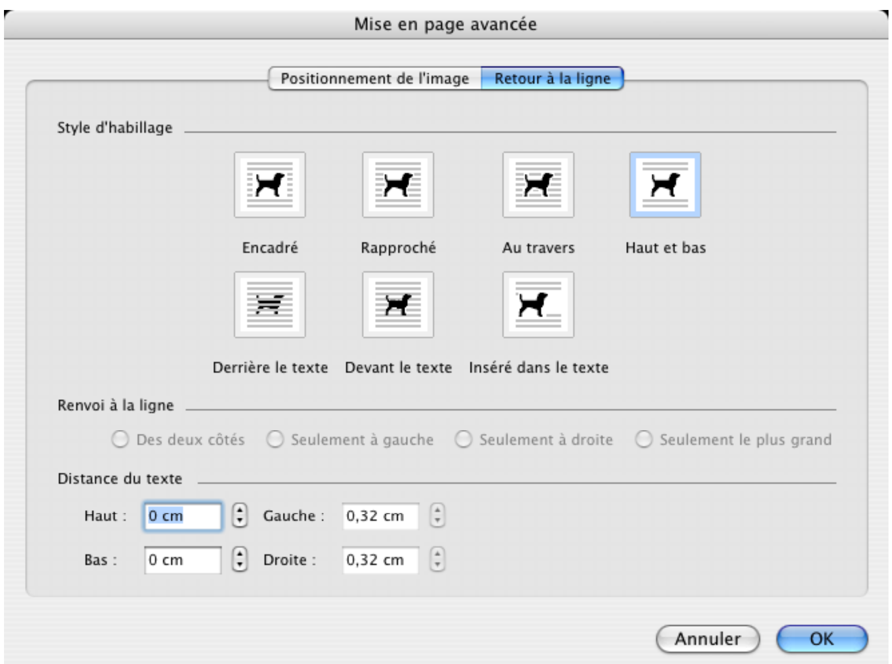


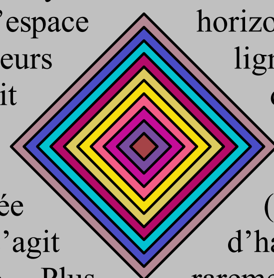
Figure 1.4.9 Les options supplémentaires disponibles via le bouton « Avancée... » de la figure 1.4.8. (Noter que dans la version Windows « Retour à la ligne » est remplacé par « Habillage du texte».)

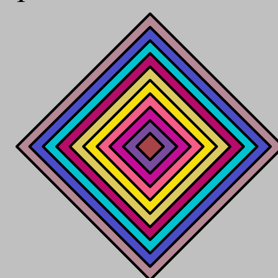
1.5 Utilisation du presse-papier

Il est bien connu que le presse-papier est l'endroit où vont se déposer les objets (textes, graphiques, etc.) qu'on a coupé ou copié : c'est donc aussi l'endroit où l'on ira chercher les objets qu'on va coller. Ce qui est moins connu par contre, c'est que les divers programmes vont placer les objets qu'on a coupé ou copié sous plusieurs formes, et que les logiciels à partir desquels on va coller (qui ne sont pas nécessairement les mêmes que les programmes où la coupure/copie a été faite) iront chercher la forme qui leur semblera la meilleure (parmi celles qu'ils reconnaîtront).

Word va nous permettre non seulement de voir les diverses formes de l'objet qui a été coupé ou copié, mais aussi de choisir manuellement laquelle de ces formes sera utilisée lors du collage. Illustrons ceci à partir du paragraphe suivant, que nous avons déjà rencontré à la section précédente. Supposons que nous avons sélectionné la totalité de ce paragraphe et nous en voulons en effectuer une copie.

Avant de décrire comment procéder, voyons les divers résultats que nous pourrons obtenir. Le graphique pourra occuper tout l'espace horizontal, un peu comme si nous avions réservé plusieurs lignes à cet effet (voir ci-dessus) : pour *Word*, il s'agit d'un habillage « Haut et bas » du texte. On pourra aussi l'insé-rer dans un paragraphe, de sorte que le texte l'en-cadre de façon plus ou moins serrée (voir au centre et à droite) : pour *Word* il s'agit d'habillages de ty-pes « Rapproché » et « Encadré ». Plus rarement, on voudra placer le graphique « Derrière le texte » (en filigrane - voir ci-dessus à gauche) ou « Devant le texte » (pour le cacher).





Au lieu de faire un simple « Coller », choisissons plutôt l'item « Collage spécial... » du menu « Édition ». La fenêtre de la figure 1.5.10 apparaît alors.

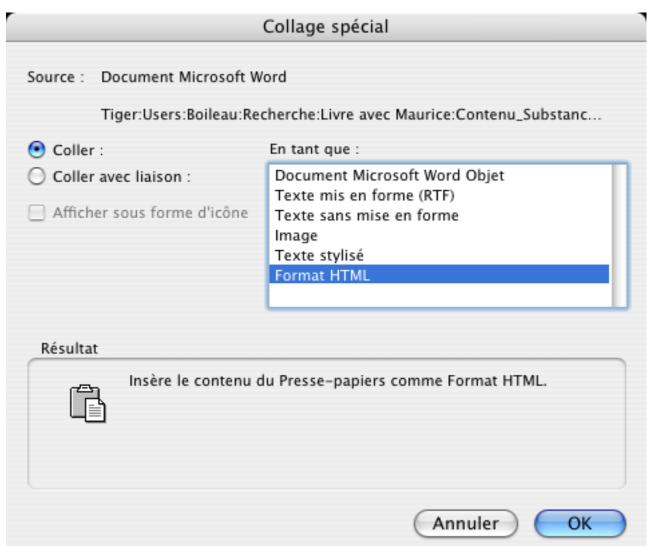


Figure 1.5.10 Fenêtre de dialogue apparaissant après un « Collage spécial... ».

On constate que l'objet copié a été placé dans le presse-papier sous au moins[1] 6 formes. Dans les exercices (Exercice **5**), vous pourrez constater quelles sont les caractéristiques (polices et styles, présence et disposition des graphiques, etc.) qui sont préservées quand on utilise chacune de ces formes lors de la recopie. Soulignons au passage la forme « Image », qui permet de coller un fragment d'un document *Word* dans un logiciel graphique[2].

De la même façon, après avoir copié une formule produite avec l'éditeur d'équations de *Word*, on constate qu'elle est placée dans le presse-papier sous, au moins, les 2 formes suivantes : « Equation Microsoft Objet » et « Image ». Si c'est une image (graphique matriciel) qui est copiée dans le presse-papier, elle est placée sous les 3 formes suivantes : « Document Microsoft Word Objet » (« Objet Dessin Microsoft Office » sous Windows), « Texte mis en forme (RTF) » et « Image ».

Et si c'est un dessin (graphique vectoriel) qui est copié dans le presse-papier, il est placé, au moins, sous les 2 formes suivantes : « Image » et « Objet Dessin Microsoft Office ». *Word* nous permet donc apparemment de transformer un graphique vectoriel (dessin) en graphique matriciel (image) en choisissant la forme « Image » lors d'un « Collage spécial... » d'un dessin. Ceci permet, entre autres choses, de choisir la disposition « Inséré dans le texte » pour le graphique ainsi transformé. Notons cependant que l'image ainsi obtenue comporte encore des données vectorielles, et n'est donc pas un graphique purement matriciel. Pour illustrer ceci, revenons à notre petit segment de la section 1.3 : — . Si nous faisons une copie de ce dessin (graphique vectoriel), que nous en faisons un collage spécial sous forme d'image, puis que nous agrandissons le tout par un facteur dix, nous obtenons le résultat suivant :

Tout se passe ici comme si **Word** avait aussi gardé trace des extrémités, qu'il avait transformées en des disques de rayons dix fois plus grands. Mais ces graphiques hybrides vectoriels-matriciels n'ont pas, en général, la qualité des graphiques purement vectoriels lors de transformations. Ainsi, dans le cas du tracé des étapes menant au **Flocon de Koch**, les graphiques hybrides se comportent aussi mal que les graphiques purement matriciels étudiés à la section 1.3.

Nous voici donc arrivé au terme de notre chapitre sur la confection de textes mathématiques avec *Word*. Nous savons maintenant comment définir et insérer des équations dans un texte. Nous savons aussi comment créer des graphiques vectoriels simples dans *Word*, pour les disposer ensuite à notre guise dans un document.

Mais pour obtenir des graphiques plus complexes, il faudra pousser notre étude de *Word* plus avant (ce que nous ferons au chapitre suivant), ou apprendre à utiliser d'autres logiciels mathématiques (ce que nous ferons plus tard dans le livre). Et quand viendra le temps d'insérer des graphiques mathématiques obtenus à l'aide d'autres logiciels, notre étude des principaux types de graphiques (vectoriels et matriciels) et du mécanisme du copier-coller nous aidera à faire de bons choix.

---

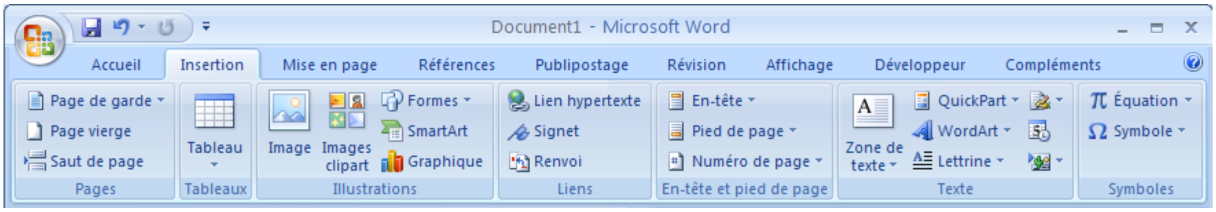
[1] Si l'objet avait été disposé dans le presse-papier par un logiciel autre que *Word*, on peut supposer que les formats non reconnus par *Word* n'apparaîtraient pas dans la liste.

[2] Mentionnons qu'une « Image » *Word*, qui devrait être en principe un graphique matriciel, contient parfois certaines informations vectorielles, qui ne sont pas toujours bien reconnues par les divers logiciels graphiques.

1.6 Appendice : les outils de dessin dans Word 2007 pour Windows

Pour insérer une forme

Cliquer sur l'onglet « Insertion » puis, dans le ruban illustré ci-dessous, sur le bouton « Formes »,

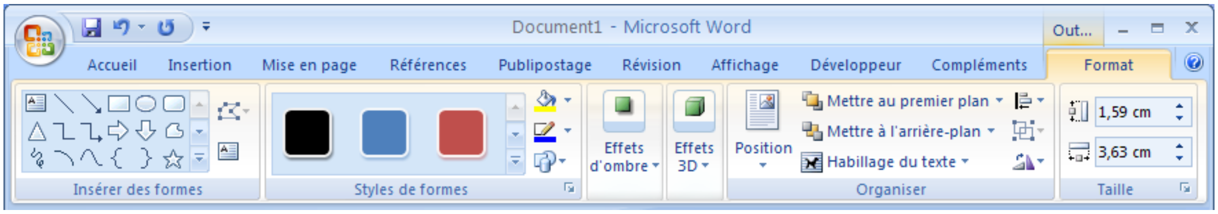


ce qui fait apparaître la fenêtre suivante:

Il ne reste plus qu'à choisir la forme désirée.

Pour agir sur une forme existante

On commence par sélectionner la forme, ce qui fait apparaître le ruban ci-dessous :



dans lequel on retrouve les outils permettant d'agir sur le remplissage, l'alignement, le groupement, la taille, la position, etc.
De même un clic droit sur une forme sélectionnée fait apparaître un menu local offrant les mêmes possibilités que celui mentionné dans le texte pour les autres versions de Word.

1.7 Exercices

1- *Utilisation de l'éditeur d'équations*

Reproduisez les formules suivantes à l'aide de l'éditeur d'équations :

a) $\dfrac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

b) $\begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x\cos(\alpha) - y\sin(\alpha) \\ x\sin(\alpha) + y\cos(\alpha) \end{pmatrix}$

c) $|x| = \begin{cases} x & \text{si} \quad x \geq 0 \\ -x & \text{si} \quad x < 0 \end{cases}$

d) $f(x) = \begin{cases} x\sin(\frac{1}{x}) & \text{si } x \neq 0 \\ 1 & \text{si } x = 0 \end{cases}$

e) $\begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc$

f) $\overrightarrow{AB} + \overrightarrow{BC} = \overrightarrow{AC}$

g) $123{,}45\overline{6789}$ et $0{,}\underbrace{12}_{7\,fois}3$

h) $\displaystyle\sum_{i=0}^{n} x^i = \frac{1 - x^{n+1}}{1 - x}$ si $x \neq 1$

2- *Utilisation de l'éditeur d'équations*

a) Reproduisez l'expression suivante : $y = \sqrt{\dfrac{a}{b}}\sin(\alpha)$.

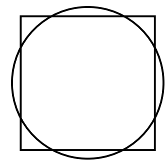
b) Sans la réécrire complètement, modifiez l'expression obtenue en a) pour obtenir :

$$y = \sqrt{\frac{a}{b}\sin(\alpha)}$$

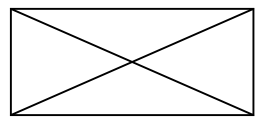
c) Sans la réécrire complètement, modifiez l'expression obtenue en a) (ou en b)) pour obtenir : $y = \sqrt{\dfrac{a\sin(\alpha)}{b}}$.

3- *Utilisation de la barre d'outils « Dessin »*
Utilisez la barre d'outils « Dessin » pour reproduire les graphiques suivants de la façon la plus exacte possible. Décrivez votre démarche et discutez le degré d'exactitude atteint :

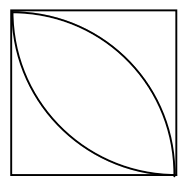
a) Un cercle et un carré de même aire et de même centre



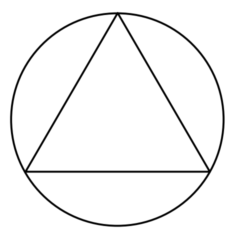
b) Un rectangle et ses diagonales.



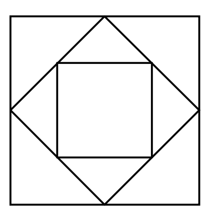
c) Un carré et deux quarts de cercles inscrits :



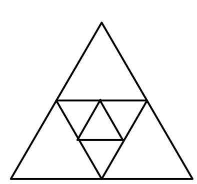
d) Un triangle équilatéral et son cercle circonscrit :



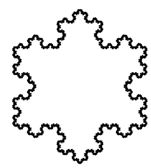
e) Trois carré emboîtés, le deuxième et le troisième ayant pour sommets les milieux des côtés du précédent:



f) Trois triangles équilatéraux emboîtés, le deuxième et le troisième ayant pour sommets les milieux des côtés du précédent:

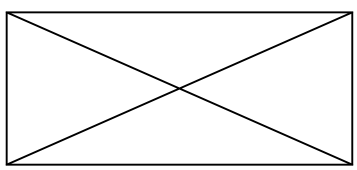


4- *Formats graphiques : vectoriel versus matriciel*
Dans le fichier des exemples de ce chapitre vous trouverez le graphique vectoriel suivant :

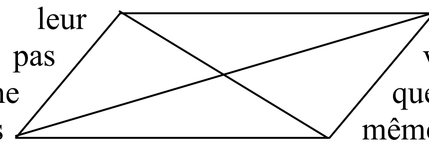


a) Copiez ce graphique, puis faites en un « Collage spécial » sous un format matriciel (Image)
b) Sélectionnez les deux graphiques, faites un clic droit, choisissez « mettre en forme l'objet.. » dans le menu contextuel et fixez la taille à 500%. Les deux graphiques sont grossis par un facteur de 5 : constatez la différence entre les formats matriciel et vectoriel.

5- *Placement de graphiques dans un texte*
a) Reproduisez le paragraphe et les graphiques suivants :



Les diagonales d'un rectangle ont la même longueur et se coupent en leur

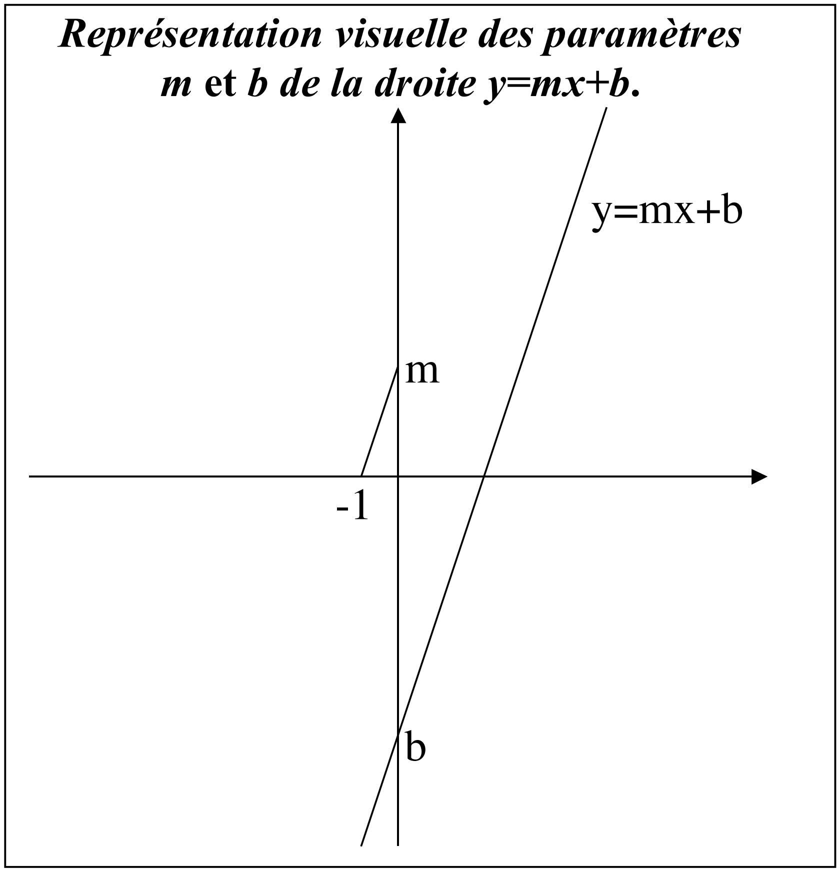


milieu. Cette propriété n'est pas vraie d'un parallélogramme quelconque car dans ce cas même si les diagonales se coupent en leur milieu, elle n'ont pas la même longueur si le parallélogramme n'est pas un rectangle.

b) Avec le paragraphe obtenu en a), expérimentez avec les fonctions d'habillage des graphiques dans un texte. En fonction de l'impact sur la lisibilité lesquels choisiriez-vous?
c) Copiez le paragraphe obtenu en a), puis utilisez la fonction de « Collage spécial » pour expérimenter avec les différents formats de « collage » qu'elle propose. Classifiez ces différents formats du plus fidèle au moins fidèle.

6- *Rédaction de textes mathématiques avec graphiques*
Recréez tous les encadrés ci-dessous en utilisant les outils graphiques (gestuels et/ou textuels) de Microsoft Word. Décrivez ensuite de façon détaillée (toujours dans Microsoft Word) comment vous avez procédé.

a)



b)

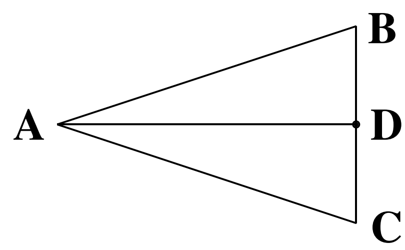
Quelques questions en géométrie

Dans la figure ci-contre, on a:

$\overline{AB} \cong \overline{AC}$ *et* $\overline{BD} \cong \overline{DC}$

A-t-on aussi :

$\angle BAD \cong \angle CAD$ **?**

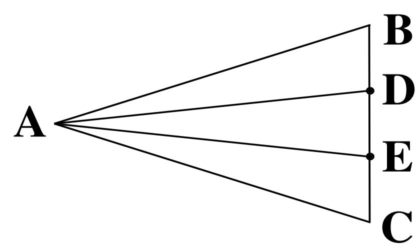


Dans la figure ci-contre, on a:

$\overline{AB} \cong \overline{AC}$ *et* $\overline{BD} \cong \overline{DE} \cong \overline{EC}$

A-t-on aussi :

$\angle BAD \cong \angle DAE \cong \angle EAC$ **?**

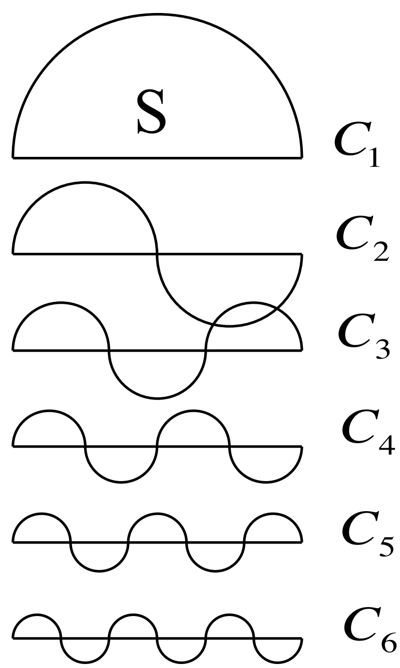


c)

Une question de limite

Soit S un segment.
Supposons qu'on veuille approximer S par une suite de courbes formées de demi-cercles comme dans le dessin ci-contre.
Dans quel sens peut-on dire que la suite de courbes C_1, C_2, C_3, C_4, C_5, C_6, ... tend vers le segment S?
On pourrait alors s'attendre à ce que la suite des longueurs des courbes C_i tende vers la longueur du segment S.
Faites le calcul, en supposant que le segment S mesure 1 unité.

Vous serez peut-être surpris !



1.8 Projets

1- *Rédaction d'un texte mathématique*

À l'aide de Microsoft Word et son éditeur d'équations, saisir le texte manuscrit ci-dessous en cherchant à respecter (et améliorer) le formatage original.

$\boxed{\text{Trouver les solutions de l'équation } aX^2+bX+c=0, \text{ où } a\neq 0}$

Soit donc $aX^2+bX+c=0$.

Comme $a\neq 0$, on peut diviser les deux membres de l'équation pour obtenir l'équation équivalente

$$X^2+\frac{b}{a}X+\frac{c}{a}=0.$$

De même, on peut soustraire $\frac{c}{a}$ aux deux membres

$$X^2+\frac{b}{a}X=-\frac{c}{a}$$

puis ajouter $\frac{b^2}{4a^2}$ (toujours aux deux membres)

$$X^2+\frac{b}{a}X+\frac{b^2}{4a^2}=\frac{b^2}{4a^2}-\frac{c}{a}$$

ce qui va nous permettre de factoriser le membre de gauche et de mettre le membre de droite sous un dénominateur commun:

$$\left(X+\frac{b}{2a}\right)^2=\frac{b^2-4ac}{4a^2}.$$

Si on extrait la racine carrée des deux membres d'une égalité, les résultats doivent être égaux à leur signe près.

D'où l'on a

$$X+\frac{b}{2a}=\frac{\sqrt{b^2-4ac}}{2a} \quad \text{ou} \quad X+\frac{b}{2a}=-\frac{\sqrt{b^2-4ac}}{2a}$$

c'est-à-dire

$$X=\frac{-b+\sqrt{b^2-4ac}}{2a} \quad \text{ou} \quad X=\frac{-b-\sqrt{b^2-4ac}}{2a}$$

ce qu'on a coutume d'écrire

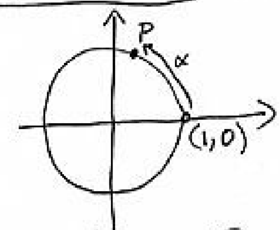
$$X=\frac{-b\pm\sqrt{b^2-4ac}}{2a}$$

2- *Rédaction d'un texte mathématique avec graphiques*
À l'aide de Microsoft Word et son éditeur d'équations, saisir le texte manuscrit ci-dessous en cherchant à respecter (et améliorer) le formatage original[1].
Les graphiques sont disponibles dans le fichier des exemples du chapitre.

Une preuve de l'identité trigonométrique

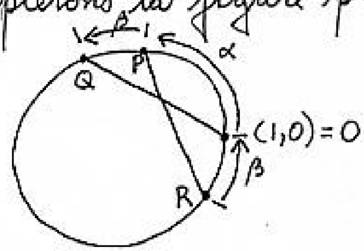
$$\cos(\alpha+\beta) = \cos\alpha\cos\beta - \sin\alpha\sin\beta$$

Considérons le cercle d'équation $x^2+y^2=1$, qui est centré à l'origine et de rayon 1.



Si P est un point sur ce cercle et si α est la longueur orientée de l'arc allant de $(1,0)$ vers P, alors on a que $P = (\cos\alpha, \sin\alpha)$.

Complétons la figure précédente comme suit:



Notons en passant qu'il y aurait plusieurs autres cas de figure, selon les signes de α et β. Notre raisonnement devra être valide pour tous ces cas.

On remarque que les coordonnées des points Q et R sont

$$Q = (\cos(\alpha+\beta), \sin(\alpha+\beta))$$
$$R = (\cos(-\beta), \sin(-\beta)) = (\cos\beta, -\sin\beta).$$

Le coeur de notre argumentation sera basé sur la constatation suivante:

Les cordes $\overline{OQ}$ et $\overline{PR}$ sont de même longueur puisqu'elles sous-tendent des arcs de même mesure (soit $\alpha+\beta$).

[1] Pour les flèches nous conseillons d'utiliser des pointes pleines. En effet l'utilisation de pointes creuses peut légèrement modifier l'apparence du graphique avec *Word 2003 pour Windows*.

Comme la longueur de la corde $\overline{OQ}$ est
$$\sqrt{[\cos(\alpha+\beta)-1]^2+[\sin(\alpha+\beta)-0]^2}$$
et que la longueur de la corde $\overline{PR}$ est
$$\sqrt{[\cos\alpha-\cos\beta]^2+[\sin\alpha-\sin\beta]^2},$$
ces deux expressions sont donc égales:
$$\sqrt{[\cos(\alpha+\beta)-1]^2+\sin^2(\alpha+\beta)}=\sqrt{[\cos\alpha-\cos\beta]^2+[\sin\alpha+\sin\beta]^2}.$$
Si on élève les deux membres de cette équation au carré, et qu'on développe par la suite, on obtient:
$$\left[\cos^2(\alpha+\beta)-2\cos(\alpha+\beta)+1\right]+\sin^2(\alpha+\beta)$$
$$=\left[\cos^2\alpha-2\cos\alpha\cos\beta+\cos^2\beta\right]+\left[\sin^2\alpha+2\sin\alpha\sin\beta+\sin^2\beta\right].$$
En utilisant l'identité
$$\cos^2 x+\sin^2 x=1$$
pour $x=\alpha$, $x=\beta$ et $x=\alpha+\beta$, on peut simplifier encore et obtenir
$$2-2\cos(\alpha+\beta)=2-2\cos\alpha\cos\beta+2\sin\alpha\sin\beta,$$
d'où l'on tire finalement
$$\cos(\alpha+\beta)=\cos\alpha\cos\beta-\sin\alpha\sin\beta$$
qui est l'identité désirée.

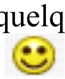
Chapitre 2

Un langage graphique pour Word

Dans ce chapitre, nous verrons comment on peut demander à *Word* de tracer des graphiques à partir de descriptions écrites. Cette approche *textuelle* est complémentaire à l'approche *gestuelle* du chapitre précédent, où nous décrivions les graphiques par des clics et des glissers de la souris. De plus, nous pourrons combiner ces deux approches pour obtenir des représentations graphiques précises d'objets mathématiques plus complexes.

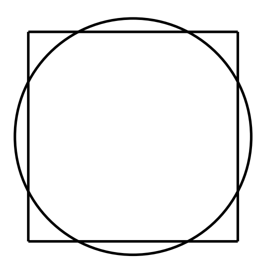
Ces descriptions textuelles de graphiques se feront dans un langage qui a été ajouté à *Word* quand nous avons installé la barre d'outils *Progiciels* au chapitre précédent. Ce langage nous permettra d'utiliser deux paradigmes (pouvant être combinés) pour décrire les graphiques : la *géométrie analytique* et la *géométrie de la tortue*. De plus, comme ce *LangageGraphique* est en fait constitué de deux modules ajoutés à l'environnement *Visual Basic for Applications*, nous pourrons avoir recours à toute la puissance d'un langage de programmation pour décrire nos graphiques. Nous en donnerons d'ailleurs quelques exemples vers la fin de ce chapitre.

2.1 Premiers pas avec LangageGraphique

Pour se faire une idée de quelques possibilités de *LangageGraphique*, il suffit de cliquer sur le bouton « Tracer la figure »  de la barre d'outils *Progiciels* : à moins que nous n'ayons apporté des changements depuis son installation, nous devrions voir apparaître successivement plusieurs fenêtres de dialogue nous permettant de préciser les valeurs des paramètres utilisés pour tracer diverses figures. Notons que l'on peut déplacer ces fenêtres de dialogue pour mieux voir les graphiques produits, et que l'on peut interrompre la démonstration en cours de route en répondant « Non » à la question « On continue ? »[1].

Après avoir visionné la démonstration, nous aimerions maintenant tracer nos propres graphiques. Nous allons commencer par quelques exemples simples, que nous avons tenté de tracer *gestuellement* au chapitre précédent : ceci nous permettra de comparer les approches *gestuelle* et *textuelle*.

Comme premier exemple, prenons le graphique du carré et du cercle de même aire. Si r est la longueur du rayon du cercle, la longueur du côté c du carré devra donc être de $r\sqrt{\pi}$.



[1] La démonstration se termine aussi quand elle ne peut obtenir la valeur d'un paramètre, que l'utilisateur clique sur le bouton « Annuler » ou entre une valeur non acceptable.

Pour entrer les instructions nécessaires, nous devons tout d'abord accéder à l'éditeur *Visual Basic* : ceci se fait par un clic sur l'icône « Visual Basic Editor » de la barre d'outils « Progiciels » : elle ressemble à ceci  pour la version *Macintosh* et à ceci  pour la version *Windows*. Nous devrions normalement voir une fenêtre intitulée

« Normal - Procedures_Utilisateur (Code) »

et contenant un texte débutant par les lignes suivantes

```
Sub Graphique()  'Placez vos commandes graphiques dans cette procédure
    Demo
End Sub
```

Si cette fenêtre n'est pas visible, suivez les indications données à la figure 2.1.1.

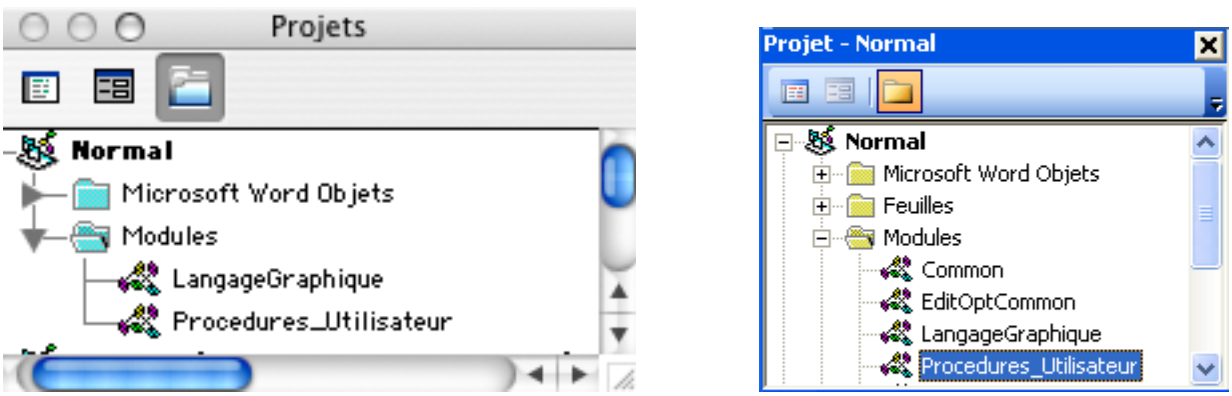


Figure 2.1.1 Pour faire apparaître la fenêtre « Normal - Procedures_Utilisateur (Code) », double-cliquer sur l'item « Procedures_Utilisateur » de la fenêtre « Projets » (version *Macintosh*) ou « Projets - Normal » (version *Windows*).

Nous remarquons que la procédure « Graphique » comporte une seule instruction, « Demo », qui a provoqué tout à l'heure l'exécution de la démonstration. Nous devons donc remplacer cette instruction par des instructions pour tracer un cercle et un carré de même aire. Si nous choisissons que notre cercle soit centré à l'origine et de rayon 100, les instructions suivantes feront l'affaire :

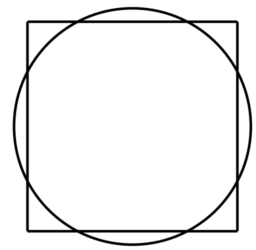
```
Sub Graphique()  'Placez vos commandes graphiques dans cette procédure
    r = 100
    c = r * Sqr(pi)
    Cercle 0 , 0 , r
    Rectangle  - c/2 ,  - c/2 ,   c/2 ,   c/2
End Sub
```

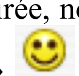
Nous reviendrons plus tard sur l'écriture et le fonctionnement des instructions ci-dessus. Notons que nous aurions aussi pu taper directement

```
Sub Graphique()
    Cercle  0 , 0 , 100
    Rectangle  -100*Sqr(pi) / 2,  -100*Sqr(pi) / 2 ,  100*Sqr(pi) / 2,  100*Sqr(pi) / 2
End Sub
```

L’important est de transmettre des valeurs numériques bien définies aux commandes « Cercle » et « Rectangle ». Notons en passant que l’origine du système d’axes utilisé est située à peu près au milieu d’une page *Word* et que l’unité de longueur utilisée est le « *point* », qui équivaut à $\frac{1}{72}$ pouce[1] soit environ 0,35 mm. Mais ceci a relativement peu d’importance puisqu’il nous sera facile de changer *gestuellement* la position et les dimensions du graphique ainsi obtenu.

Quand nous avons fini d’entrer nos instructions, nous retournons à notre page *Word* en choisissant l’item « Fermer et retourner à Microsoft Word » du menu « Fichier ». Ne fermons pas inutilement la fenêtre de l’éditeur Visual Basic, car il nous faudrait alors la réouvrir à notre prochaine visite.



Pour obtenir (enfin!) la figure désirée, nous devons maintenant faire un *clic* sur le bouton « Tracer la figure » . Si tout se passe bien, nos efforts seront récompensés par l’obtention (dans une nouvelle fenêtre) du graphique ci-contre, que nous pourrons modifier (par exemple en changeant ses dimensions) avant de la *copier* et de la *coller* dans le document *Word* de notre choix. Notez en passant que notre tâche sera facilitée par le fait que *LangageGraphique* regroupe automatiquement tous les éléments tracés, ce qui facilite tant les redimensionnements que les copies.

Si *Visual Basic for Applications* nous signale une erreur, nous serons amenés à la ligne où l’erreur a été détectée. On doit alors vérifier que l’on a bien tapé les instructions exactement comme elles apparaissent ci-dessus, en n’oubliant pas les espaces, virgules et parenthèses. Veuillez noter cependant que *VBA* (*Visual Basic for Applications*) ne fait pas de distinction entre les lettres minuscules et majuscules : par exemple, « r » et « R » dénotent la même variable. On corrige donc au besoin la ou les coquilles, on retourne à *Word* et on redemande le tracé.

Dans les sections qui suivent, nous décrirons les diverses commandes de *LangageGraphique*, en illustrant le tout par quelques exemples. Mais avant de passer à cela, disons un mot des variables utilisées en informatique. Nous avons déjà rencontré les variables « r » et « c » dans la première description de notre graphique. Au départ, nous devons[2] assigner une valeur à nos variables : c’est ce qu’on réalise avec les instructions « r = 100 » ou « c = r*Sqr(pi) ». Par la suite, nous pourrons utiliser ces variables dans les commandes subséquentes (« Cercle 0,0,r » par exemple).

Notation informatique versus notation mathématique

Il importe ici de souligner que, contrairement à ce qui se passait quand on voulait insérer des formules dans un document *Word*, on ne peut utiliser la notation mathématique usuelle dans *LangageGraphique* (comme dans la plupart des environnements de programmation) : on doit inscrire « r*Sqr(pi) » (en n’oubliant pas le signe de multiplication *) et non pas $r\sqrt{\pi}$. De même, on constate l’utilisation de l’anglais dans *Visual Basic for Applications* : la commande « Sqr » évoque l’expression anglaise « square root », qui signifie « racine carrée ».

[1] Ceci correspond à un pixel pour certaines dimensions et résolutions de l’écran.

[2] Certains environnements de programmation (dont *Visual Basic for Applications*) assignent automatiquement des valeurs par défaut aux variables, alors que d’autres signalent une erreur dès qu’on tente d’utiliser une variable qui n’a pas été affectée explicitement d’une valeur. Nous recommandons de toujours donner explicitement des valeurs aux variables utilisées, ce qui évite souvent de faire des erreurs et a le mérite de rendre le programme plus clair.

L'égalité informatique diffère de l'égalité mathématique

Veuillez noter que le signe d'égalité est employé ici dans un sens différent du sens mathématique usuel : celui d'affectation. L'instruction « r = 100 » peut être interprétée comme signifiant « aller placer la valeur 100 dans une *boîte* nommée *r* », et certains langages de programmation utilisent une notation différente de l'égalité pour désigner l'affectation. Par exemple, on utilisera « r := 100 » en *Pascal* et « r ← 100 » en *APL*. Cette interprétation *informatique* de l'égalité comme une simple affectation permet de donner un sens à des expressions comme « n = n+1 » : prendre la valeur contenue dans la boîte *n*, lui ajouter 1, puis mettre le résultat obtenu dans la boîte *n*.

2.2 Commandes de géométrie analytique

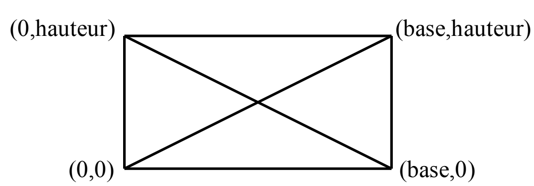
Dans notre premier exemple, nous avons rencontré les commandes « Cercle » et « Rectangle ». Ce sont deux exemples de commandes décrivant des formes à l'aide de la géométrie analytique, qui utilise coordonnées et mesures (de longueurs et d'angles). Les commandes de géométrie analytique disponibles dans *LangageGraphique* sont décrites à la figure 2.2.2.

Commande	Description
Segment x1, y1, x2, y2 *Exemple : Segment 0,0,100,50*	Trace un segment joignant deux points de coordonnées (x1, y1) et (x2, y2).
Rectangle x1, y1, x2, y2 *Exemple : Rectangle 0,0,100,50*	Dessine un rectangle (sans remplissage) dont on spécifie les coordonnées d'une diagonale.
RectanglePlein x1, y1, x2, y2 *Exemple : RectanglePlein 0,0,100,50*	Dessine un rectangle rempli avec la couleur de remplissage.
Ellipse x1, y1, x2, y2, Remplir, Angle *Exemple : Ellipse 0,0,100,50,true,45*	Trace l'ellipse déterminée par le rectangle, et la tourne ensuite de « Angle » degrés. Remplir peut prendre les valeurs « true » ou « false ».
Cercle x, y, R *Exemple : Cercle 0,0,100*	Trace le cercle de centre et rayon donnés
Disque x, y, R *Exemple : Disque 0,0,100*	Trace le disque (cercle plein) de centre et rayon donnés
Arc x, y, rayon, depart, arrivee *Exemple : Arc 0,0,100,45,135*	Trace un arc déterminé par un cercle, un angle de départ et un angle d'arrivée (sens +). Notons que tous les angles sont en degrés.
Grille x, y, dx, dy, nx, ny *Exemple : Grille 0,0,10,10,20,10*	Dessine une grille à la position (x,y) • nx colonnes de dimension dx • ny lignes de dimension dy

Figure 2.2.2 Les commandes de géométrie analytique de *LangageGraphique*.

Il est maintenant facile de réaliser le rectangle muni de ses deux diagonales que nous avons évoqué au chapitre précédent. Une façon possible de procéder est d'utiliser les commandes suivantes

```
Sub Graphique()
    base = 100
    hauteur = 50
    Rectangle 0, 0, base, hauteur
    Segment 0, 0, base, hauteur
    Segment base, 0, 0, hauteur
End Sub
```

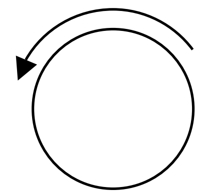


traçant le rectangle de sommets (0,0), (base,0), (base,hauteur) et (0,hauteur), et ses diagonales.

Si désiré, nous pourrons par la suite déplacer et redimensionner gestuellement (avec la souris) le graphique obtenu. Notons en passant que les noms de variables peuvent comporter plusieurs lettres, mais qu'il est plus prudent de n'utiliser que des lettres sans accents pour les former.

Terminons cette section en réalisant un graphique faisant appel à la fois aux approches textuelle et gestuelle. Cet exemple est peu intéressant en soi, mais illustre une technique qui vous servira dans les exercices. Dans un premier temps, nous décrivons textuellement notre graphique comme suit :

```
Sub Graphique()
    rayon = 30
    angleDepart = 37
    angleArrivee = 163
    Cercle 0, 0, rayon
    Arc 0, 0, rayon * 1.25, angleDepart, angleArrivee
End Sub
```



Il ne nous reste plus qu'à doter notre arc d'une flèche, ce qui n'a pas été prévu dans *LangageGraphique*. Ceci peut cependant se faire de manière gestuelle :

- On choisit « Groupe » ➤ « Dissocier » dans le menu local associé au graphique.
- On sélectionne seulement l'arc puis on le dote d'une flèche en faisant appel à l'outil approprié de la palette d'outils « Dessin ».
- On choisit « Groupe » ➤ « Regrouper » dans le menu local associé à l'arc (maintenant doté d'une flèche). *Word* reconstituera alors le groupe que nous avions dissocié.

2.3 Commandes de géométrie de la tortue

Les premières *tortues informatiques* étaient de petits robots qui se déplaçaient sur le plancher : ils pouvaient avancer et reculer, ainsi que pivoter à droite ou à gauche[1]. Ils étaient dotés d'un crayon rétractable qui pouvait laisser une trace sur une feuille glissée sous le robot. Les élèves pouvaient ainsi réaliser des dessins en utilisant un langage informatique nommé *Logo* pour déplacer le robot. Par la suite, la tortue a migré du plancher vers les écrans des ordinateurs, où elle a gagné en vitesse et en précision.

Il ne faut cependant pas croire que l'usage de la tortue soit limité à de jeunes élèves qui ont besoin de métaphores concrètes : le pilotage de la tortue peut nécessiter de solides connaissances mathématiques[2]. On a même assisté au développement d'une *géométrie de la tortue*, avec ses définitions et ses théorèmes.

Un exemple de « théorème tortue » : le théorème du tour complet

Si, après une série de déplacements et de pivotements, la tortue revient exactement à son orientation initiale (elle regarde dans la même direction), alors la somme de tous ses pivotements est un multiple de 360º. (Précisons que les pivotements vers la droite sont affectés d'un signe positif, tandis que ceux vers la gauche sont affectés d'un signe négatif.)
Application : Pour faire un triangle équilatéral, il faudra pivoter trois fois de 120º = 360º/3.

[1] Certaines versions étaient dotés de senseurs pouvant détecter les obstacles. On pouvait ainsi les faire circuler dans des labyrinthes en étant « conscients » des murs.

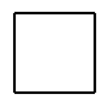
[2] Voir par exemple le livre *Turtle Geometry : The Computer as a Medium for Exploring Mathematics* d'Harold Abelson et Andrea diSessa, publié chez *MIT Press* en 1980.

LangageGraphique est doté de commandes pour piloter une tortue (qui restera malheureusement toujours invisible). Au départ, la tortue est positionnée à l'origine et pointe vers le haut. Les commandes de géométrie de la tortue disponibles dans *LangageGraphique* sont décrites à la figure 2.3.3. Notez que certaines d'entre elles possèdent aussi une forme abrégée : par exemple, on peut utiliser indifféremment « Avance 50 » ou « Av 50 ».

Commande	**Forme abrégée**	**Description**
Avance distance *Exemple : Avance 30*	Av *Av 30*	Fait avancer la tortue d'un certain nombre de pas (points). Pour reculer, utiliser un nombre de pas négatif.
Droite angle *Exemple : Droite 90*	Dr *Dr 90*	Fait pivoter la tortue vers la droite d'un certain nombre de degrés. Pour pivoter à gauche, utiliser un angle négatif..
FixePosition x, y *Exemple : FixePosition 0,0*	FPos *FPos 0,0*	Fixe la position de la tortue.
FixeCap angle *Exemple : FixeCap 45*	FCap *FCap 45*	Fixe le cap de la tortue.
PosX () *Exemple : a = PosX ()*		Fonction (sans arguments) retournant la coordonnée en x de la position de la tortue.
PosY () *Exemple : b = PosY ()*		Fonction (sans arguments) retournant la coordonnée en y de la position de la tortue.
Cap () *Exemple : c = Cap ()*		Fonction (sans arguments) retournant le cap de la tortue. Le cap est l'angle (en degrés) que fait la tortue avec la verticale vers le haut.
LeveCrayon	LC	Lève le crayon de la tortue. Si la tortue se déplace, elle ne laissera pas de trace.
BaisseCrayon	BC	Baisse le crayon de la tortue. Si la tortue se déplace, elle laissera une trace.

Figure 2.3.3 Les commandes de géométrie de la tortue de *LangageGraphique*.

Voyons maintenant ces commandes-tortue à l'oeuvre. Si nous voulons tracer un carré de côté *c*, nous pouvons commander à la tortue de répéter 4 fois les deux commandes suivantes[1] : « Avance c » et « Droite 90 ». Comparons donc les instructions nécessaires pour obtenir notre carré avec la géométrie de la tortue (à gauche), et avec la géométrie analytique (à droite) :

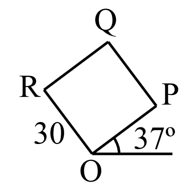


```
Sub Graphique()
    c = 30
    Avance c
    Droite 90
    Avance c
    Droite 90
    Avance c
    Droite 90
    Avance c
    Droite 90
End Sub
```

```
Sub Graphique()
    c = 30
    Segment 0, 0, 0, c
    Segment 0, c, c, c
    Segment c, c, c, 0
    Segment c, 0, 0, 0
End Sub
```

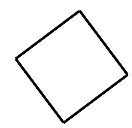
[1] La dernière des commandes (le dernier « Droite 90 ») n'est pas nécessaire. Mais il s'avère utile, quand on veut tracer un graphique en plusieurs phases indépendantes, de laisser la tortue dans le même état (position et orientation) avant et après le tracé de chaque phase.

On ne voit pas encore l'avantage de doter *LangageGraphique* de commandes-tortue. Modifions donc très légèrement notre exemple pour que l'intérêt des commandes-tortue devienne apparent. Supposons donc que nous voulions maintenant tracer un carré auquel nous avons fait subir une rotation de 37º autour de l'origine, comme illustré dans le graphique ci-contre. Pour le tracé via la tortue, il suffit d'ajouter au départ une instruction « Droite -37 » (qui fait pivoter la tortue de 37º *vers la gauche*) pour donner une orientation convenable avant de commencer le tracé. Voici les instructions-tortues après ce changement :




```
Sub Graphique()
        Droite - 37
        c = 30
        Avance c
        Droite 90
        Avance c
        Droite 90
        Avance c
        Droite 90
        Avance c
        Droite 90
End Sub
```

Seule cette commande a été ajoutée.



Pour le tracé en termes de coordonnées cartésiennes, il faut d'abord calculer les coordonnées des points P, Q et R comme suit[1] :

- $P = \left(30\cos\left(37^\circ\right), 30\sin\left(37^\circ\right)\right)$
- $Q = \left(30\sqrt{2}\cos\left(37+45\right)^\circ, 30\sqrt{2}\sin\left(37+45\right)^\circ\right)$
- $R = \left(30\cos\left(37+90\right)^\circ,\ 30\sin\left(37+90\right)^\circ\right)$

On peut ensuite modifier les instructions précédentes pour les adapter aux coordonnées que nous venons de calculer :

```
Sub Graphique()
        c = 30
        d = c * Sqr(2)
        Segment 0, 0, c * CosD(37 + 90), c * SinD(37 + 90)
        Segment c * CosD(37+90), c * SinD(37+90), d * CosD(37+45), d * SinD(37+45)
        Segment d * CosD(37 + 45), d * SinD(37 + 45), c * CosD(37), c * SinD(37)
        Segment c * CosD(37), c * SinD(37), 0, 0
End Sub
```

Veuillez noter l'utilisation des fonctions « CosD » et « SinD » qui supposent que leurs arguments sont exprimés en degrés. Ces fonctions sont rendues disponibles dans *LangageGraphique* car, dans *VBA*, toutes les fonctions trigonométriques supposent que les arguments sont en radians.

---

[1] La vérification de ces calculs est laissée au lecteur. Notons au passage qu'on doit recourir à des mathématiques plus sophistiquées : le théorème de Pythagore (pour calculer la longueur de la diagonale) et un peu de trigonométrie.

L'exemple du carré incliné illustre bien l'un des avantages de l'utilisation de la tortue : le tracé se fait *relativement* à l'état (position et orientation) de la tortue. C'est un peu comme si la tortue pouvait transporter sur son dos un système d'axes, que l'on peut positionner à notre guise pour nous simplifier la vie (alors que le tracé en termes de coordonnées est décrit en termes d'un système d'axes fixe).

Il ne faut pas croire cependant que la géométrie de la tortue soit toujours la meilleure approche : par exemple, comme la tortue ne peut tracer que des segments (en avançant), elle ne peut pas produire de cercles. (Traditionnellement, la tortue utilise des polygones réguliers avec un grand nombre de côtés pour obtenir des représentations approximatives des cercles.)

2.4 Commandes diverses

Nous allons maintenant voir comment modifier certaines caractéristiques physiques de nos tracés : couleur et épaisseur des traits, couleur de remplissage des formes, etc. Ceci se fera à l'aide des commandes présentées à la figure 2.4.4.

Commande	Forme abrégée	Description
ViderEcran	VE	Efface tout le dessin
CouleurCrayon R, V, B *Exemple : CouleurCrayon 0,0,0*	CC *CC 0,0,0*	Détermine la couleur du crayon. Le paramètres peuvent varier de 0 à 255.
TailleCrayon épaisseur *Exemple : TailleCrayon 2*	TC *TC 2*	Détermine l'épaisseur du crayon.
DebutRemplir		Débute un polygone à remplir.
FinRemplir		Fin du polygone à remplir.
CouleurRemplissage R, V, B *Exemple : CouleurRemplissage 255,255,0*	CR *CR 0,0,0*	Détermine la couleur des remplissages. Le paramètres peuvent varier de 0 à 255.
Dessous Dessus		La forme courante est placée *au desssous* ou *au dessus* de toutes les autres.
Hasard (min, max) *Exemple : CC Hasard (0,255),0,0*		Fonction retournant un nombre entier x choisi au hasard tel que min $\leq x \leq$ max.
Texte chaîne, x, y, police, taille, gras, italique *Exemple : Texte "Bonjour",0,0, "Times",12,true,false*		Dessine une chaîne de caractères à la position et avec les caractéristiques spécifiées.

Figure 2.4.4 Les commandes modifiant l'apparence des formes de *LangageGraphique*.

Avant d'utiliser quelques-unes de ces commandes, disons quelques mots sur la façon de décrire les couleurs. Une couleur est ici spécifiée par trois composantes (rouge, verte et bleue)[1] qui reflètent à la fois la physiologie de la vision humaine (les cônes du fond des yeux) et la structure des écrans d'ordinateurs (les composantes colorées d'un pixel). Plutôt que d'utiliser des pourcentages, nous avons choisi de décrire l'importance relative de chaque composante colorée par un entier variant de 0 (dénotant l'absence) à 255 (dénotant une présence maximale), ce qui correspond à la réalité informatique où un octet est utilisé à cette fin. Le tableau ci-dessous établit la correspondance entre certaines couleurs et leurs codes (R,V,B).

(R,V,B)	(0,0,0)	(255,255,255)	(255,0,0)	(0,255,0)	(0,0,255)	(0,255,255)	(255,0,255)	(255,255,0)
Couleur	noir	blanc	rouge	vert	bleu	cyan	magenta	jaune

[1] Voir, par exemple, http://fr.wikipedia.org/wiki/RVB et les divers liens qui en émanent.

Si l'on désire trouver le code (R,V,B) correspondant à une couleur spécifique, *Word* peut nous aider : il suffit de tracer un rectangle (à partir de la palette d'outils « Dessin »), puis de le remplir (via l'outil « Couleur de remplissage » dont l'icône est un pot de peinture[1]) en choisissant « Autres couleurs... ».

Avec *Word* 2004 pour *Macintosh*, une fenêtre de dialogue apparaît, et on choisit le mode « Curseurs RVB » : en déplaçant les trois curseurs, on peut essayer d'obtenir la couleur recherchée (voir figure 2.4.5). Si cela est plus facile, on peut aussi retrouver la couleur en utilisant une autre représentation, puis passer au mode « Curseurs RVB » pour lire les valeurs des trois composantes.

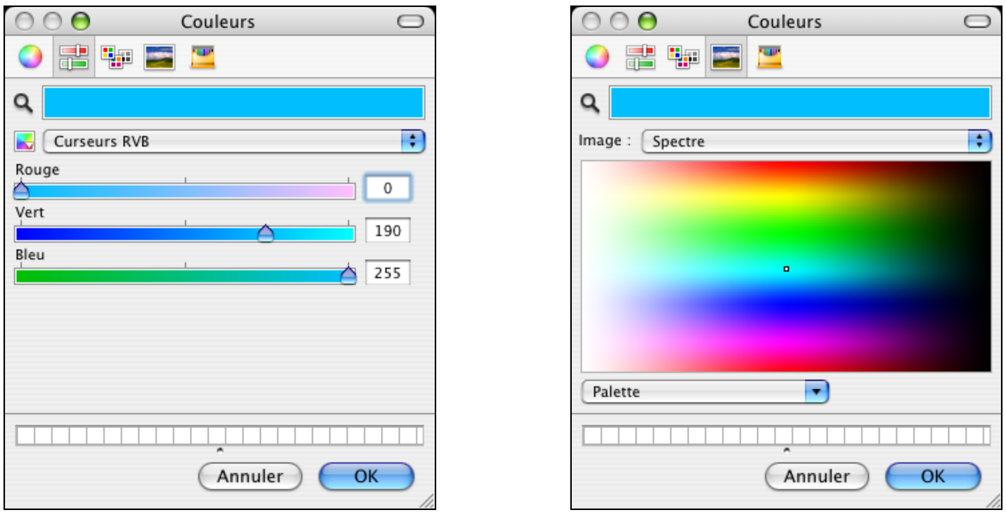


Figure 2.4.5 Deux façons de choisir une couleur dans *Word 2004* pour *Macintosh*.

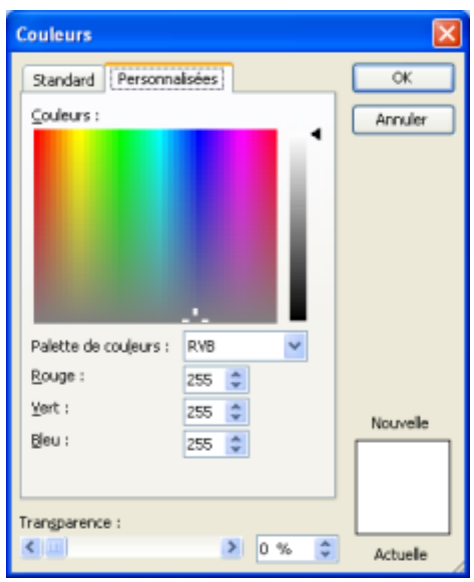
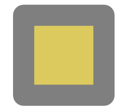


Figure 2.4.6 Deux façons de choisir une couleur dans *Word 2003* pour Windows.

[1] Voir la section 1.6 pour savoir comment l'atteindre dans *Word 2007* pour *Windows*.

Avec *Word 2003* et *2007* pour *Windows*, une fenêtre de dialogue apparaît : on clique sur l'onglet « Personnalisées », puis on choisit RVB comme « Palette de couleurs ». On peut essayer d'obtenir la couleur recherchée en modifiant les valeurs associées à « Rouge », « Vert » et « Bleu ». On peut aussi retrouver les composantes RVB d'une couleur donnée en cliquant sur celle-ci dans la fenêtre de couleur.



Illustrons quelques-unes des commandes précédentes par un exemple simple : le tracé d'un carré dont la frontière consiste en des segments gris d'épaisseur 8 et qui est rempli d'une couleur choisie au hasard (voir ci-contre). Voici la liste des instructions utilisées pour réaliser ce graphique :

```
Sub Graphique()
        TailleCrayon 8
        CouleurCrayon 127, 127, 127
        CouleurRemplissage Hasard(0, 255), Hasard(0, 255), Hasard(0, 255)
        DebutRemplir
        c = 30
        Segment 0, 0, 0, c
        Segment 0, c, c, c
        Segment c, c, c, 0
        Segment c, 0, 0, 0
        FinRemplir
End Sub
```

Notons que toutes les commandes pour tracer les divers segments constituants doivent être placées entre un « DebutRemplir » et un « FinRemplir » afin d'informer *LangageGraphique* du contour de la forme à remplir. Soulignons aussi en général que tout nombre peut être remplacé par une expression calculant un nombre : c'est ici le cas pour « CouleurRemplissage », où les nombres spécifiant les trois composantes sont remplacés par des expressions « Hasard(0,255) ».

Précautions à prendre pour remplir des formes polygonales

Quand nous voulons remplir une forme polygonale, les sommets doivent être joints dans un ordre de parcours de la figure. Par exemple, si nous voulons remplir un triangle, nous devrons

- tracer le segment allant du premier au second point
- puis tracer le segment allant du second au troisième point
  (et non pas le segment allant du troisième au second point)
- et enfin tracer le segment allant du troisième au premier point
  (et non pas le segment allant du premier au troisième point).

Notons que, si on trace notre polygone avec des instructions *Avance* plutôt qu'avec des instructions *Segment*, l'ordre de parcours de la figure sera automatiquement respecté.

Mentionnons, pour terminer, les commandes de la figure 2.4.7. Nous avons déjà eu l'occasion d'utiliser les fonctions « cosD » et « sinD » à la section précédente. Les autres fonctions sont d'usage moins courant. Elles seraient utiles si on voulait *imprimer*[1] des graphiques dont les

[1] Notez que les dimensions mesurées directement sur l'écran de l'ordinateur dépendent de plusieurs facteurs : les réglages du moniteur, les dimensions de l'écran la définition/résolution utilisée, le zoom choisi dans *Word*, etc. Il peut même arriver que les dimensions horizontale et

dimensions doivent être déterminées avec précision (des feuilles de papier millimétrique par exemple).

Commande	Description
sinD (x) *Exemple : a = sinD (45)*	Fonction sinus où l'angle est exprimé en degrés. La fonction sin(x) suppose que x est en radians.
cosD (x) *Exemple : b = cosD (45)*	Fonction cosinus où l'angle est exprimé en degrés. La fonction cos(x) suppose que x est en radians.
cmAunites (x) *Exemple : Avance cmAunites (1)*	Fonction qui calcule le nombre de *points* (notre unité de longueur) dans x centimètres.
unitesAcm (x) *Exemple : x = unitesAcm (1)*	Fonction qui calcule le nombre de centimètres correspondant à x *points* (notre unité de longueur).

Figure 2.4.7 Commandes pour faciliter certains calculs.

2.5 Commandes pour tracer des graphes de fonctions

LangageGraphique comporte aussi des commandes pour représenter des fonctions à une variable dans une région rectangulaire du plan réel (voir figure 2.5.8).

Commande	Description
RegionFonctions xmin, xmax, ymin, ymax *Exemple :RegionsFonctions -2*pi, 2*pi, -2, 2*	Spécifie quelle région du plan mathématique sera utilisée pour visualiser les fonctions. Par défaut, on déclare implicitement RegionFonctions -10, 10, -10, 10
DimGraphesFonctions base, hauteur *Exemple : DimGraphesFonctions 100,100*	Spécifie la base et la hauteur du rectangle qui contiendra les graphes des fonctions. Par défaut, base = 500 et hauteur = 400.
FenetreFonctions gauche,droite,haut,bas *Exemple :FenetresFonctions 0, 300, 0, 200*	Spécifie quelle région du plan informatique (le document Word) sera utilisée pour représenter les fonctions. Par défaut, on déclare implicitement FenetreFonctions -50, 50, -25, 25
TracerAxes	Noter que les axes tracés se limitent (pour l'instant) à deux segments avec flèches mais sans graduations.
GrilleFonctions nbHoriz, nbVert *Exemple : GrilleFonctions 10, 10*	Trace une grille comportant *nbHoriz* colonnes et *nbVert* lignes.
TracerFonction numéro *Exemple TracerFonction 3*	Commande le tracé de la fonction dont le numéro est spécifié.
PasFonctions x *Exemple : PasFonctions 2*	Par défaut, la fonction est calculée à chaque *unité*. On peut augmenter (x<1) ou diminuer (x>1) le nombre de points utilisés.
PointsRelies PointsNonRelies	Tout programme de tracé de fonctions n'évalue la fonction qu'en un nombre fini de points. On peut spécifier si ces points seront réliés (choix par défaut) entre eux ou non via ces commandes.

Figure 2.5.8 Commandes pour tracer des graphes cartésiens de fonctions.

Les représentations ainsi obtenues sont plutôt primitives si on les compare à celles produites par des environnements tels *Graphing Calculator*, *Excel*, *Maple* ou *Cabri*. Mais les graphiques produits sont *vectoriels*, et l'on verra que l'on peut obtenir des représentations très malléables.

---

verticale d'un carré apparaîssent inégales parce que la combinaison moniteur-définition choisie produit des pixels qui ne sont pas eux-même carrés.

On se sert souvent de traceurs de courbes pour étudier l'effet de certains paramètres sur les représentations graphiques de familles de fonctions : par exemple, l'effet des paramètres *m* et *b* sur les graphes des fonctions définies par des équations du type $y = mx + b$. Pour entreprendre une telle étude avec *LangageGraphique*, on doit tout d'abord spécifier les fonctions que l'on veut utiliser : pour ce faire, on dispose de 10 fonctions, identifiées par leur numéro (de 1 à 10). Nous n'utilisons pour l'instant qu'une seule fonction (d'équation $f(x) = mx + b$, où les valeurs de *m* et de *b* devront être précisées), à laquelle nous choisissons d'assigner le numéro 1. Nous allons donc dans l'éditeur *VBA* et nous cherchons, dans la fenêtre

« Normal - Procedures_Utilisateur (Code) »,

la description de la procédure « Function F(n, x) » (voir ci-contre). Ensuite, immédiatement sous la ligne « Case 1 » (qui correspond à la description de la fonction numéro 1), nous remplaçons (si nécessaire) l'expression à droite de « F = » par la description de la fonction numéro 1, soit « m*x+b » en l'occurrence.

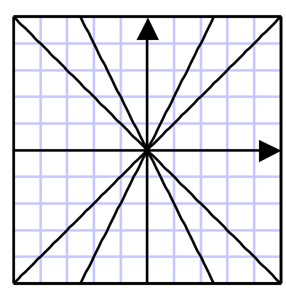
```
Function F(n, x) 'Définissez ici les ...
  F = "ND"
  On Error GoTo Erreur
  Select Case n
    Case 1
      F = Sin(x)
    Case 2
      F = 7
    Case 3
      F = Cos(x)
    Case 4
      F = Tan(x)
    Case 5
      F = SinD(x)
    Case 6
      F = CosD(x)
    Case 7
      F = TanD(x)
    Case 8
      F = 2 * x ^ 3
    Case 9
      F = 2 * x * x * x
    Case 10
      F = 0
    Case Else

  End Select
Erreur:
End Function
```

Dans ce cas, on remplace « Sin(x) » par « m*x+b ».

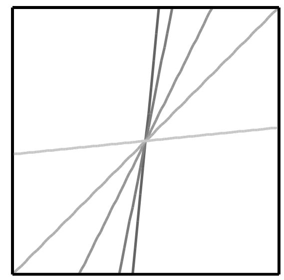
Une fois les fonctions à étudier ainsi spécifiées, nous devons préciser quels tracés devront être effectués. Supposons que nous voulions donner à la pente les valeurs -2, -1, 1, 2, tout en laissant l'ordonnée à l'origine à 0, et visualiser ces 6 graphes dans la région $[-10,10]\times[-10,10]$. Nous utiliserons alors les commandes suivantes :

```
Dim m, b As Double
Sub Graphique()
  RegionFonctions -10, 10, -10, 10
  DimGraphesFonctions 100, 100
  CC 200, 200, 255
  GrilleFonctions 10, 10
  CC 0, 0, 0
  TracerAxes
  b = 0
  m = -2
  TracerFonction 1
  m = -1
  TracerFonction 1
  m = 1
  TracerFonction 1
  m = 2
  TracerFonction 1
End Sub
```



La première ligne (« Dim m, b As Double ») mérite une explication. Quand on veut utiliser les mêmes variables (*m* et *b*, dans notre cas) dans plusieurs procédures (*F* et *Graphique*, dans notre cas), on doit les déclarer avant toutes les procédures du module. C'est précisément ce que fait notre première ligne, en spécifiant de plus que *m* et *b* doivent être des nombres décimaux à double précision (la précision maximale dans *VBA*). Mentionnons aussi que l'ordre des commandes est particulièrement important dans ce cas : **avant** de pouvoir tracer les axes, il faut savoir la région du plan représentée ainsi que les dimensions de cette représentation.. De même la valeur de la pente doit être déterminée **avant** de tracer la fonction. Soulignons enfin que la couleur du crayon (déterminée par *CC* ou *CouleurCrayon*) affecte le tracé de toutes les composantes du graphique (grille, axes et graphes de fonctions).

Cependant, au lieu de faire varier les fonctions (via leurs paramètres), on pourrait plutôt choisir de varier les *régions du plan*, tout en gardant la même fonction. Par exemple, on pourrait chercher à se faire une idée des diverses représentations graphiques de la fonction $y = x$ dans des régions de la forme $[-10,10]\times[-a,a]$, pour diverses valeurs positives du paramètre *a*. Si nous donnons à *a* les valeurs 1, 2, 5, 10 et 100, nous obtenons le graphe ci-contre. (Question : pouvez-vous dire si les « droites » deviennent plus pâles ou plus foncées quand la valeur du paramètre *a* croît ?) Soulignons que ce graphique est très différent de ceux que nous avons l'habitude de voir : il s'agit d'une même fonction, mais représentée simultanément dans plusieurs systèmes d'axes.



Une telle représentation est difficile à obtenir avec des traceurs de courbes traditionnels, mais est relativement simple à réaliser avec *LangageGraphique* :

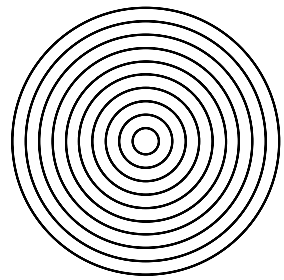
```
Sub Graphique()
    m = 1
    b = 0
    DimGraphesFonctions 100, 100
    a = 1
    RegionFonctions -10, 10, -a, a
    CC 100, 100, 100
    TracerFonction 1
    a = 2
    RegionFonctions -10, 10, -a, a
    CC 125, 125, 125
    TracerFonction 1
    a = 5
    RegionFonctions -10, 10, -a, a
    CC 150, 150, 150
    TracerFonction 1
    a = 10
    RegionFonctions -10, 10, -a, a
    CC 175, 175, 175
    TracerFonction 1
    a = 100
    RegionFonctions -10, 10, -a, a
    CC 200, 200, 200
    TracerFonction 1
End Sub
```

En terminant, mentionnons que la production de tels graphiques cartésiens peut prendre beaucoup de temps sur des ordinateurs plus lents, surtout quand on utilise un petit pas (voir *PasFonctions*) et un grand rectangle (voir *DimGraphesFonctions*). Mais votre patience sera récompensée par la précision obtenue lors de l'impression.

2.6 Coup d'oeil sur la programmation

Jusqu'à présent, tous les graphiques que nous avons tracés avec *LangageGraphique* ont été engendrés par une suite de commandes exécutées une à la fois, dans l'ordre de leur apparition dans la procédure *Graphique*. Mais *LangageGraphique* est un module ajouté à un langage de programmation à part entière (*VBA*), ce qui implique que nous avons aussi accès à toutes les techniques usuelles de programmation. Sans vouloir nous lancer dans un apprentissage de la programmation *VBA*, jetons cependant un bref coup d'oeil à une ou deux possibilités, en illustrant le tout par un exemple :

```
Sub Graphique()
   n = InputBox("Nombre de cercles", "Cercles concentriques", 10)
   For i = 1 To n
      Cercle 0, 0, 50 * i / n
   Next i
End Sub
```



Ces commandes comportent deux nouvelles instructions : *InputBox* et *For...Next*. La fonction *InputBox* nous sert à demander à l'utilisateur une information (ici, le nombre de cercles concentriques à tracer) qui pourra être utilisée par les commandes subséquentes. Pour plus de renseignements, consultez l'aide de *VBA*.

Pour accéder à l'aide de *Visual Basic for Applications* (*VBA*)

Cette aide n'est disponible que lorsque nous sommes dans l'éditeur *VBA*.

Word 2004 pour Macintosh : choisir l'item « Aide Visual Basic » du menu « Aide », puis l'onglet « Recherche » et taper « InputBox ».

Word 2003* et *2007 pour Windows : choisir l'item « Aide sur Microsoft Visual Basic » du menu « ?», puis taper « InputBox » dans le champ « Rechercher ».

Nous avons aussi utilisé l'instruction composée *For...Next*. Dans notre exemple, en supposant que l'utilisateur ait donné à *n* la valeur 4, l'instruction

```
For i = 1 To n
  Cercle 0, 0, 50 * i / n
Next i
```

revient à faire les actions suivantes

- on donne d'abord à *i* la valeur 1 et on trace un cercle centré en (0,0) et de rayon 50*1/4 = 12,5
- on donne ensuite à *i* la valeur 2 et on trace un cercle centré en (0,0) et de rayon 50*2/4 = 25
- on donne ensuite à *i* la valeur 3 et on trace un cercle centré en (0,0) et de rayon 50*3/4 = 37,5
- on donne ensuite à *i* la valeur 4 et on trace un cercle centré en (0,0) et de rayon 50*4/4 = 50

pour ensuite s'arrêter puisque la variable *i* a atteint la valeur maximale *n* = 4.

Une autre façon de comprendre notre invocation de l'instruction *For...Next* est de l'exprimer en termes d'instructions simples. Toujours dans le cas où *n* égale 4, on pourrait remplacer notre instruction *For...Next* par la suite d'instructions suivante :

i = 1 Cercle 0, 0, 50 * i / n
i = 2 Cercle 0, 0, 50 * i / n
i = 3 Cercle 0, 0, 50 * i / n
i = 4 Cercle 0, 0, 50 * i / n

Notons que, si *n* égalait 8, on devrait utiliser 8 blocs de deux instructions; en général, il nous faudrait *n* blocs de deux instructions, soit 2*n* instructions au total.

Examinons maintenant une forme plus générale de l'instruction *For...Next* :

```
For variable = valeur_initiale To valeur_finale
   liste_d'instructions_pouvant_contenir_la_variable
Next variable.
```

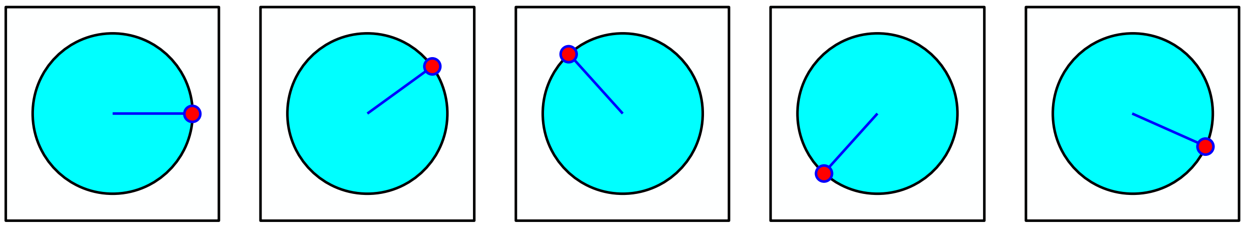
Voici succintement comment le tout fonctionne (on suppose, bien sûr, que *valeur_finale* est plus grande que *valeur_initiale*):

- *VBA* donne à la *variable* la valeur *valeur_initiale*
  puis exécute la *liste_d'instructions*
- *VBA* donne à la *variable* la valeur *valeur_initiale* + 1
  puis exécute la *liste_d'instructions*
- *VBA* donne à la *variable* la valeur *valeur_initiale* + 2
  puis exécute la *liste_d'instructions*
- ...
- *VBA* donne à la *variable* la valeur *valeur_finale*
  [car *valeur_finale* = *valeur_initiale* +(*valeur_finale* - *valeur_initiale*)]
  puis exécute la *liste_d'instructions.*

Nous aurons l'occasion d'utiliser les instructions *InputBox* et *For...Next* dans certains exercices et certains projets à la fin de ce chapitre. Mais, pour l'heure, même si nous n'avons fait qu'effleurer le sujet, nous terminons notre brève incursion dans le monde de la programmation.

2.7 La fabrication d'animations mathématiques

LangageGraphique peut aussi nous aider à fabriquer de petites « animations mathématiques ». On qualifie de « mathématique » une animation constituée d'images obtenue à partir d'une « image-maître » en faisant varier un ou plusieurs paramètres. Par exemple, en faisant varier le paramètre « angle », on fait tourner un point (et le rayon qui lui est associé) sur un cercle, comme illustré par les quelques images ci-dessous.



Décrivons tout d'abord les grandes lignes des étapes que nous devrons franchir pour obtenir notre animation :

1. (1) Nous devrons décider de la taille de l'image-maître, de sa couleur de fond, et du nombre total d'images qui seront engendrées.
2. (2) Nous devrons décrire l'image-maître.
3. (3) Nous devrons engendrer et sauvegarder les images.
4. (4) Nous devrons réunir toutes ces images dans une animation.
   Pour ce faire, nous ferons appel à la version « Pro » du logiciel « QuickTime Player », disponible au http://www.apple.com/fr/quicktime/.

Expliquons maintenant de façon plus détaillée chacune de ces étapes.

(1) Décider des paramètres de l'animation

Pour ce faire, nous introduirons dans la procédure *Graphique* une commande et une seule, dont la forme générale est

FaireFilm nombreImages, x_0, y_0, x_1, y_1, rouge, vert, bleu

qui spécifiera

- le nombre total d'images de notre animation
- deux points (x_0, y_0) et (x_1, y_1) définissant les extrémités d'une diagonale d'un rectangle contenant toutes nos images
- les composantes colorées (rouge, vert, bleu) de la couleur de fond de toutes nos images.

Dans notre exemple de point tournant sur un cercle, on pourrait donc avoir

```
Sub Graphique()
   FaireFilm 30, -240, -240, 240, 240, 255, 255, 255
End Sub
```

pour indiquer un total de 30 images avec un fond blanc, contenues dans une région décrite par les inégalités suivantes :

$$-240 \le x \le 240 \text{ et } -240 \le y \le 240.$$

Notez encore une fois que *FaireFilm* doit être **la seule commande** de la procédure *Graphique.*

(2) Description de l'image-maître

Cette description doit se faire dans la procédure *TracerImage*, qui est initialement vide et comporte un seul paramètre (n), qui indique le numéro de l'image à produire.

Dans notre exemple, le numéro de l'image n nous servira à calculer l'angle de rotation θ (en degrés) qu'on appliquera au point et au rayon : $\theta = 360 \dfrac{n-1}{nbTotalImages}$. (La vérification des calculs correspondants est laissée au lecteur.)

Notre image-maître pourra donc être décrite comme suit :

```
Sub TracerImage(n) ' Utiliser seulement pour faire un film
  r = 200
  nbTotalImages = 30
  theta = 360 * (n - 1) / nbTotalImages
  CR 0, 255, 255
  Disque 0, 0, r
  CC 0, 0, 255
  Segment 0, 0, r * CosD(theta), r * SinD(theta)
  CR 255, 0, 0
  Disque r * CosD(theta), r * SinD(theta), 5
End Sub
```

(3) Obtention de la suite d'images

Un clic sur l'icône « Tracer la figure » produira un nouveau document *Word* comptant le nombre d'images spécifiées dans la commande *FaireFilm*, à raison d'une image par page[1]. Il suffit maintenant de sauvegarder ce document en tant que page web. Dans notre exemple, choisissons d'enregistrer la page web sous le nom « Animation.htm ».

Pour sauvegarder un document en tant que page web avec *Word* 2004 pour *Macintosh*, il suffit de choisir l'item « Enregistrer en tant que page web... » du menu « Fichier », en nous assurant de choisir l'option « Enregistrer tout le fichier au format HTML » (voir la figure 2.7.9). Nous obtiendrons alors un fichier *Animation.htm* ainsi qu'un dossier *Animation_fichiers* contenant toutes les images (au format *gif* ou *png*, selon ce qui a été spécifié via le bouton « Options Web... »).

Pour sauvegarder un document en tant que page web avec *Word 2003* ou *2007* pour *Windows*, il suffit de choisir l'item « Enregistrer sous... » du menu « Fichier » (version *2003*) ou du bouton « Office » (version *2007*), en nous assurant de choisir pour type de fichier « Page Web (*.htm;*.html) » à l'exclusion de tout autre type (voir la figure 2.7.10). Nous obtiendrons alors un fichier *Animation.htm* et un dossier correspondant *Animation_fichiers* contenant toutes les images (en format *gif*).

[1] Veuillez noter que les images ne sont pas toutes placées au même endroit dans les pages. Ceci n'aura aucune incidence sur la production de l'animation.

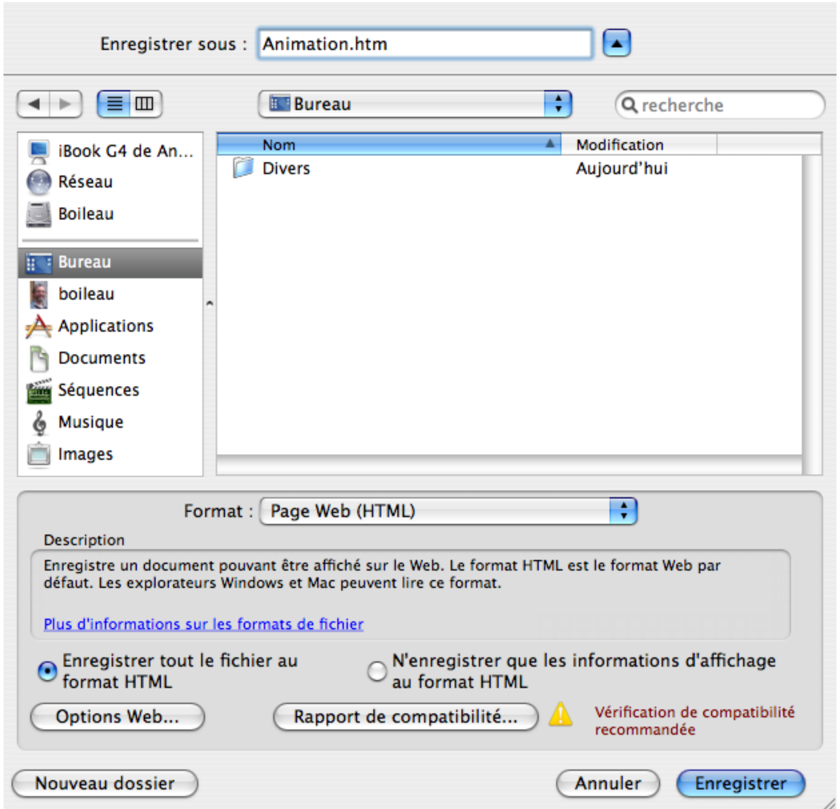


Figure 2.7.9 Sauvegarde d'un document *Word* au format « Page Web » (*Macintosh*).

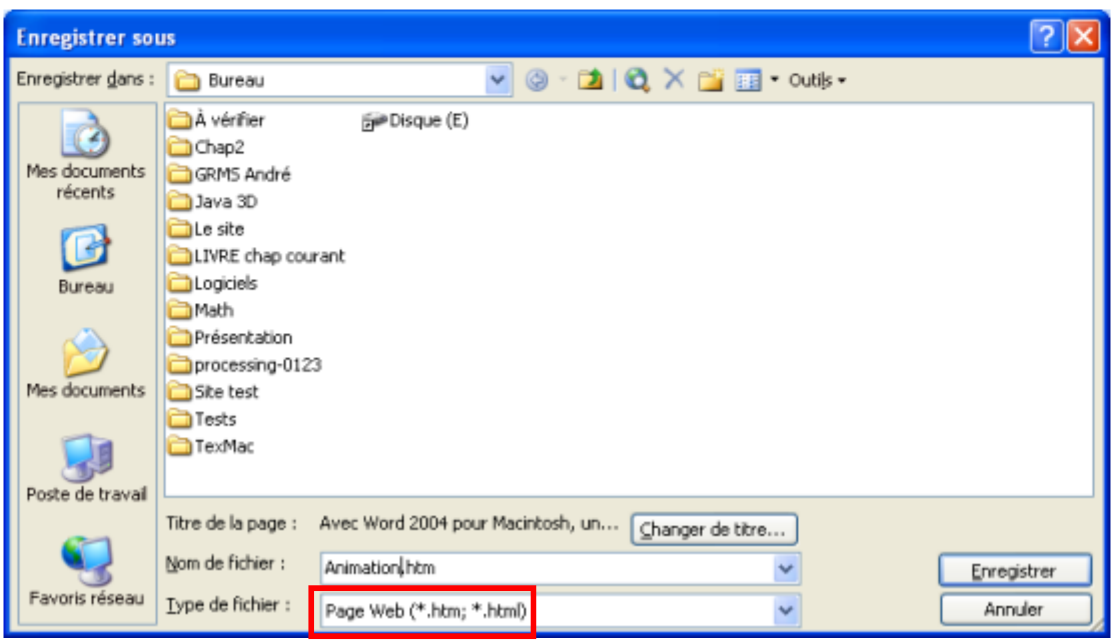


Figure 2.7.10 Sauvegarde d'un document *Word* au format « Page Web » (*Windows*).

.

(4) Fabrication d'une animation QuickTime

Nous allons maintenant utiliser le logiciel *QuickTime Player*[1] pour rassembler en un film toutes nos images. Après avoir ouvert ce logiciel, nous sélectionnons l'item « Ouvrir une séquence d'images... » du menu « Fichiers » et nous choisissons l'une des images du dossier *Animation_fichiers* que nous avons obtenues à l'étape précédente. Après avoir décidé de la vitesse de défilement des images (voir figure 2.7.11), nous obtenons une fenêtre nous permettant de voir l'animation ainsi produite. Comme le mouvement est cyclique, il peut être intéressant de voir le film en boucle (item « Boucle » du menu « Présentation »). Au moment d'enregistrer ce film, n'oubliez pas de choisir « Enregistrer comme séquence autonome » : sinon, notre film nécessiterait la présence de tous les fichiers d'images pour pouvoir être visionné.

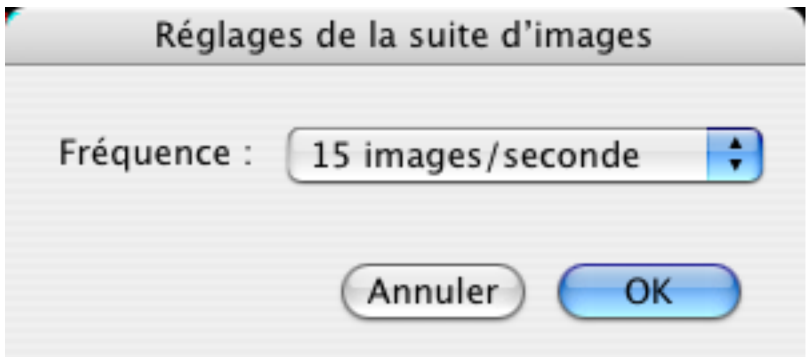
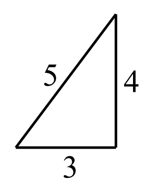


Figure 2.7.11 Ouverture d'une séquence d'images avec *QuickTime Player PRO* : on doit spécifier le nombre d'images/seconde.

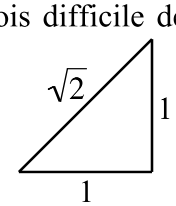
On pourrait même utiliser le logiciel *QuickTime Player PRO* pour combiner nos animations de diverses façons : bout-à-bout, côte-à-côte, ou même l'une dans l'autre. Mais nous terminons ici notre brève visite dans le monde de l'animation.

2.8 Ajout d'étiquettes pour annoter nos graphiques



Jusqu'à présent, nous n'avons pas mentionné l'icône « Ajouter une étiquette »  de la barre d'outils « Progiciels », qui nous aidera à ajouter des annotations à nos graphiques, comme illustré dans l'exemple ci-contre.

Disons tout d'abord que *LangageGraphique* dispose de la commande *Texte* pour insérer une chaîne de caractères dans un graphique. Mais il faut avouer que, si le placement des diverses composantes géométriques d'une figure requiert souvent une précision toute mathématique, l'ajout d'annotations devrait plutôt obéir à des critères esthétiques qu'il est parfois difficile de saisir textuellement. D'où la présence du bouton « Ajouter une étiquette », qui ajoute au document *Word* courant une zone de texte transparente qu'on peut remplir, formater et déplacer à la main. Un avantage de cette façon de faire est



[1] Rappelons qu'il faut utiliser la version PRO (qui est payante) pour créer nos films. Par la suite, on pourra lire ceux-ci en utilisant indifféremment la version PRO ou la version gratuite de *QuictTime Player*.

qu'il est possible d'incorporer des équations (voir ci-contre) dans la zone de texte ainsi créée, ce qui est impossible avec la commande *Texte*.

Un conseil sur l'utilisation du bouton  : redimentionnez votre graphique aux dimensions voulues **avant** d'ajouter vos étiquettes[1], puis groupez en un seul objet le graphique et les étiquettes ajoutées, ce qui facilitera d'éventuels déplacements de l'objet (ou de copies de celui-ci) dans votre document.

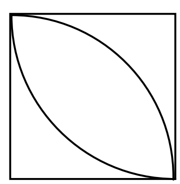
Nous sommes maintenant arrivés au terme de ce chapitre. Le lecteur intéressé par la programmation pourrait poursuivre l'étude de *LangageGraphique* pour réaliser des graphiques plus élaborés, tels ceux produits par la démonstration fournie lors de l'installation.

2.9 Exercices

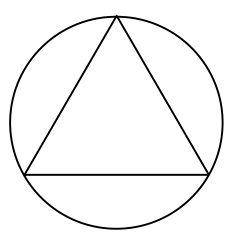
L'utilisation de *LangageGraphique* a pour but de réaliser des graphiques avec une grande précision. L'objectif des exercices qui suivent est de vous amener à utiliser *LangageGraphique*, en profitant de ses capacités de précision. En regardant un graphique avec un grossissement important, vous pourrez faire apparaître d'éventuels défauts invisibles à taille normale et ce faisant vérifier la précision du tracé.

1- *Dessiner avec LangageGraphique*
Utilisez *LangageGraphique* pour reproduire les graphiques ci-dessous. Pour c) et d) réalisez le travail de deux façons : en utilisant d'abord des commandes de géométrie analytique, puis des commandes de géométrie de la tortue.

a) Un carré et deux quarts de cercles inscrits :

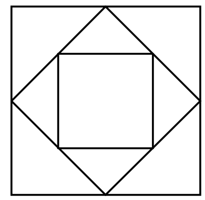


b) Un triangle équilatéral et son cercle circonscrit :

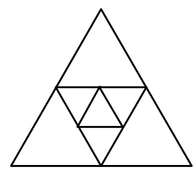


[1] Si vous vous demandez pourquoi, je vous réfère à l'exercice **2**, ou vous aurez l'occasion de constater qu'une réduction du second graphique de la section par un facteur $\frac{1}{2}$ fait carrément disparaître les étiquettes.

c) Trois carré emboîtés, le deuxième et le troisième ayant pour sommets les milieux des côtés du précédent:

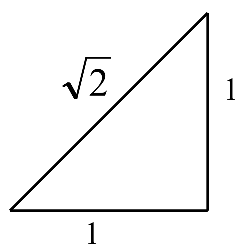


d) Trois triangles équilatéraux emboîtés, le deuxième et le troisième ayant pour sommets les milieux des côtés du précédent:



2- *Dessiner avec LangageGraphique*

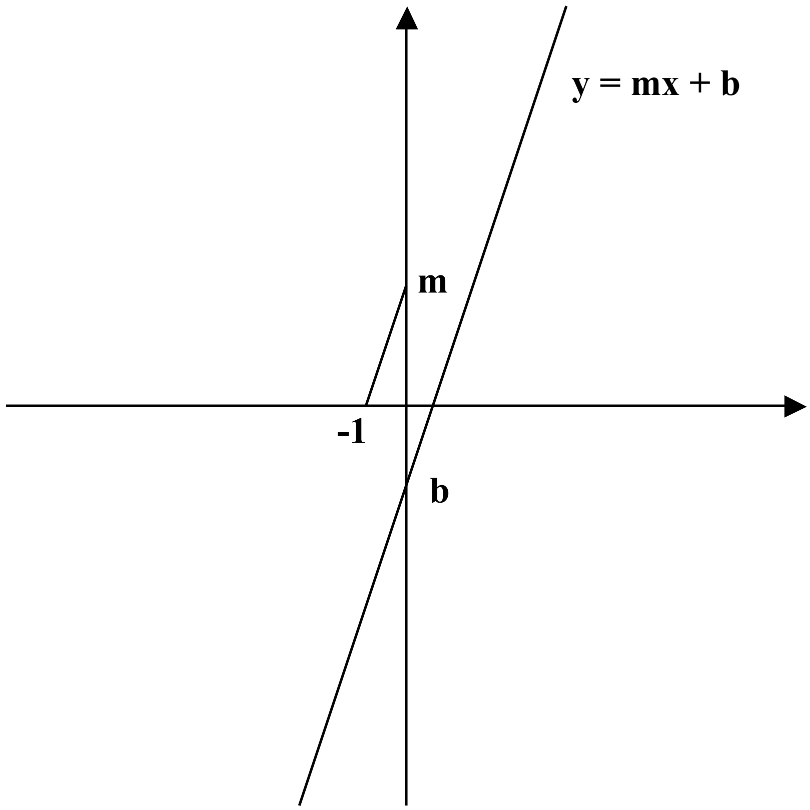
a) Reproduisez le graphique ci-dessous (avec à peu près la même taille) en utilisant LangageGraphique et ajoutez les étiquettes manuellement.



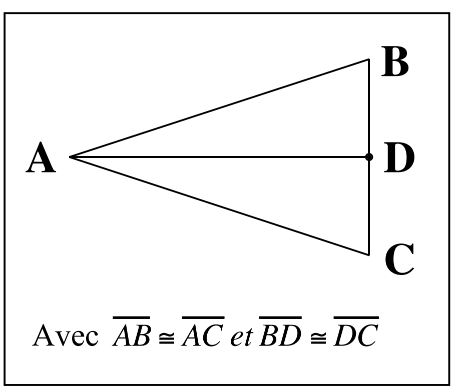
b) Groupez le graphique et les étiquettes, puis ajustez la taille à 50% (ou moins) de la taille originale.
Pouvez-vous, maintenant, dire pourquoi on vous conseille de placer les étiquettes **après** avoir donné au graphique la taille désirée?

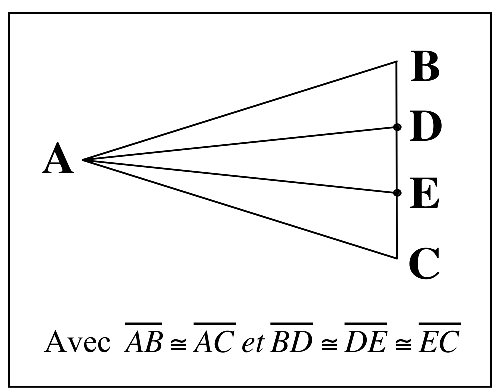
3- *Dessiner avec LangageGraphique*
Réalisez, avec *LangageGraphique*, les graphiques des items a) et b) de l'exercice 6 du chapitre 1. Vous pourrez ajouter les flèches et les étiquettes manuellement.

a)

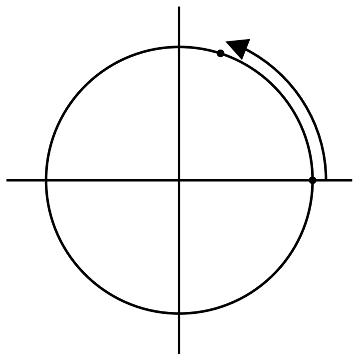
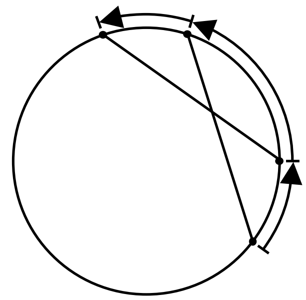


b)

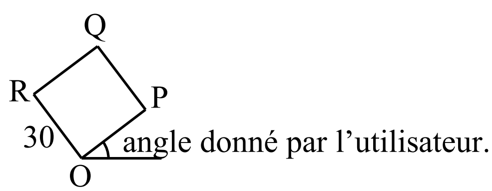




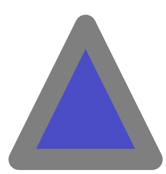
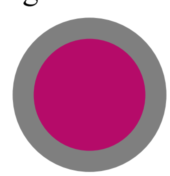
4- *Dessiner avec LangageGraphique*
Utilisez *LangageGraphique* pour réaliser les graphiques qui ont été utilisés dans le projet 2 du chapitre 1 :

5- *Dessiner avec LangageGraphique*
Modifiez l'exemple de la page 31 (géométrie analytique et géométrie de la tortue), pour que la procédure *Graphique* demande à l'utilisateur l'angle de rotation à utiliser (voir section 2.6).



6- *Dessiner avec LangageGraphique*
Modifiez l'exemple de la section 2.4 pour tracer un triangle et un cercle.

7- *Tracer des graphes de fonctions avec LangageGraphique*
Modifiez l'exemple de la page 36 pour tracer, sur un même graphique, la parabole $y = ax^2$ pour différentes valeurs de a (par exemple a = -2, -1, 0, 1, 2). Variez les couleurs des graphes pour les distinguer facilement les uns des autres.

8- *Utiliser l'instruction For…Next*

a) Reproduisez l'exemple de la section 2.6 en remplaçant les cercles par des disques de couleur pour former l'image d'une cible (les couleurs peuvent très bien être choisies au hasard).
b) Utilisez l'instruction *For…Next* pour tracer une série de rectangles emboîtés ayant tous le même centre. Colorez l'intérieur des rectangles.

9- *Tracer des polygones*

a) Faites tracer un polygone à 7 côtés sans utiliser l'instruction *For…Next* :
   1. par la méthode de la géométrie de la tortue
   2. par la méthode de la géométrie analytique

b) Faites tracer un polygone à 7 côtés en utilisant l'instruction *For…Next* :
   1. par la méthode de la géométrie de la tortue
   2. par la méthode de la géométrie analytique

c) Généralisez b) pour que la procédure *Graphique* trace un polygone à *n* côtés où *n* est demandé à l'utilisateur :
   1. par la méthode de la géométrie de la tortue
   2. par la méthode de la géométrie analytique

10- *Réaliser un papier quadrillé*

Utilisez la procédure « grille » de *LangageGraphique* pour remplir la page (en laissant des marges) avec une grille composée de la façon suivante :

- des lignes horizontales et verticales de couleur foncée et espacées de 1 cm
- entre les lignes foncées, des lignes horizontales et verticales de couleur claire et espacées de 2 mm.

2.10 Projets

1- *Composer un texte mathématique avec Word, son éditeur d'équation et Langage graphique*

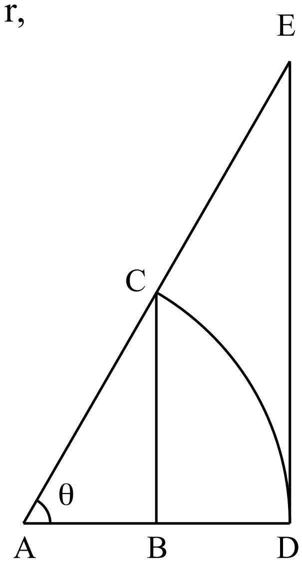
Reproduisez l'encadré suivant en utilisant *LangageGraphique* pour réaliser le graphique. Vous ajouterez les étiquettes manuellement.

Considérons un arc de cercle sur un cercle de rayon r, comme sur la figure ci-contre.

On obtient alors :

$$\underbrace{\text{aire du triangle ABC}}_{\frac{1}{2}r^2\cos\theta\,\sin\theta} < \underbrace{\text{aire du secteur ADC}}_{\frac{1}{2}r^2\theta}$$

$$< \underbrace{\text{aire du triangle ADE}}_{\frac{1}{2}r^2\,\tan\theta}$$



Au terme de quelques manipulations algébriques, et après un passage à la limite, on obtiendra :

$$\lim_{\theta\to 0}\frac{\sin\theta}{\theta} = 1.$$

2- *Composer un texte mathématique avec Word, son éditeur d'équation et Langage graphique*
Reproduisez l'encadré suivant en utilisant *LangageGraphique* pour réaliser le graphique. Vous ajouterez les étiquettes manuellement.

Dans un projet du chapitre précédent, nous avons vu que

$$\cos(\alpha+\beta) = \cos\alpha\,\cos\beta - \sin\alpha\,\sin\beta\,.$$

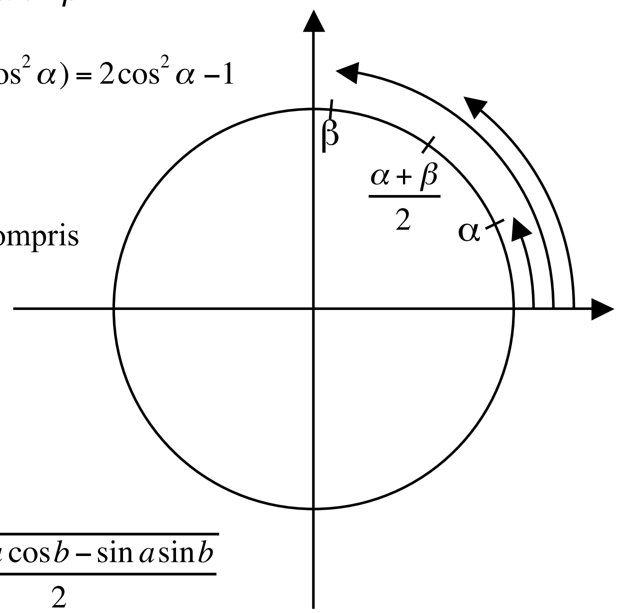
Si, de plus, on choisit $\alpha = \beta$, on obtient

$$\cos(2\alpha) = \cos^2\alpha - \sin^2\alpha = \cos^2\alpha - (1-\cos^2\alpha) = 2\cos^2\alpha - 1$$

En résolvant pour cos α, on obtient

$$\cos\alpha = \pm\sqrt{\frac{1+\cos(2\alpha)}{2}}$$

où l'on doit choisir le signe + quand α est compris entre 0 et 90 degrés.



Posons maintenant γ = 2α. On obtient alors

$$\cos\left(\frac{\gamma}{2}\right) = \sqrt{\frac{1+\cos\gamma}{2}}$$

De plus, si $\gamma = a + b$, on obtient finalement :

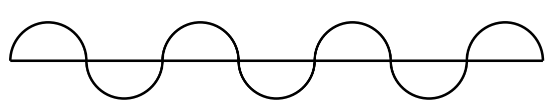
$$\cos\left(\frac{a+b}{2}\right) = \sqrt{\frac{1+\cos(a+b)}{2}} = \sqrt{\frac{1+\cos a\cos b - \sin a \sin b}{2}}$$

$$= \sqrt{\frac{1+\cos a\cos b - \sqrt{1-\cos^2 a}\sqrt{1-\cos^2 b}}{2}}$$

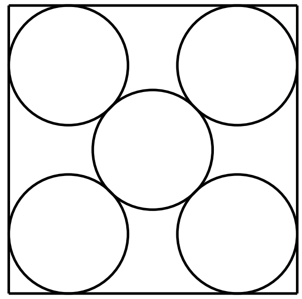
(Pour être rigoureux, il faut noter que nous avons pu utiliser l'identité $\sin x = \sqrt{1-\cos^2 x}$ parce que les angles *x* en présence sont tous supposés valoir entre 0 et 90 degrés, et que nous savons que la fonction sinus est positive dans cet intervalle.)

Cette formule nous permet donc de calculer le cosinus de la moyenne de deux angles quand on connaît le cosinus de chacun de ces deux angles.

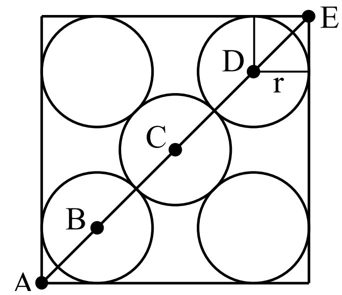
3- *Réaliser un graphique complexe avec LangageGraphique*
Utilisez *LangageGraphique* pour réaliser des graphiques du type suivant, où le nombre de demi-cercles sera donné par l'utilisateur :



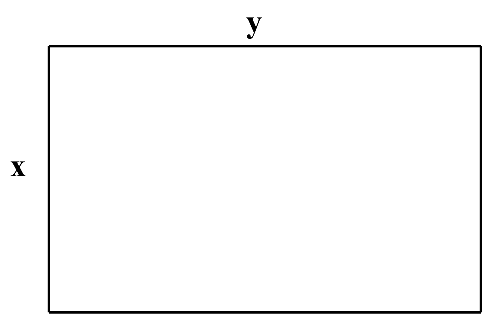
4- *Réaliser un graphique complexe avec LangageGraphique*
Utilisez *LangageGraphique* pour réaliser le graphique ci-dessous, où la longueur du côté du carré est donnée par l'utilisateur.



Les cercles doivent être tangents entre eux et, à l'exception de celui du centre, tangents à deux côtés du carré.
Toute la difficulté réside dans le calcul du rayon *r* en fonction du côté du carré. Le graphique ci-contre devrait vous aider :

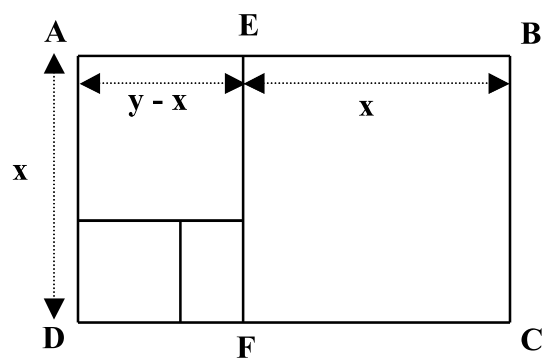


5- *Réaliser un graphique complexe avec LangageGraphique*
On appelle *nombre d'or* le nombre $\frac{1+\sqrt{5}}{2}$ et on appelle *rectangle d'or* tout rectangle dont le rapport des côtés est égal au nombre d'or.

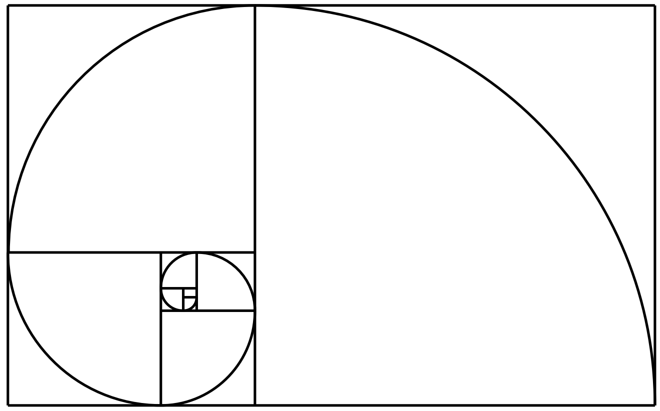


$$\frac{y}{x} = \frac{1+\sqrt{5}}{2}$$

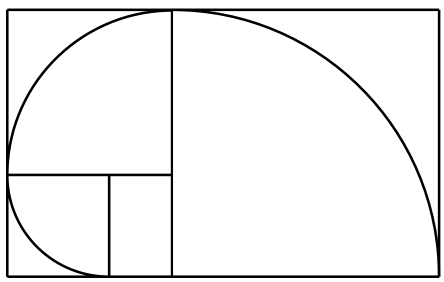
Considérons ABCD un rectangle d'or et formons le carré *EBCF*, le rectangle AEFD est alors, lui aussi, un rectangle d'or. (Pourquoi ?). Si on répète le processus avec le rectangle AEFD, on obtient un rectangle d'or encore plus petit, et, au moins théoriquement, on peut continuer indéfiniment.



Si dans les carrés obtenus on inscrit des quarts de cercles comme sur la figure ci-dessous, on obtient ce que l'on appelle la *spirale d'or*.



Utilisez *LangageGraphique* pour reproduire le graphique suivant, qui illustre les trois premières étapes de la construction d'une spirale d'or :



6- *Tracer des graphes de fonctions avec LangageGraphique*
Inspirez vous de l'exemple page 37 pour tracer, dans un même graphique, le graphe du cercle d'équation $x^2 + y^2 = 1$ dans différentes régions du plan de la forme $[-a,a]\times[-b,b]$.
Commentez les résultats obtenus en fonction des différentes valeurs données à *a* et *b*.

7- *Réaliser une animation*
Réalisez l'animation de la section 2.7 en ajoutant un triangle rectangle ayant pour hypoténuse le rayon qui tourne et dont un côté de l'angle droit est horizontal et l'autre vertical. Colorez le triangle avec une couleur différente de celle du disque.

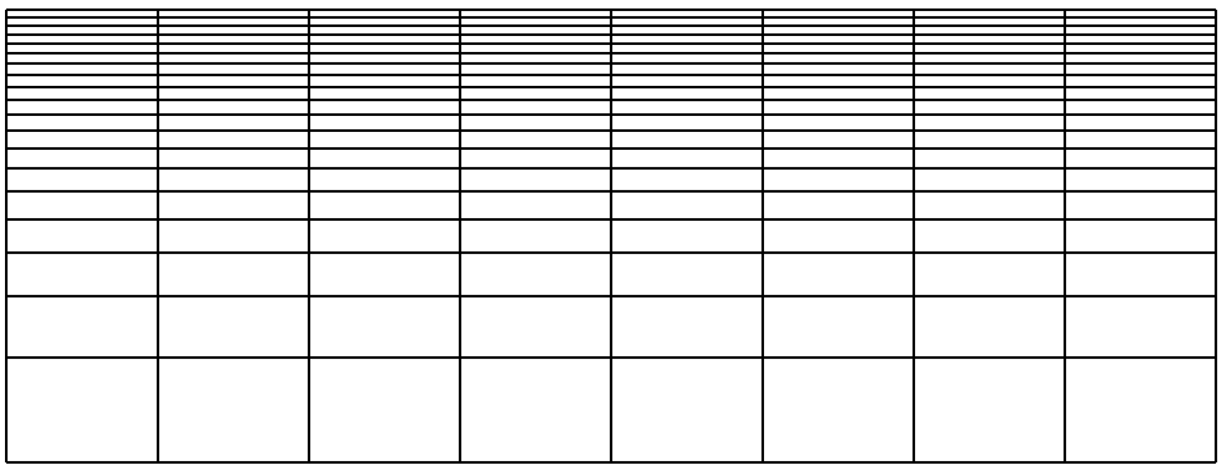
8- *Réaliser des animations*
Réalisez des animations d'un triangle équilatéral qui tourne autour d'un de ses sommets.

a) En utilisant la géométrie analytique.
b) En utilisant la géométrie de la tortue.

9- *Réaliser des papiers quadrillés*

a) Sans utiliser la procédure « grille » de *LangageGraphique,* remplissez la page (en laissant des marges) avec une grille composée de la façon suivante :
   - des lignes horizontales et verticales de couleur foncée et espacées de 1 cm
   - entre les lignes foncées, des lignes horizontales et verticales de couleur claire et espacées de 2 mm.

b) Réalisez une grille « semi-logarithmique », c'est à dire une grille où les lignes verticales sont régulièrement espacées mais ou les lignes horizontales sont espacées comme log(1), log(2), log(3), etc.



Chapitre 3

Tableur et calculs numériques

Le tableur (aussi appelé chiffrier électronique) est un outil très intéressant lorsque vient le moment de faire des mathématiques ou de les enseigner. Il permet de réaliser des calculs numériques tout en conservant les formules qui sont à la base de ces calculs. Il permet aussi de représenter graphiquement, sous de multiples formes, des données numériques obtenues par l'expérience ou le calcul. Il permet enfin l'ajout d'éléments d'interface (boutons, barres de défilement, etc.) susceptibles de rendre plus dynamiques nos représentations numériques ou graphiques. Sans compter, comme nous le verrons plus tard, que le tableur permet par surcroît plusieurs formes de programmation.

Dans ce qui suit, nous travaillerons avec *Excel*, en version française[1], car c'est le tableur le plus répandu dans les écoles québécoises. Notre but ne sera pas de décrire toutes les possibilités d'*Excel*. Nous vous proposerons plutôt quelques visites guidées qui nous semblent pertinentes pour qui s'intéresse aux mathématiques ou à leur enseignement. En dépit de son apprentissage un peu lourd au départ, nous pensons que le tableur peut s'avérer un outil pédagogique précieux dans l'enseignement des mathématiques, et qu'il est sous-utilisé à l'heure actuelle.

Décrivons donc à grands traits le contenu de ce chapitre. Après une brève description générale, nous apprendrons à utiliser *Excel* au moyen d'une suite d'exemples pertinents pour un professeur de mathématiques à l'ordre secondaire, et qui ont en commun d'exhiber les possibilités de calcul numérique des tableurs : registre de notes, remboursements hypothécaires, calcul de racines carrées, et calcul de zéros d'une fonction via la méthode de bissection. Chemin faisant, nous aborderons plusieurs sujets importants dans le contexte des tableurs : description des diverses composantes, styles d'utilisation des références (absolues, relatives et mixtes), désignation des divers éléments d'une feuille de calcul, emploi d'instructions conditionnelles, etc. Les possibilités graphiques et dynamiques des tableurs seront abordées au chapitre suivant.

3.1 Description générale

Les tableurs agissent sur des documents appelés *classeurs*. Ceux-ci sont constitués d'une ou de plusieurs *feuilles de calcul*, elles-mêmes formées de plusieurs *cellules* disposées en configuration rectangulaire. Chaque cellule peut comporter plusieurs éléments : sa *valeur* (nombre, texte, date, etc.), une *formule* permettant de calculer sa valeur, des indications de *formatage*, etc. Ces divers éléments sont illustrés dans les figures 3.1.1 et 3.1.2 .

---

[1] Avec pour conséquence l'utilisation de virgules décimales (et non de points), et l'emploi de points-virgules pour séparer les arguments des fonctions.

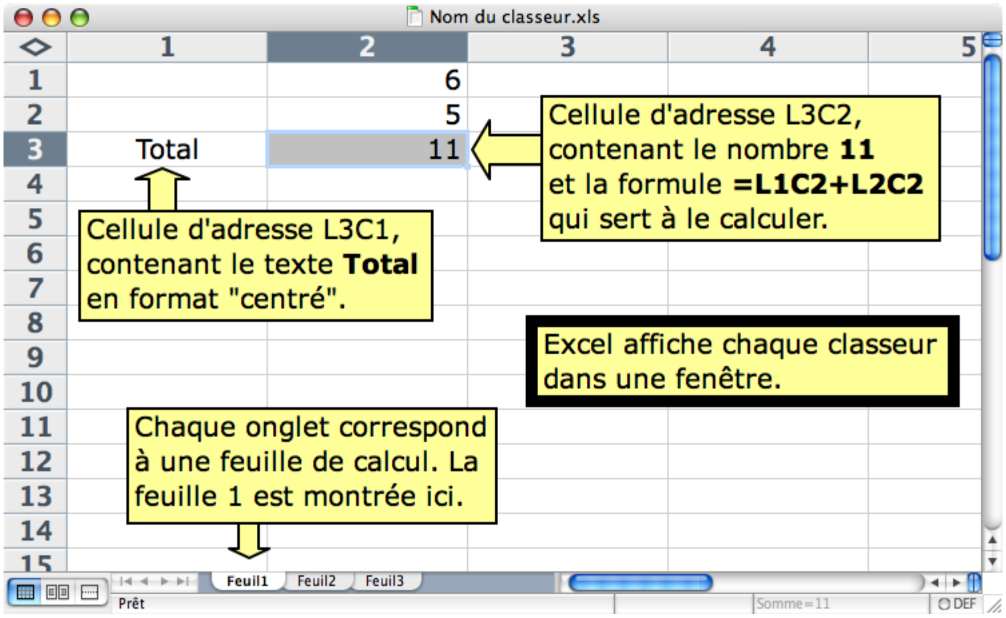


Figure 3.1.1 Fenêtre correspondant à un classeur dans Excel 2004 (version Macintosh). Notez que la cellule L3C2 (ligne 3, colonne 2) est sélectionnée.

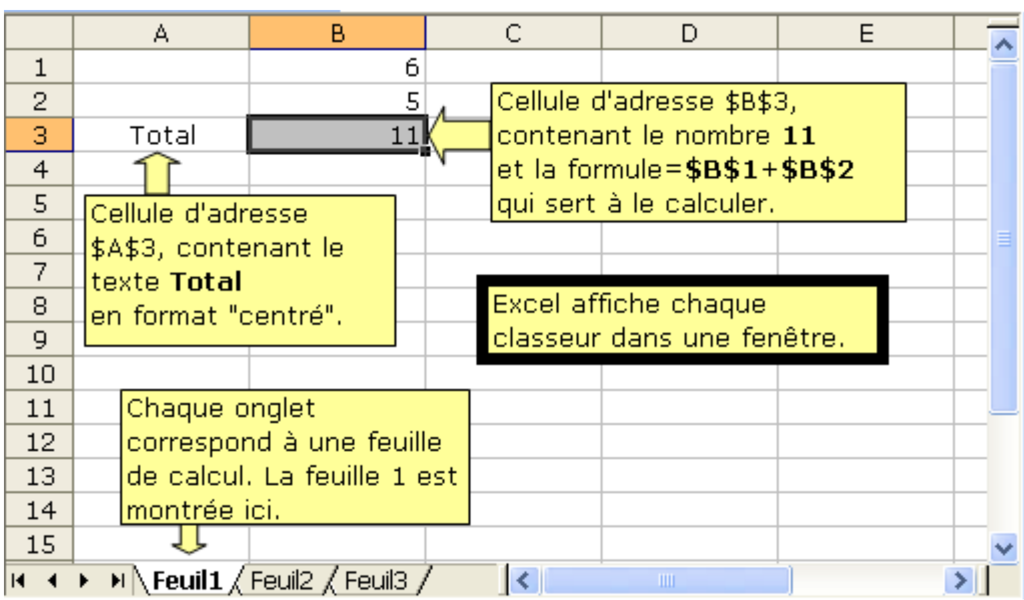


Figure 3.1.2 Fenêtre correspondant à un classeur dans Excel 2003 (version Windows). Notez que la cellule B3 (ligne 3, colonne B) est sélectionnée.

Chaque cellule peut être identifiée par son *adresse*, et notre tableur nous permet de choisir entre deux styles de référence, désignés respectivement par **L1C1** et **A1** et illustrés dans les figures 3.1.1 et 3.1.2 . Nous sommes d'avis que la notation **L1C1** comporte des avantages pédagogiques pour les débutants, et c'est pourquoi nous l'utiliserons systématiquement dans ce livre.

Comment configurer *Excel* pour utiliser le style de référence L1C1

Excel 2004 pour Macintosh : à partir du menu « **Excel** », faire successivement les choix suivants : « Préférences... » ➤ « Général » ➤ cocher « Style de référence L1C1 ».

Excel 2003 pour Windows : à partir du menu « **Outils** », faire successivement les choix suivants : « Options... » ➤ « Général » ➤ cocher « Style de référence L1C1 ».

Excel 2007 pour Windows : cliquer le bouton « **Office** » ➤ cliquer le bouton « Options Excel » et choisir successivement « Formules » ➤ « Style de référence L1C1 » (dans le groupe « Manipulation de formules »).

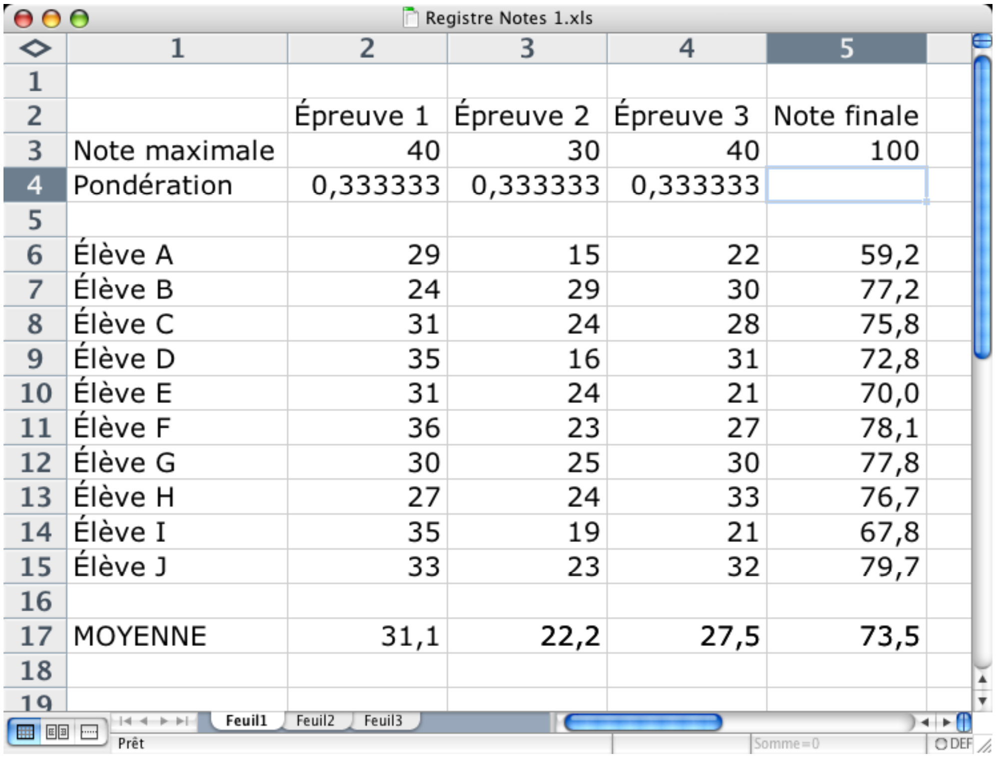
Pourquoi ce choix du style L1C1?

Nous ne mentionnerons ici que la raison la plus déterminante. Dans un tableur, il arrive très fréquemment qu'un calcul (décrit par une formule) doive être répété à de nombreuses reprises. Pour ce faire, il suffit en général de recopier plusieurs fois cette formule. Quand la formule en question est écrite en utilisant le style **L1C1**, sa signification ne dépend pas de l'endroit où elle est recopiée : elle est donc recopiée telle qu'elle, sans devoir subir de changements. Quand elle est écrite en utilisant le style **A1**, sa signification dépend parfois de l'endroit où elle est recopiée : en fait c'est le cas quand elle utilise des références relatives (voir la section 3.7). Pour que sa signification soit préservée, Excel doit en faire une version différente pour chaque cellule dans laquelle elle est recopiée, ce qui n'est pas sans poser de problèmes d'interprétation pour les débutants.

3.2 Un premier exemple : un registre de notes

Pour se familiariser avec notre tableur, nous allons créer une feuille de calcul pouvant servir de registre de notes. Outre le fait qu'une telle feuille puisse avoir une utilité pratique pour un professeur, cet exemple simple permettra d'introduire plusieurs concepts de base des tableurs : comment référer aux cellules de façon à pouvoir faire des calculs arithmétiques simples, comment formuler ces calculs de façon à pouvoir les réutiliser (via une recopie), et comment nommer divers éléments d'une feuille de calcul pour rendre le tout plus clair.

Pour débuter, nous allons considérer la situation où la note finale doit être calculée à partir de trois épreuves de poids égaux. Puis nous généraliserons afin d'obtenir une feuille de calcul plus utile. Voyons à quoi pourrait ressembler notre première version (figure 3.2.1).



Registre Notes 1.xls

	1	2	3	4	5
1					
2		Épreuve 1	Épreuve 2	Épreuve 3	Note finale
3	Note maximale	40	30	40	100
4	Pondération	0,333333	0,333333	0,333333	
5					
6	Élève A	29	15	22	59,2
7	Élève B	24	29	30	77,2
8	Élève C	31	24	28	75,8
9	Élève D	35	16	31	72,8
10	Élève E	31	24	21	70,0
11	Élève F	36	23	27	78,1
12	Élève G	30	25	30	77,8
13	Élève H	27	24	33	76,7
14	Élève I	35	19	21	67,8
15	Élève J	33	23	32	79,7
16					
17	MOYENNE	31,1	22,2	27,5	73,5
18					
19					

Figure 3.2.1 Première version de notre registre de notes.

Quand on veut placer un nombre, un texte ou une formule dans une cellule, il suffit de sélectionner celle-ci, en faisant un clic dans la cellule en question, puis de taper directement ce qu'on veut y placer. Parfois, *Excel* va interpréter autrement ce que vous venez de taper : par exemple, si vous tapez « 1/3 », *Excel* pourrait transformer ceci en « 01-mars ». Vous pouvez forcer *Excel* à vous comprendre en spécifiant le format de la cellule (Nombre, Texte, Date, Heure, Pourcentage, etc.) *avant* de taper son contenu. Par exemple, si vous spécifiez le format « Texte » et vous tapez « 1/3 » par la suite, *Excel* conservera ceci sous la forme d'un texte. Par contre, si vous spécifiez le format « Nombre» (avec deux décimales) et vous tapez « 1/3 » par la suite, *Excel* affichera alors « 0,33 ». On verra au paragraphe suivant que, dans ce cas précis, une approche alternative, utilisant une formule plutôt qu'un format, est préférable.

Comment spécifier le format d'une cellule?

Excel 2004 pour Macintosh et ***Excel 2003 pour Windows*** : à partir du menu « Format » faire successivement les choix suivants : « Cellule… » ➤ « Nombre » ➤ sélectionner le format désiré (« Texte » par exemple).

Excel 2007 pour Windows : à partir de l'onglet « **Accueil** » faire successivement les choix suivants : « Format » (dans le groupe « Cellules ») ➤ « Format de cellule » ➤ sélectionner le format désiré (« Texte » par exemple).

Dans tous les cas, on peut faire un clic-droit sur la cellule et choisir « Format de cellule... » dans le menu contextuel.

On peut, de cette façon, entrer tous les textes et tous les nombres non calculés de notre exemple. Il ne restera plus qu'à entrer les formules permettant de calculer les notes finales et les moyennes. Notez qu'il vaut mieux spécifier les pondérations par la formule « =1/3 » que par le nombre « 0,333333 », et ce pour au moins deux raisons : d'une part parce que notre feuille de calcul garde ainsi trace de la provenance de ce nombre, et d'autre part parce que le calcul est alors fait avec toute la précision dont *Excel* est capable (près d'une vingtaine de décimales, même si seulement six sont affichées).

Voyons maintenant comment calculer la note finale de l'éleve A. On obtient la contribution d'une épreuve à sa note finale en divisant la note obtenue à cette épreuve par la note maximale possible (ce qui ramène sa note sur un), puis en multipliant par 100 fois la pondération de cette épreuve. Il suffit par la suite de sommer la contribution de toutes les épreuves pour obtenir la note finale. Dans le cas présent, ceci peut se faire de plusieurs façons, dont les deux suivantes :

= (29/40)*(100/3) + (15/30)*(100/3) + (22/40)*(100/3)

= (L6C2/L3C2)*(100*L4C2) + (L6C3/L3C3)*(100*L4C3) + (L6C4/L3C4)*(100*L4C4)

Soulignons au passage qu'on indique à *Excel* qu'on désire entrer une formule en débutant celle-ci par le signe « = ».

La première des formules ci-dessus est la plus limitée : si jamais on changeait l'une des notes de l'élève A, il faudrait modifier la formule pour que le calcul reste correct. La seconde formule est plus intéressante à cet égard : si on modifie une ou plusieurs des notes, la formule effectue automatiquement[1] un nouveau calcul, ce qui fait que la note affichée restera correcte. Cette possibilité de re-calcul automatique en cas de changement de la valeur d'une ou de plusieurs cellules est une des caractéristiques les plus fondamentales et les plus intéressantes des tableurs.

Comment placer Excel en mode « Calcul automatique »

Excel 2004 pour Macintosh : à partir du menu « **Excel** », faire successivement les choix suivants : « Préférences... » ➤ « Calcul » ➤ sélectionner « Mode de calcul : Automatique ».
Excel 2003 pour Windows : à partir du menu « **Outils** », faire successivement les choix suivants : « Options... » ➤ « Calcul » ➤ sélectionner « Automatique ».
Excel 2007 pour Windows : à partir de l'onglet « **Formules** », faire successivement les choix suivants : « Options de calcul » (dans le groupe « Calcul ») ➤ sélectionner « Automatique ».

Le problème avec cette seconde formule, c'est qu'elle n'est valable que pour l'élève A, alors qu'on voudrait trouver une formule unique qui pourrait servir pour chaque élève. Pour réaliser ceci, nous devons introduire un nouveau concept, d'une importance capitale dans les tableurs : l'identification des cellules au moyen de références (ou d'adresses) **relatives**.

Dans notre contexte, l'idée de base est la suivante : pour désigner, par exemple, le résultat de la première épreuve, on utilisera non pas le fait que celui-ci se retrouve à l'intersection de la ligne 6 et de la colonne 2 (ce qui est spécifique à l'élève A), mais plutôt le fait qu'il se retrouve à la même ligne mais 3 colonnes plus à gauche (par rapport à la colonne 5 où on veut calculer la note finale), ce qui fonctionne alors pour tous les élèves. Notre tableur utilise la notation L(0)C(-3), ou

[1] Si Excel est placé en mode de calcul automatique. Si ce n'est pas le cas voir l'encadré.

plus simplement LC(-3) pour désigner une cellule se trouvant à la même ligne mais 3 colonnes à gauche de la cellule où la référence en question apparaît. De la même façon, on pourra utiliser LC(-2) pour désigner la note obtenue à la seconde épreuve, et LC(-1) pour désigner celle obtenue à la dernière épreuve. On peut alors utiliser ces références relatives aux cellules contenant les notes obtenues dans les trois épreuves pour obtenir la formule suivante, valable pour calculer la note finale de tous les élèves (pourvu qu'elle soit placée à la ligne correspondante de la colonne 5) :

= (LC(-3)/L3C2)*(100*L4C2) + (LC(-2)/L3C3)*(100*L4C3) + (LC(-1)/L3C4)*(100*L4C4)

Références (adresses) absolues, relatives, mixtes

De façon générale, si *m* et *n* sont deux entiers (positifs ou négatifs), alors

- L*m*C*n* désigne la cellule à l'intersection de la ligne *m* et de la colonne *n* Notez que ceci n'a de sens que lorsque *m* et *n* sont tous deux strictement positifs.
- L(*m*)C(*n*) désigne la cellule située |*m*| lignes en dessous (ou au dessus, si *m*<0) et |*n*| colonnes à droite (ou à gauche, si *n*<0) de la cellule où la référence en question apparaît

On peut aussi combiner ces notations : par exemple

- L(*m*)C*n* désigne la cellule à la colonne *n* et |*m*| lignes en dessous (ou au dessus, si *m*<0) de la cellule où la référence en question apparaît
- L*m*C(*n*) désigne la cellule à la ligne *m* et |*n*| colonnes à droite (ou à gauche, si *n*<0) de la cellule où la référence en question apparaît

Les références de style

- L*m*C*n* sont dites absolues
- L(*m*)C(*n*) sont dites relatives
- L(*m*)C*n* ou L*m*C(*n*) sont dites mixtes.

Note : on peut omettre « (0) » dans toute référence.

Notons ici qu'on ne doit surtout pas rendre relatives les références aux notes maximales et aux pondérations des épreuves. (Pourquoi?)

En utilisant des références mixtes, on peut obtenir une formule calculant la note finale de tous les élèves, peu importe la colonne de la cellule où elle est placée, pourvu qu'elle soit placée sur la même ligne que l'élève visé :

= (LC2/L3C2)*(100*L4C2) + (LC3/L3C3)*(100*L4C3) + (LC4/L3C4)*(100*L4C4)

Notez ici que nous avons utilisé l'écriture LC*n* plutôt que L(0)C*n* : de toutes façons, si nous avions utilisé la seconde forme, *Excel* l'aurait transformée automatiquement en la première.

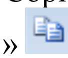
À la recherche d'efficacité (peu payante, dans le cas présent)

Remarquons en passant que nous pouvons remplacer la formule ci-dessus par une formule mathématiquement équivalente, mais informatiquement plus économe (puisqu'elle comporte deux multiplications de moins) :

= 100*((LC2/L3C2)*L4C2 + (LC3/L3C3)* L4C3 + (LC4/L3C4)* L4C4)

Même pour une classe nombreuse de *k* élèves (2*k* multiplications de moins), la différence de vitesse dans les calculs du tableur ne sera pas perceptible. On verra plus tard des cas où une attention portée à l'efficacité informatique sera plus payante.

On peut donc taper cette formule à la cellule L6C5 pour ensuite la recopier dans les cellules associées aux autres élèves. On peut réaliser ceci comme suit :

- On sélectionne (par un clic souris) la cellule qu'on veut recopier.
- Par un choix dans le menu « Édition » ou via un équivalent clavier, on demande un « Copier ». (Dans *Excel 2007*, on clique plutôt sur l'onglet « Accueil », puis sur l'icône « Copier » .)
- On sélectionne (par un glisser souris) une plage (bloc rectangulaire) de cellules.
- Par un choix dans le menu « Édition » ou via un équivalent clavier, on commande un « Coller ». (Dans *Excel 2007*, on clique plutôt sur l'onglet « Accueil », puis sur le bouton « Coller »).

On peut aussi utiliser une méthode alternative si on veut recopier vers le bas ou vers la droite :

- On sélectionne (par un clic souris) la cellule qu'on veut recopier.
- On approche le curseur-souris du coin inférieur droit de ladite cellule jusqu'à ce que le pointeur prenne la forme d'une croix.
- On fait alors un « glisser » vers le bas ou vers la droite jusqu'à la dernière cellule où le « Coller » doit être effectué.

Reste maintenant à calculer les moyennes. Pour ce faire, on utilisera la fonction « MOYENNE» d'*Excel*, qui peut s'appliquer à une plage de cellules. Pour désigner une telle plage de cellules, on utilisera la notation suivante : on désigne d'abord la cellule en haut à gauche de la plage (par une référence absolue, relative ou mixte), on fait suivre du caractère « : », puis on termine par une référence à la cellule en bas à droite de la plage. Dans notre cas, la plage pourra être désignée par L(-11)C:L(-2)C ou par L6C:L15C, et la formule recherchée sera l'une des formules suivantes :

=MOYENNE(L(-11)C:L(-2)C)
=MOYENNE(L6C:L15C)

Entrée automatique d'une référence dans une formule

Pour obtenir le -11 de la référence relative ci-dessus, on a dû soit compter le nombre de sauts pour passer de L6C à L17C, soit faire une soustraction (6-17).
(Pour les curieux : l'écriture de références relatives est l'une des seules tâches où la notation A1 s'avère plus commode que la notation L1C1.)

Mais il existe un truc bien utile pour nous faciliter la tâche. Au moment de spécifier une référence dans une formule, on peut utiliser la souris pour désigner la cellule (par un clic) ou la plage de cellules (par un glisser) : *Excel* se chargera d'insérer la référence relative correspondante dans la formule. Si on désire un autre type de référence (absolue ou mixte), la touche commande-T (Mac) ou F4 (Windows) changera cycliquement le type de référence.

3.3 Nommer certaines composantes dans un tableur

Nous venons de voir comment faire référence aux cellules ou plages de cellules d'un tableur. Il faut cependant avouer que la notation utilisée (L3C2, par exemple) n'évoque pas la signification de la valeur stockée à cet endroit (« la note maximale possible pour l'épreuve 1 », dans le cas de L3C2). Heureusement, les tableurs nous permettent de donner des noms (par exemple « Max1 » pour L3C2) qui renseigneront mieux les humains amenés à lire et à interpréter les formules d'un tableur. Après tout, dans un contexte pédagogique, les diverses indications informatiques sont souvent faites pour être lues autant par des humains que par des machines.

Nous allons donc nommer certains éléments de notre feuille de calcul. Après quoi, la formule permettant de calculer la note finale pourra s'écrire

=100*((Note1/Max1)*Poids1+(Note2/Max2)*Poids2+(Note3/Max3)*Poids3)

ce qui a le mérite d'être plus clair (pour un humain) que

= 100*((LC2/L3C2)*L4C2 + (LC3/L3C3)* L4C3 + (LC4/L3C4)* L4C4)

Pour nommer une composante (cellule, plage de cellules, ligne, colonne) d'un tableur, on peut procéder comme suit :

- Sélectionner (par un clic ou un glisser souris) la composante[1] en question
- Faire ensuite un clic dans la Zone Nom de la barre de formule[2]

Figure 3.3.1 La barre de formule avec « Max1 » tapé dans la Zone Nom.

- Taper le nom désiré, et confirmer par un retour de chariot (« return » ou « enter »). Comme toujours, il demeure sage d'éviter les caractères spéciaux ou accentués dans un nom.

Dans le cas de notre registre de notes, on pourrait nommer les éléments suivants :

- « Max1 », « Max2 » et « Max3 » les cellules contenant les notes maximales possibles
- « Poids1 », « Poids2 » et « Poids3 » les cellules contenant les poids respectifs des épreuves
- « Note1 », « Note2 » et « Note3 » les colonnes correspondant aux épreuves.

Après la définition de ces noms, on obtient la formule annoncée pour calculer la note finale

=100*((Note1/Max1)*Poids1+(Note2/Max2)*Poids2+(Note3/Max3)*Poids3)

Il faut souligner ici les conventions utilisées par *Excel* qui assurent le bon fonctionnement du mécanisme d'attribution des noms ci-dessus. Quand on nomme une cellule ou une plage, *Excel* utilise automatiquement des références absolues. (En fait, comment pourrait-il en être autrement, dans les circonstances, puisque ces noms réfèrent à des cellules.) Par contre, la référence à une colonne est automatiquement mixte : absolue pour la colonne et relative pour la ligne. (De même, la référence à une ligne est automatiquement mixte : absolue pour la ligne et relative pour la colonne.) Avec ces conventions, les deux formules précédentes ont donc exactement la même signification.

Si on fait une erreur dans le processus d'attribution des noms (erreur dans l'écriture du nom ou dans la désignation de la composante), nous ne pourrons corriger en recommençant comme précédemment. Il nous faudra plutôt faire apparaître la fenêtre de dialogue représentée à la figure 3.3.2 et supprimer ou corriger les noms en question.

[1] Pour sélectionner une ligne, il suffit de faire un clic sur l'étiquette située à gauche de la ligne en question. Pour sélectionner une colonne, il suffit de faire un clic sur l'étiquette située en haut de la colonne en question.

[2] Si la barre de formule n'est pas apparente, on peut la rendre visible à partir du menu « Affichage » en cochant l'item « Barre de formule ».

Pour faire apparaître la fenêtre de dialogue « Définir un nom »

Excel 2004 pour Macintosh et ***Excel 2003 pour Windows*** : à partir du menu « **Insertion**», faire successivement les choix suivants : « Nom » ➤ « Définir... ».
Excel 2007 pour Windows : à partir de l'onglet « **Formules** », dans le groupe « Noms définis » cliquer sur le bouton « Gestionnaire de noms ».

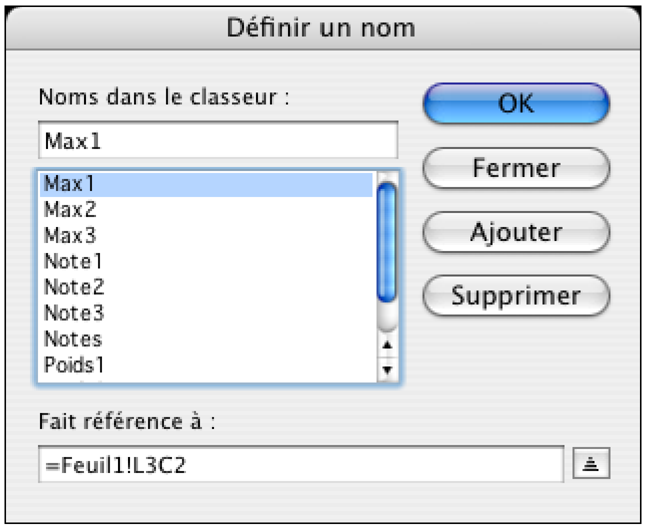


Figure 3.3.2 La fenêtre de dialogue permettant de gérer les noms.
(Illustrée dans le cas *d'Excel 2004* pour Macintosh.)

Libre à nous de supprimer un nom pour le redéfinir ensuite comme précédemment, ou de le corriger directement dans cette fenêtre. Remarquons au passage qu'*Excel* complète les références en préfixant la référence usuelle du nom de la feuille de calcul suivi du caractère « ! ».

Restrictions dans le choix des noms

Ici comme dans d'autres contextes, nous évitons d'utiliser des noms comportant des caractères spéciaux (espace, accents, ponctuation, etc.) : bien que certains environnements technologiques nous le permettraient, ce principe de précaution évite de surcharger notre mémoire avec le détail de ce qui est permis et ce qui ne l'est pas dans tel ou tel contexte.

Dans le cas précis d'*Excel*, des restrictions inhabituelles s'ajoutent. On ne doit pas choisir des noms ressemblant de trop près aux références désignant des cellules. Ainsi les noms du type L1C1, L1, C1 ou même L et C doivent être bannis. Il est sage d'éviter aussi la lettre R (pour « row ») et ses dérivés, qui joue le rôle du L (pour « ligne ») dans la version anglaise d'*Excel*. (De même, si nous voulons pouvoir passer aux références du type A1, il faut éviter des noms comme « A1 ».)

Notons au passage qu'*Excel* ne distingue pas entre majuscules et minuscules dans les noms.

Soulignons ici que cette fenêtre de dialogue nous permet d'attribuer des noms qui seraient impossibles à obtenir via la Zone Nom de la barre de formule. Par exemple, on pourrait définir le nom « Notes » comme faisant une référence mixte à une plage « Feuil1!L6C:L15C », ce qui nous permettrait d'utiliser la formule suivante pour calculer les moyennes pour chaque épreuve et pour la note finale :

=MOYENNE(Notes)

Excel nous permet aussi de représenter autrement nos données, par exemple de les trier en ordre décroissant en fonction de la note finale. Un mot d'avertissement est ici de rigueur : *Excel* nous permet tout aussi facilement de trier les notes en perdant la correspondance avec les noms des étudiants (en n'incluant pas la première colonne dans la plage à trier), avec les conséquences qu'on imagine... Il faut donc être très attentif quand vient le moment de faire des tris.

Pour trier des données

Excel 2004 pour Macintosh et ***Excel 2003 pour Windows*** : Il faut sélectionner tout d'abord la plage des données à trier (L6C1:L15C5 dans notre cas) puis, à partir du menu « Données », choisir « Trier... », puis spécifier la colonne à trier (la colonne 5 dans notre cas) et le mode de tri (« Décroissant » dans notre cas). Notons en passant que le bouton option nous permet de trier par lignes plutôt que par colonnes.

Excel 2007 pour Windows : Il faut sélectionner tout d'abord la plage des données à trier (L6C1:L15C5 dans notre cas) puis, à partir de l'onglet « Données », cliquer sur le bouton « Trier... » dans le groupe « Trier et filtrer », puis spécifier la colonne à trier (la colonne 5 dans notre cas) et le mode de tri (« Du plus grand au plus petit » dans notre cas). Notons en passant que le bouton option nous permet de trier par lignes plutôt que par colonnes, en choisissant « De la gauche vers la droite ».

Même si on ne trie pas les notes, on pourrait voir apparaître automatiquement en rouge les notes finales qui sont inférieures à la note de passage. On y arrive via une « Mise en forme conditionnelle... ». Pour plus de détails, on pourra consulter l'aide en ligne d'Excel.

Dans les exercices, vous serez appelé à étendre le registre de notes pour permettre de traiter jusqu'à dix épreuves contribuant au calcul de la note finale. Plus tard, quand nous verrons les opérations matricielles, on pourra généraliser encore plus.

L'aide en ligne dans *Excel*

Dans *Excel,* l'aide en ligne fonctionne de façon très semblable à celle de *Word*. Soulignons, ici encore, la richesse et l'intérêt de cette ressource, qui est toujours disponible.

3.4 Second exemple : calcul d'un remboursement hypothécaire

Nous allons maintenant créer une feuille de calcul nous permettant de trouver le remboursement mensuel d'une hypothèque. Il existe des formules mathématiques nous permettant de trouver directement le montant de ce remboursement, mais nous choisirons plutôt une approche plus élémentaire, nécessitant moins de connaissances mathématiques mais plus de recours à la technologie. Cela illustre un phénomène passablement général : on peut souvent compenser un manque de sophistication mathématique par un appel à la puissance de calcul des outils technologiques, et réciproquement on peut souvent compenser un manque de puissance de calcul par un appel à des mathématiques plus avancées. Et quand on dispose à la fois des deux, on arrive généralement à écrire des logiciels beaucoup plus efficaces!

Vous trouverez à la figure 3.4.1 une reproduction d’une partie de la feuille de calcul que nous allons réaliser. Elle est incomplète car elle comporte 488 lignes (pour laisser place à un prêt d’une durée maximale de 40 années, soit 480 mois). Mais il ne faut pas se laisser impressionner par les dimensions de la feuille : après avoir rempli les dix premières lignes, il suffira de recopier la ligne 10 de la ligne 11 à la 488^{e}.

Hypotheques.xls

	1	2	3	4	5	6
1						
2		Montant emprunté		40000		
3		Nombre d'années (max 40)		20		
4		Taux d'intérêt mensuel		0,003		
5		Remboursement mensuel		234,0446		
6		Reste		0,00		
7						
8	Mois	Montant initial	Intérêts courus	Montant dû	Remboursement	Montant résiduel
9	1	40000,00	120,00	40120,00	234,04	39885,96
10	2	39885,96	119,66	40005,61	234,04	39771,57
11	3	39771,57	119,31	39890,88	234,04	39656,84
12	4	39656,84	118,97	39775,81	234,04	39541,76
13	5	39541,76	118,63	39660,39	234,04	39426,35
14	6	39426,35	118,28	39544,62	234,04	39310,58
15	7	39310,58	117,93	39428,51	234,04	39194,47
16	8	39194,47	117,58	39312,05	234,04	39078,01
17	9	39078,01	117,23	39195,24	234,04	38961,20
18	10	38961,20	116,88	39078,08	234,04	38844,03
19	11	38844,03	116,53	38960,57	234,04	38726,52
20	12	38726,52	116,18	38842,70	234,04	38608,66

Feuil1 Feuil2 Feuil3

Prêt Somme=234,0446 DEF MAJ NUM

Figure 3.4.1 Feuille de calcul pour trouver le montant requis pour rembourser une hypothèque, par une méthode d’essais et d’erreurs.

Pour réaliser cette feuille de calcul, on peut d’abord remplir toutes les cellules qui contiennent du texte, puis remplir (et nommer) les cellules contenant les données initiales : montant emprunté (« Montant »), nombre d’années (« NombreAnnees »), et taux d’intérêt (« TauxInteret »). Notez que le taux d’intérêt affiché par les banques (canadiennes) est un taux d’intérêt annuel (mais qui doit être composé deux fois par an), alors que le taux d'intérêt utilisé dans cette feuille est le taux mensuel correspondant[1].

Laissons temporairement de côté les cellules L5C4 (« Paiement ») et L6C4 (« Reste »), et voyons comment remplir le tableau. Pour rendre les formules à venir plus claires, on peut commencer par nommer les colonnes 1 à 6 : « Mois », « MontantInitial », « Interets », « MontantDu », « Remboursement » et « MontantResiduel ».

Puis on décrit les 6 cellules de la ligne 9 : on définit la cellule de la colonne nommée

- « Mois » en y plaçant la valeur 1
- « MontantInitial » via la formule « =Montant »
- « Interets » via la formule « =MontantInitial * TauxInteret »
- « MontantDu » via la formule « =MontantInitial + Interets »
- « Remboursement » via la formule « =Paiement »
- « MontantResiduel » via la formule « =MontantDu - Remboursement ».

On définit ensuite de la même façon les 6 cellules de la ligne 10, aux deux exceptions suivantes :

- on définit la cellule de la colonne nommée « Mois » via la formule « =L(-1)C+1 », ce qui revient à ajouter 1 au numéro de mois de la ligne précédente
- on définit la cellule de la colonne nommée « MontantInitial » via la formule « =L(-1)C(4) », ce qui correspond au « MontantResiduel » de la ligne précédente.

Pour recopier ces 6 cellules de la ligne 10 partout en bas du tableau, on peut procéder de la façon suivante : on sélectionne la plage L10C1:L10C6 de ces six cellules, on demande (via un item de menu ou un équivalent clavier) de « Copier », on sélectionne le bas du tableau (plage L11C1:L488C6), et on demande « Coller ». On peut aussi sélectionner les 6 cellules initiales puis amener le curseur en bas à droite de la plage jusqu'à prenne la forme d'un croix : il suffit alors de glisser jusqu'à la fin du tableau.

Pour définir la cellule L6C4 (« Reste »), il faut dire à *Excel* d'y recopier le Montant résiduel qu'on retrouve dans notre tableau à la fin de la période de remboursement, ce qui correspond au contenu de la cellule de la ligne 8+12*NbAnnees et de la colonne 6. Mais comme on ne peut utiliser (directement ou indirectement) de calculs[2] dans les références L1C1, il faudra procéder autrement. On va commencer par donner le nom de « Tableau » à la plage L9C1:L488C6 », et on va placer dans la cellule « Reste » la formule « =Index(Tableau;12*NbAnnees;6) ». On utilise ici la fonction *Excel* « Index(plage;ligne;colonne) », qui retourne le contenu de la cellule située à la ligne et à la colonne de la plage, où la numérotation des lignes et des colonnes se fait relativement à la plage en question : par exemple « Index(Tableau;1;1) » réfèrera à la cellule L9C1 (puisque la première ligne de la plage est à la 9^{e} ligne de la feuille).

---

[1] Si b est le taux d'intérêt affiché par les banques et t est le taux mensuel devant être utilisé dans la feuille, on doit avoir $(1+t)^{12} = \left(1+\frac{b}{2}\right)^{2}$. Vous pourriez, en exercice, intégrer ce calcul à la feuille *Excel*.

[2] Par exemple, on ne peut pas écrire « L2+3C2 » pour désigner L5C2 ».

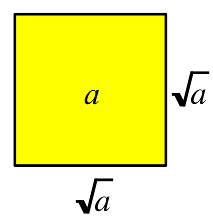
Notre feuille de calcul est maintenant terminée : c'est l'utilisateur qui entrera itérativement des valeurs dans la cellule L5C4 (« Paiement »), jusqu'à ce que la cellule L6C4 (« Reste ») du montant restant à rembourser au bout du nombre d'années spécifié soit nul : si le montant affiché en L6C4 est plus grand que zéro, c'est signe que le paiement n'est pas assez grand; si au contraire il est négatif, c'est signe qu'il est trop grand. Notez que, pour obtenir un reste affiché nul, il faut souvent utiliser plus de deux décimales, ce qui n'a pas d'équivalent monétaire exact : dans la vie courante, on pourrait obtenir le même résultat avec un dernier paiement légèrement différent.

3.5 Calcul des racines carrées

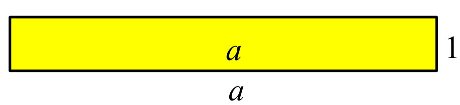
Jadis, on apprenait à l'école un algorithme[1] d'extraction des racines carrées. Bien que fastidieux, il était bien adapté à un calcul à la main. La nécessité de pratiquer de tels calculs a disparu avec l'arrivée des calculatrices et autres ordinateurs. Cependant, on peut encore se poser la question suivante : comment nos outils technologiques réalisent-ils, en général, de tels calculs, que ce soit l'extraction de racines (carrées, cubiques ou autres), le calcul des fonctions transcendantes (trigonométriques, logarithmiques ou exponentielles), ou la détermination numérique de solutions d'équations?

Il n'est peut-être pas nécessaire de connaître la méthode exacte utilisée par un outil technologique donné pour effectuer un certain calcul, et ce pour plusieurs raisons : la mathématique sous-jacente peut être, dans certains cas, hors de notre portée[2] ou de celle de nos élèves; le fabricant de l'outil en question n'a peut-être pas rendu publique la méthode qu'il utilise; ou encore nous ne voulons pas investir un temps trop grand pour éclairer ce point précis.

Dans ce qui suit, nous nous contenterons donc d'exposer une méthode à la fois simple, claire et efficace pour calculer les racines carrées; puis nous l'implanterons dans *Excel*. Notre point de départ est le suivant : calculer la racine carrée d'un nombre positif a revient à trouver la longueur du côté d'un carré d'aire a. À première vue, cela ne nous avance guère : il ne semble pas plus facile de résoudre une version du problème que l'autre.



Essayons donc de résoudre un problème plus facile : trouver les côtés d'un rectangle (au lieu d'un carré) d'aire a. Ici, au contraire, ça semble trop facile, car on peut trouver aisément une infinité de solutions : pour trouver un tel rectangle, on peut choisir tout nombre positif b comme base, et calculer ensuite la hauteur nécessaire pour que l'aire soit a. Une solution particulièrement simple s'obtient en choisissant un rectangle de base a et de hauteur 1.



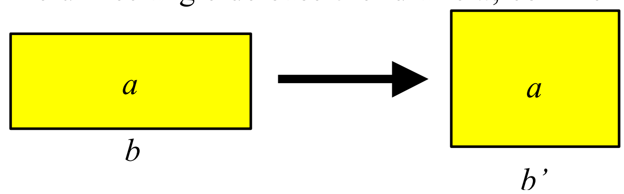
Rappelons-nous que nous voulons calculer une valeur approchée de la racine carrée du nombre positif a. En effet, il est impossible en général d'obtenir une réponse exacte sous la forme d'un nombre décimal fini puisque plusieurs racines carrées (même de nombres très simples comme 2,

[1] Voir par exemple **A. Doneddu**, *Arithmétique générale*, **Dunod, 1962**.

[2] Pour ne citer qu'un cas, il existe de très nombreuses façons de calculer des valeurs approchées du nombre π, et la justification mathématique de certaines est assez délicate. Voir par exemple **Jean-Paul Delahaye**, *Le fascinant nombre π*, **Pour la Science Diffusion Belin, 1997**.

3 ou 5 — mais non 4) sont des nombres irrationnels, nécessitant donc une écriture décimale infinie non périodique.

Posons-nous donc la question suivante : étant donné un rectangle de base b et d'aire a, comment pouvons-nous obtenir une nouveau rectangle, de base b' mais toujours d'aire a, qui soit plus près d'un carré que le premier?



Si le rectangle de départ n'est pas déjà carré, on sait que la base et la hauteur ne peuvent être toutes deux inférieures ou toutes deux supérieures à la racine carrée cherchée. (Pourquoi?) L'une des deux valeurs est donc plus petite (et l'autre plus grande) que cette racine carrée. On peut alors espérer que si on prend la moyenne b' de ces deux valeurs comme base d'un nouveau rectangle, on obtiendra une meilleure approximation de la racine carrée en question.

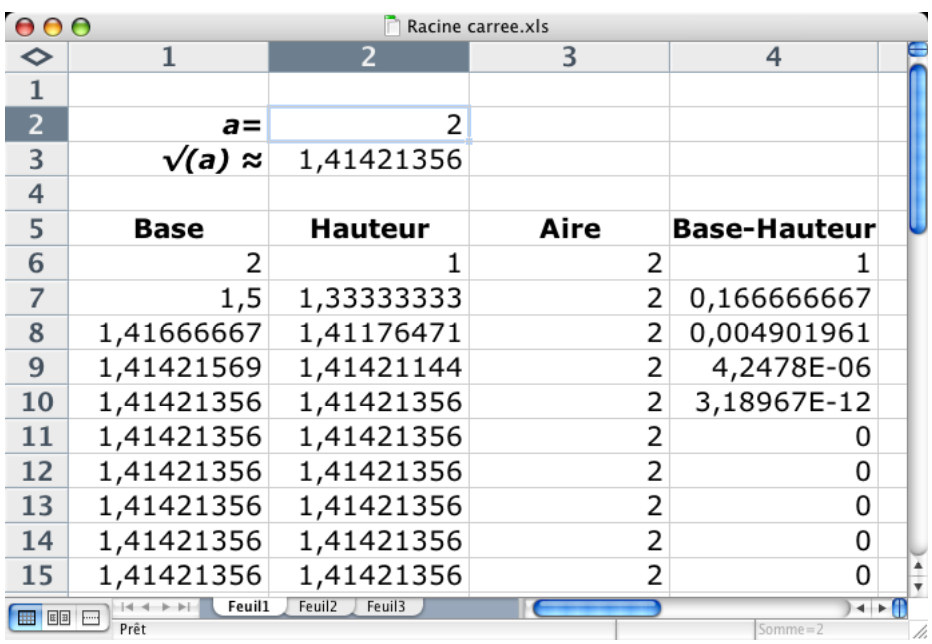
En fait, on peut montrer (voir Exercice 7) que si b et h sont la base et la hauteur du rectangle initial, et si b' et h' sont la base et la hauteur du nouveau rectangle, on a

$$b < h' < \sqrt{a} < b' < h \text{ (dans le cas où } b < h\text{)}$$

ou

$$h < h' < \sqrt{a} < b' < b \text{ (dans le cas où } h < b\text{).}$$

Nous sommes maintenant prêts à créer une feuille Excel réalisant cet algorithme. Celle-ci pourrait ressembler à la figure 3.5.1. Notez que seules les deux premières colonnes sont nécessaires : les colonnes 3 et 4 sont ajoutées comme outils de vérification.


Racine carree.xls

	1	2	3	4
1				
2	a=	2		
3	√(a) ≈	1,41421356		
4				
5	Base	Hauteur	Aire	Base-Hauteur
6	2	1	2	1
7	1,5	1,33333333	2	0,166666667
8	1,41666667	1,41176471	2	0,004901961
9	1,41421569	1,41421144	2	4,2478E-06
10	1,41421356	1,41421356	2	3,18967E-12
11	1,41421356	1,41421356	2	0
12	1,41421356	1,41421356	2	0
13	1,41421356	1,41421356	2	0
14	1,41421356	1,41421356	2	0
15	1,41421356	1,41421356	2	0

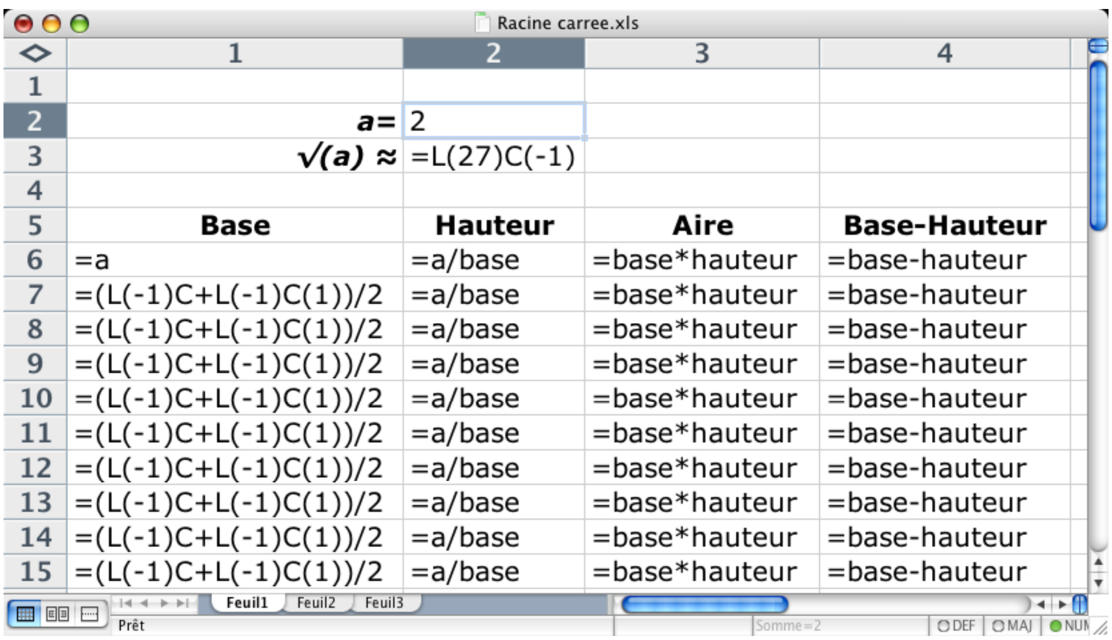
Feuil1 Feuil2 Feuil3
Prêt Somme=2

Figure 3.5.1 Feuille de calcul permettant de calculer des racines carrées par la méthode des « rectangles de plus en plus carrés ».

Pour réaliser cette feuille de calcul, nous procédons de la façon suivante: d'abord définir les cellules contenant du texte, ensuite nommer les cellules (L2C2 sera « a » et L3C2 « reponse ») et les colonnes (« base » et « hauteur ») pertinentes, puis remplir la table des valeurs, pour enfin spécifier le contenu de L3C2 via une formule. La définition de la table de valeurs se fera en trois temps : remplir d'abord la ligne 6 (valeurs initiales), puis définir la ligne 7 en fonction de la ligne 6, et enfin recopier la ligne 7. La figure 3.5.2, où nous avons paramétré *Excel* pour qu'il affiche les formules plutôt que les valeurs, rend compte de tout ceci.

Comment faire afficher les formules plutôt que les valeurs?

Excel 2004 pour Macintosh : à partir du menu « **Excel** » faire successivement les choix suivants : « Préférences... » ➤ « Affichage » ➤ cocher « Formules ».
Excel 2003 pour Windows : à partir du menu « **Outils** » faire successivement les choix suivants : « Options... » ➤ « Affichage » ➤ cocher « Formules ».
Excel 2007 pour Windows : cliquer sur le bouton « **Office** » ➤ cliquer sur le bouton « Options Excel » et faire successivement les choix suivants : « Options avancées » ➤ cocher « Formules dans les cellules au lieu de leurs résultats calculés » (dans la section « Afficher les options pour cette feuille de calcul : »).



	1	2	3	4
1				
2	***a=***	2		
3	***√(a) ≈***	=L(27)C(-1)		
4				
5	**Base**	**Hauteur**	**Aire**	**Base-Hauteur**
6	=a	=a/base	=base*hauteur	=base-hauteur
7	=(L(-1)C+L(-1)C(1))/2	=a/base	=base*hauteur	=base-hauteur
8	=(L(-1)C+L(-1)C(1))/2	=a/base	=base*hauteur	=base-hauteur
9	=(L(-1)C+L(-1)C(1))/2	=a/base	=base*hauteur	=base-hauteur
10	=(L(-1)C+L(-1)C(1))/2	=a/base	=base*hauteur	=base-hauteur
11	=(L(-1)C+L(-1)C(1))/2	=a/base	=base*hauteur	=base-hauteur
12	=(L(-1)C+L(-1)C(1))/2	=a/base	=base*hauteur	=base-hauteur
13	=(L(-1)C+L(-1)C(1))/2	=a/base	=base*hauteur	=base-hauteur
14	=(L(-1)C+L(-1)C(1))/2	=a/base	=base*hauteur	=base-hauteur
15	=(L(-1)C+L(-1)C(1))/2	=a/base	=base*hauteur	=base-hauteur

Figure 3.5.2 Les formules de la feuille de calcul des racines carrées.

La plupart des formules utilisées semblent aller de soi et être conformes à la démarche mathématique précédente. Soulignons au passage que le contenu de la cellule L7C1 est obtenu en faisant la moyenne de la base et de la hauteur de la ligne précédente. Seule la formule définissant la cellule L3C2 peut sembler étrange : elle va chercher la valeur dans la colonne « base » 27 lignes plus bas, soit à la ligne 30 de la feuille, c'est-à-dire à la ligne 25 de notre table de valeurs. Est-on certain que 25 étapes suffisent toujours pour obtenir une bonne approximation de la racine carrée?

Bien que nous sachions que cette méthode[1] converge rapidement vers la racine carrée (voir exercice 7), nous ne pouvons pas affirmer avec une certitude mathématique que 25 étapes seront **toujours** suffisantes. Par mesure de précaution, nous pourrions prolonger notre table à 100, 1000 ou même 10000 étapes et adapter notre formule de façon correspondante. Mais nous ne devons pas oublier que de multiplier la taille de nos calculs par un facteur de 4, 40 ou 400 s'accompagne d'une multiplication correspondante du temps nécessaire à les réaliser, ce qui peut rendre notre feuille de calcul peu pratique sur des ordinateurs moins rapides.

Une autre approche possible est de faire vérifier la précision de notre résultat en utilisant notre feuille de calcul. Une façon de faire serait de mettre au carré la valeur trouvée pour voir si ce carré correspond à la valeur « a » initiale, par exemple en ajoutant la formule

« = reponse*reponse »

dans la cellule L3C3. Une autre façon serait de contrôler si notre réponse est correcte à un certain nombre de décimales (disons 8) près, en affichant un avertissement si ce n'est pas le cas. En se rappelant que, à partir de la seconde itération de notre algorithme de calcul, on a toujours

$$h < \sqrt{a} < b \,, \text{ et donc } \; 0 < b - \sqrt{a} < b - h,$$

on peut en conclure que, lorsque $b - h < 0{,}00000001$, on est certain d'avoir la précision désirée. On pourrait réaliser ceci en ajoutant la formule

=SI(L(27)C4<0,00000001; "Précision atteinte";"Imprécision possible")

dans une autre cellule de la ligne 4, disons en L3C4.

Prenons un moment pour décrire la **fonction « SI »** que nous venons de rencontrer. Sa forme générale est

SI (*condition* ; *valeur1* ; *valeur2*).

La valeur retournée par la fonction SI sera *valeur1* (si la *condition* est vérifiée) ou *valeur2* (si la *condition* n'est pas vérifiée). Notez que les valeurs en question peuvent être des nombres, des textes, des dates, etc. Dans une formule, pour indiquer qu'il s'agit bien d'un texte, on doit l'encadrer de guillemets.

Dans notre cas particulier, notre fonction SI peut s'interpréter de la façon suivante : au termes de 25 étapes, si les 8 premières décimales de la différence entre la base et la hauteur sont 0, alors on placera le texte *Précision atteinte* dans L3C4, sinon on y placera le texte *Imprécision possible*. Voici à quoi pourrait ressembler notre feuille après ces ajouts (figure 3.5.3).

---

[1] Notons en passant que notre méthode, toute intuitive et simple qu'elle soit, est équivalente à un cas particulier de la méthode de Newton-Raphson. Voir par exemple **Marc Bourdeau, Jacques Gélinas**, *Analyse numérique élémentaire*, 2e édition, **Gaetan Morin Éditeur, 1987**.

	1	2	3	4
1				
2	a=	2		
3	√(a) ≈	1,41421356	2	Précision atteinte
4				
5	Base	Hauteur	Aire	Base-Hauteur
6	2	1	2	1
7	1,5	1,33333333	2	0,166666667
8	1,41666667	1,41176471	2	0,004901961
9	1,41421569	1,41421144	2	4,2478E-06
10	1,41421356	1,41421356	2	3,18967E-12
11	1,41421356	1,41421356	2	0
12	1,41421356	1,41421356	2	0
13	1,41421356	1,41421356	2	0
14	1,41421356	1,41421356	2	0
15	1,41421356	1,41421356	2	0

Figure 3.5.3 Ajout de contrôle de la précision à notre feuille précédente.

Jetons un coup d'oeil rétrospectif sur ce que nous venons de réaliser. Si nous voulons simplement calculer des racines carrées, l'approche via des « rectangles de plus en plus près d'un carré » et son implantation dans *Excel* s'avèrent assez simples, et paraissent accessible à des élèves de niveau secondaire ayant une certaine expérience avec un tableur. Par contre, quand vient le temps de certifier que le résultat obtenu est correct, les connaissances mathématiques et technologiques (usage de la fonction SI) nécessaires sont plus importantes. Ceci est un phénomène bien connu : il est plus facile de décrire que de prouver!

3.6 La méthode de bissection

Il nous est déjà arrivé de presser de façon répétée sur la touche « cos » d'une petite calculatrice scientifique pour nous apercevoir que, quel que soit le nombre de départ, nous obtenions finalement un nombre invariant, à peu près égal à 0,739085133 (en mode radians, et 0,999847742 en mode degrés). Un instant de réflexion nous amère à conclure que ce nombre doit être l'unique solution de l'équation $\cos(x) = x$ (voir figure 3.6.1).

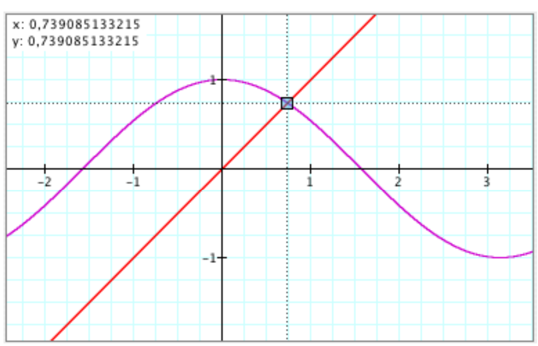


Figure 3.6.1 Intersection des courbes $y = \cos(x)$ et $y = x$.

En mathématiques, on rencontre souvent de telles équations pour lesquelles il n'existe pas de formules explicites permettant d'exprimer la ou les solutions. Par exemple, Abel et Galois nous ont montré que c'est le cas pour les équations polynomiales de degré supérieur à quatre (si on exclut quelques cas particuliers tel $x^5 - 3 = 0$). Dans ces cas là, on est alors conduits à chercher des valeurs approchées des solutions. Il y a toute une branche des mathématiques, l'analyse numérique, qui s'intéresse à ce genre de questions.

Dans ce qui suit, nous nous limiterons à la recherche de solutions d'équations de la forme $f(x) = 0$ dans le cas où la fonction f prend des valeurs de signes différents aux bornes d'un intervalle $[\ a\ ,\ b\ ]$ donné, comme l'illustre la figure 3.6.2 où la fonction f est positive en 0 et négative en 1. En supposant que cette fonction soit continue, nous somme certains qu'elle s'annulera pour au moins une valeur dans l'intervalle en question.

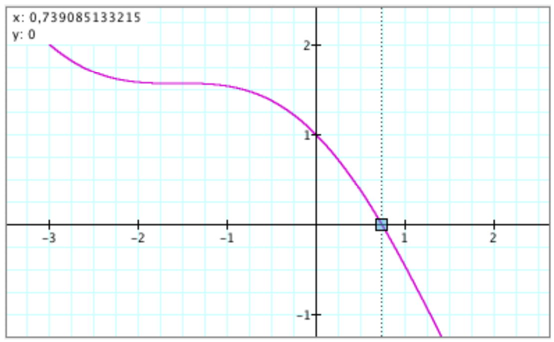


Figure 3.6.2 Zéro de la courbe $y = \cos(x)-x$.

Notons qu'on peut ramener la recherche de la solution de $\cos(x) = x$ à la recherche du zéro de la fonction $f(x) = \cos(x) - x$, qui est positive en 0 et négative en 1 : c'est le cas illustré à la figure précédente. De même, on peut ramener l'extraction de la racine carrée de 2 à la recherche du zéro de la fonction $f(x) = x^2 - 2$, qui est négative en 0 et positive en 2. Plus généralement, on obtient $\sqrt[n]{a}$ (où $a > 0$) comme seul zéro de la fonction $f(x) = x^n - a$, qui est négative en 0 et positive en a+1. (Pourquoi? Pourquoi ne peut-on dire que cette fonction sera nécessairement positive en a?)

Revenons au cas général d'une fonction f dont les valeurs sont de signes différents aux bornes d'un intervalle $[\ a\ ,\ b\ ]$. La figure 3.6.3 pourra nous aider à visualiser la description qui suit. Nous allons subdiviser cet intervalle en deux sous-intervalles de même longueur en utilisant $c = \dfrac{a+b}{2}$ le point milieu de cet intervalle : $[a,b] = [a,c] \cup [c,b]$. La fonction f change forcément de signe dans un de ces sous-intervalles (Pourquoi?), et nous pouvons donc continuer notre recherche de zéro de f dans un intervalle deux fois plus petit. En continuant de la même façon à subdiviser ce nouvel intervalle en deux, nous nous ramenons à un nouvel intervalle deux fois plus petit que ce dernier, et donc quatre fois plus petit que l'intervalle initial. Si on itère ce processus n fois, on obtient finalement un intervalle 2^n fois plus petit que l'intervalle initial et qui contient un zéro de f. Plus n est grand, plus l'intervalle ainsi obtenu est petit et par conséquent plus l'approximation de notre zéro est précise.

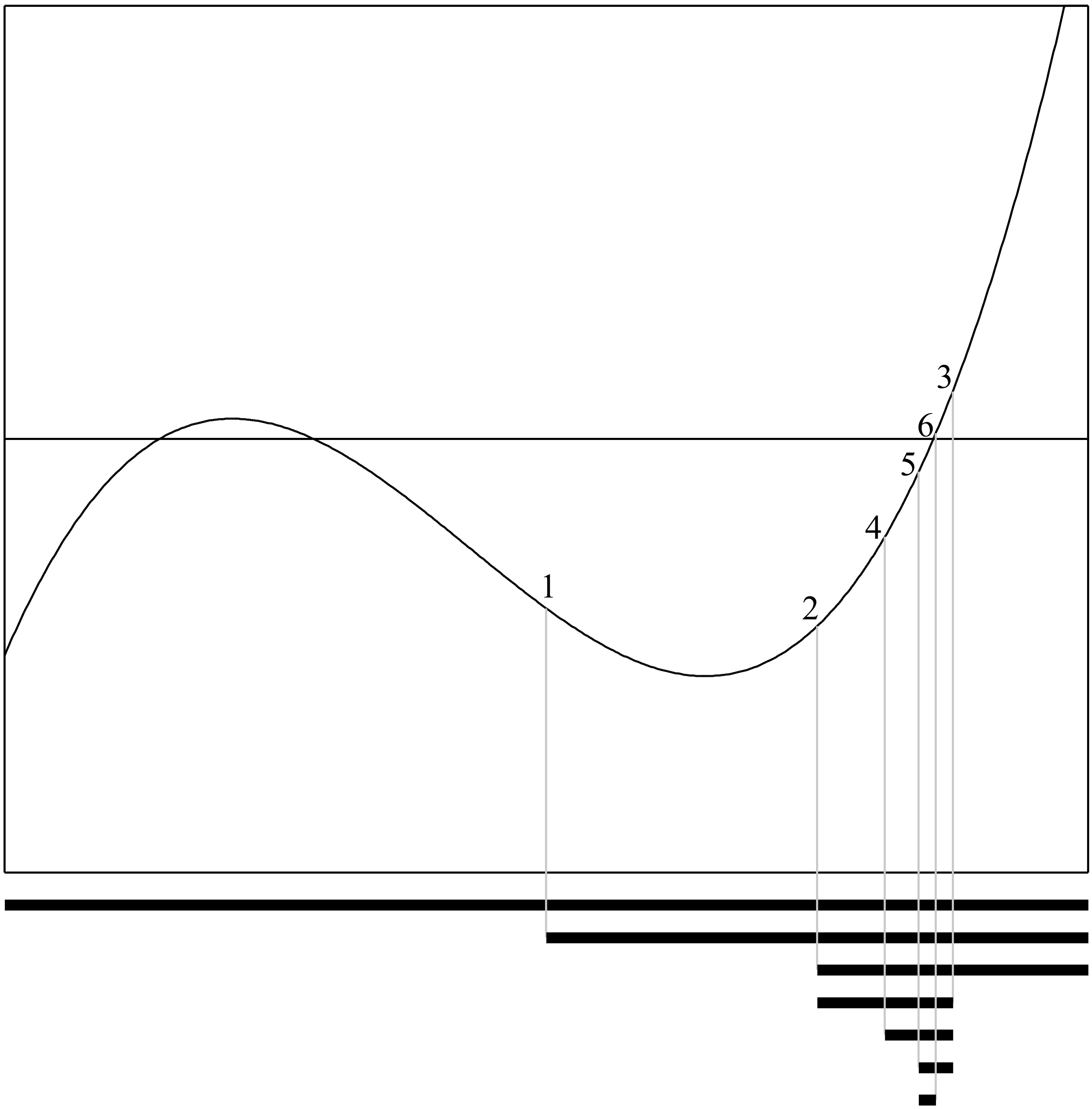
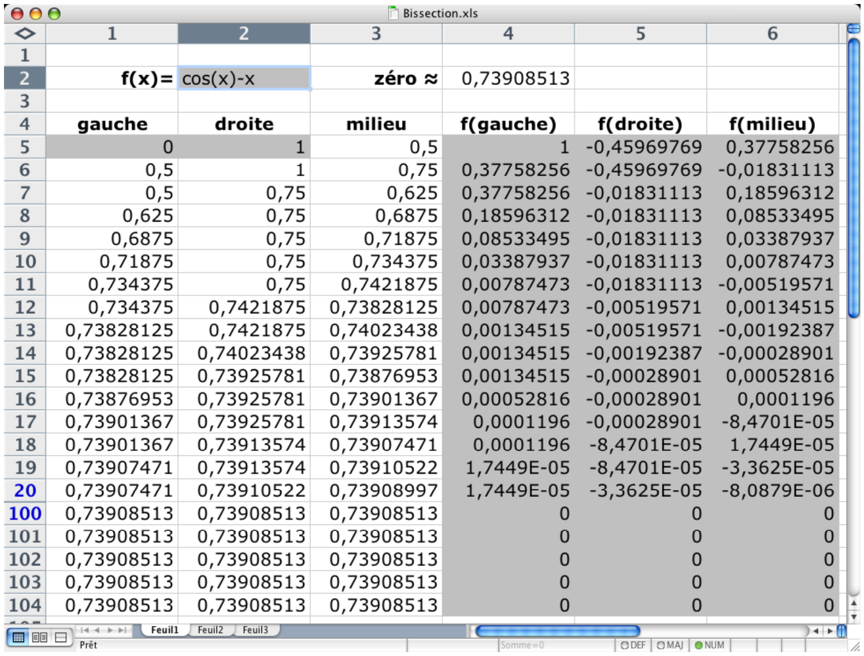


Figure 3.6.3 La méthode de bissection en action. Les points 1 à 6 correspondent aux milieux successifs, et, en bas du graphique, on peut voir les intervalles déterminés par ces points milieux. Dans notre cas particulier, les milieux 1, 2, 4 et 5 définissent de nouvelles bornes inférieures (à gauche), tandis que les milieux 3 et 6 définissent de nouvelles bornes supérieures (à droite). Notez que cet algorithme, appliqué à une fonction donnée dans un intervalle donné, cerne un et un seul zéro de la fonction, même si celle-ci en possède plus d'un.

Nous allons donc créer une feuille *Excel* implantant la méthode de bissection. Le résultat final ressemblera à la figure 3.6.4.



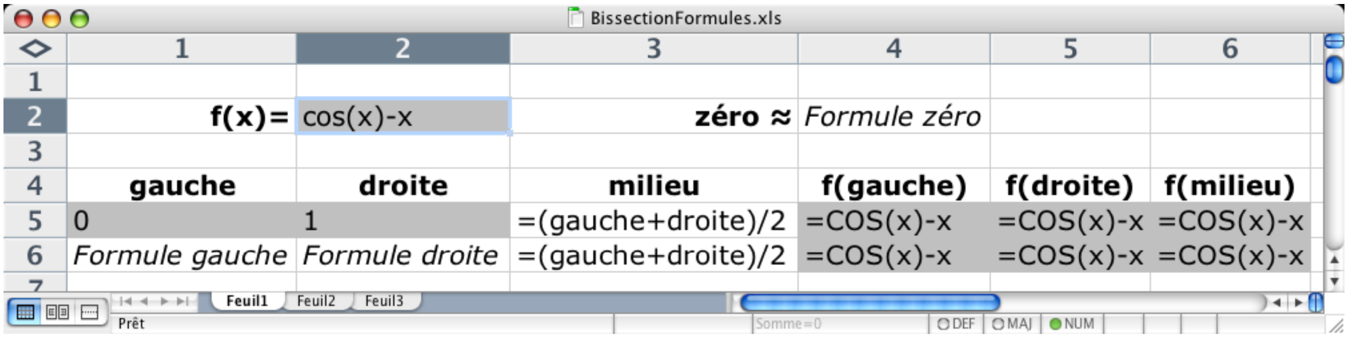
	1	2	3	4	5	6
1						
2	f(x)=	cos(x)-x	zéro ≈	0,73908513		
3						
4	gauche	droite	milieu	f(gauche)	f(droite)	f(milieu)
5	0	1	0,5	1	-0,45969769	0,37758256
6	0,5	1	0,75	0,37758256	-0,45969769	-0,01831113
7	0,5	0,75	0,625	0,37758256	-0,01831113	0,18596312
8	0,625	0,75	0,6875	0,18596312	-0,01831113	0,08533495
9	0,6875	0,75	0,71875	0,08533495	-0,01831113	0,03387937
10	0,71875	0,75	0,734375	0,03387937	-0,01831113	0,00787473
11	0,734375	0,75	0,7421875	0,00787473	-0,01831113	-0,00519571
12	0,734375	0,7421875	0,73828125	0,00787473	-0,00519571	0,00134515
13	0,73828125	0,7421875	0,74023438	0,00134515	-0,00519571	-0,00192387
14	0,73828125	0,74023438	0,73925781	0,00134515	-0,00192387	-0,00028901
15	0,73828125	0,73925781	0,73876953	0,00134515	-0,00028901	0,00052816
16	0,73876953	0,73925781	0,73901367	0,00052816	-0,00028901	0,0001196
17	0,73901367	0,73925781	0,73913574	0,0001196	-0,00028901	-8,4701E-05
18	0,73901367	0,73913574	0,73907471	0,0001196	-8,4701E-05	1,7449E-05
19	0,73907471	0,73913574	0,73910522	1,7449E-05	-8,4701E-05	-3,3625E-05
20	0,73907471	0,73910522	0,73908997	1,7449E-05	-3,3625E-05	-8,0879E-06
100	0,73908513	0,73908513	0,73908513	0	0	0
101	0,73908513	0,73908513	0,73908513	0	0	0
102	0,73908513	0,73908513	0,73908513	0	0	0
103	0,73908513	0,73908513	0,73908513	0	0	0
104	0,73908513	0,73908513	0,73908513	0	0	0

Figure 3.6.4 Méthode de la bissection appliquée à $f(x) = \cos(x) - x$, dans l'intervalle [0,1].

Les cellules grisées correspondent aux cellules qu'il faudra redéfinir quand on voudra changer la fonction (L2C2) et l'intervalle (L5C1 et L5C2) de recherche. Notez que la cellule L2C2 contient du texte et non une formule. On voudrait que l'utilisateur puisse procéder comme suit :

- il tape les bornes gauche (dans L5C1) et droite (dans L5C2)
- il tape dans L2C2 le texte correspondant à la fonction à utiliser
- il copie cette cellule et il entre dans L5C4 la formule obtenue en tapant « = » suivi de coller
- il copie ensuite la cellule L5C4
- il sélectionne toute la plage des valeurs de la fonction (L5C4:L104C6) et il fait « coller ».

Pour que ceci soit possible, il faudra définir le nom « *x* » comme signifiant « la cellule trois colonnes à gauche sur la même ligne » (=Feuil1!LC(-3)) de la façon décrite à la figure 3.3.2. Pour compléter la feuille de calcul, on peut nommer les colonnes 1 (« gauche ») et 2 (« droite »), puis entrer textes et formules comme indiqué dans la figure 3.6.5 . Il suffira ensuite de recopier la ligne 6 vers le bas pour remplir notre table.



BissectionFormules.xls

	1	2	3	4	5	6
1						
2	**f(x)=**	cos(x)-x	**zéro ≈**	*Formule zéro*		
3						
4	**gauche**	**droite**	**milieu**	**f(gauche)**	**f(droite)**	**f(milieu)**
5	0	1	=(gauche+droite)/2	=COS(x)-x	=COS(x)-x	=COS(x)-x
6	*Formule gauche*	*Formule droite*	=(gauche+droite)/2	=COS(x)-x	=COS(x)-x	=COS(x)-x

Figure 3.6.5 Formules de la feuille implantant la méthode de bissection.

Clarté versus efficacité

Un coup d'oeil sur la figure 3.6.5 vous surprendra peut-être. À priori, il peut sembler surprenant de retrouver la même formule pour désigner **f(gauche)**, **f(droite)** et **f(milieu)**.
Il aurait été *plus clair* d'utiliser « =cos(gauche)-gauche », « =cos(droite)-droite » et « =cos(milieu)-milieu », mais il aurait alors fallu taper et recopier vers le bas trois formules différentes.
Par contre il est certainement *plus efficace* de taper une seule formule (« =cos(x)-x ») et de la recopier une seule fois dans une plage de cellules. Et cette efficacité est souhaitable si on désire utiliser à répétition notre feuille de calcul, en changeant au besoin la fonction et l'intervalle où elle change de signe.
Un compromis possible serait de se passer de noms, et d'utiliser « =cos(LC(-3))-LC(-3) », qui est à la fois claire (pour un utilisateur familier avec les références *Excel*) et efficace.

Trois formules n'ont pas été explicitées, car elles requièrent des explications plus détaillées. Voyons tout d'abord comment définir la *Formule gauche*. Si **f(gauche)** et **f(milieu)** ont même signe, alors **milieu** doit devenir notre nouvelle borne gauche, puisqu'il y a alors un changement de signe entre le milieu et la borne droite (Pourquoi?); sinon, notre borne gauche doit rester inchangée, puisqu'il y a alors un changement de signe entre le milieu et la borne gauche (**milieu** devra alors devenir notre nouvelle borne droite).

Avant d'expliciter tout cela, nous allons voir comment exprimer que deux valeurs ont même signe : on peut dire bien sûr que nos deux valeurs sont toutes deux positives, ou bien qu'elles sont toutes deux négatives; cependant, un instant de réflexion nous permet d'affirmer que c'est le cas si et seulement si leur produit est positif. Le calcul de la nouvelle borne gauche peut alors se formuler comme suit :

Si f(gauche) × f(milieu) ≥ 0
alors prendre pour nouvelle borne gauche le milieu de la ligne précédente
sinon prendre pour nouvelle borne gauche la borne gauche de la ligne précédente

ce qui peut s'exprimer par la formule *Excel* suivante :

=SI(L(-1)C(3)*L(-1)C(5)>=0;L(-1)C(2);L(-1)C)

Il faut aussi considérer le cas (exceptionnel, mais possible) où **f(milieu)** = 0 : on voudrait alors que **milieu** devienne à la fois notre nouvelle borne gauche et notre nouvelle borne droite, ce qui est pris en compte dans la formule *Excel* précédente par l'utilisation du « plus grand **ou égal** ».
Passons maintenant à la *Formule droite*. Un raisonnement semblable à celui fait pour la borne gauche nous amène à la formulation suivante

Si f(droite) × f(milieu) ≥ 0
alors prendre pour nouvelle borne droite le milieu de la ligne précédente
sinon prendre pour nouvelle borne droite la borne droite de la ligne précédente

ce qui peut s'exprimer par la formule *Excel* suivante :

=SI(L(-1)C(3)*L(-1)C(4)>=0;L(-1)C(1);L(-1)C)

Il ne nous reste plus qu'à décrire la *Formule zéro*. Le zéro approximatif trouvé par nos calculs correspondra à la valeur de **milieu** à la dernière ligne de notre tableau (soit L104C3, dans le cas illustré ici). Nous devons cependant parler d'un problème que nous avons évité de mentionner jusqu'à présent : combien d'étapes seront nécessaires pour cerner adéquatement un zéro ou, en d'autres mots, combien de lignes doit comporter notre tableau? Il semble difficile de faire une analyse mathématique de la situation pour le déterminer, étant donné que la fonction et l'intervalle à utiliser ne sont pas fixés une fois pour toutes : on veut que l'utilisateur puisse les modifier à sa guise. Nous avons choisi, un peu arbitrairement, un tableau de 100 lignes, mais nous avons voulu que l'utilisateur puisse constater par lui-même si ces 100 étapes avaient été suffisantes pour obtenir un résultat précis. Aussi avons-nous choisi de masquer les lignes 21 à 99 de la feuille de calcul (correspondant aux étapes de 16 à 96 de l'algorithme de bissection) pour que l'utilisateur puisse voir d'un seul coup d'oeil quelques étapes au début et à la fin du processus, et ainsi vérifier aux colonnes 4 à 6 du bas du tableau si les valeurs finalement obtenues sont presque nulles ou pas.

Comment masquer des lignes (ou des colonnes)

Excel 2004 pour Macintosh et ***Excel 2003 pour Windows :***
On sélectionne tout d'abord les lignes (ou les colonnes) à masquer, par un glisser englobant les étiquettes correspondantes. Puis on fait les choix suivants : Menu « Format » ➤ « Ligne » ➤ « Masquer » (ou Menu « Format » ➤ « Colonne » ➤ « Masquer »).
Pour réafficher les lignes (ou les colonnes), il suffira de sélectionner une zone d'étiquettes contenant les étiquettes des lignes (ou colonnes) masquées, puis de faire les choix suivants : Menu « Format » ➤ « Ligne » ➤ « Afficher » (ou Menu « Format » ➤ « Colonne » ➤ « Afficher »).
Excel 2007 pour Windows
On sélectionne tout d'abord les lignes (ou les colonnes) à masquer, par un glisser englobant les étiquettes correspondantes. Puis on fait les choix suivants : onglet « Accueil » ➤ « Format » (dans le groupe « Cellules ») ➤ « Masquer & afficher » ➤ « Masquer les lignes » (ou « Masquer les colonnes »)
Pour réafficher les lignes (ou les colonnes), il suffira de sélectionner une zone d'étiquettes contenant les étiquettes des lignes (ou colonnes) masquées, puis de faire les choix suivants : onglet « Accueil » ➤ « Format » (dans le groupe « Cellules ») ➤ « Masquer & afficher » ➤ « Afficher les lignes » (ou « Afficher les colonnes »).

Il nous a semblé utile cependant d'avertir l'utilisateur qui aurait spécifié des bornes initiales (gauche et droite) ne provoquant pas de changement de signe de la fonction choisie. D'où la *Formule zéro* suivante :

=SI(L5C4*L5C5>=0; "ERREUR: le signe ne change pas…";L104C3)

3.7 Annexe : le style de références A1

Dans ce livre, nous avons choisi de privilégier le style de référence L1C1 dans notre approche des tableurs. Nous avons déjà vu qu'*Excel* permet aussi l'utilisation d'un autre style de référence, désigné par A1. Certains autres tableur ne nous donnent pas un tel choix : ils ne mettent à notre disposition que le style A1. Pour ceux qui désireraient utiliser ces tableurs, et aussi ceux qui voudraient pouvoir suivre des explications données en utilisant le style A1, nous allons discuter de la correspondance entre ces deux styles de référence.

Les références de style A1 utilisent une lettre majuscule ou une paire de lettres majuscules pour désigner les colonnes, et un nombre naturel pour désigner les lignes. Les références absolues sont indiquées par l'utilisation du préfixe « $ », à la fois pour les lignes et les colonnes. On a donc la correspondance suivante :

A1	...	Z1	AA1	...	AZ1	BA1	...	BZ1	...
L1C1	...	L1C26	L1C27	...	L1C52	L1C53	...	L1C78	...

et, plus généralement, le numéro de la colonne se calcule via la formule

26 × (le numéro d'ordre de la lettre de gauche) + (le numéro d'ordre de la lettre de droite),

quand la référence comporte deux lettres.

Le cas des références relatives est plus délicat. Illustrons tout d'abord ceci par un exemple :

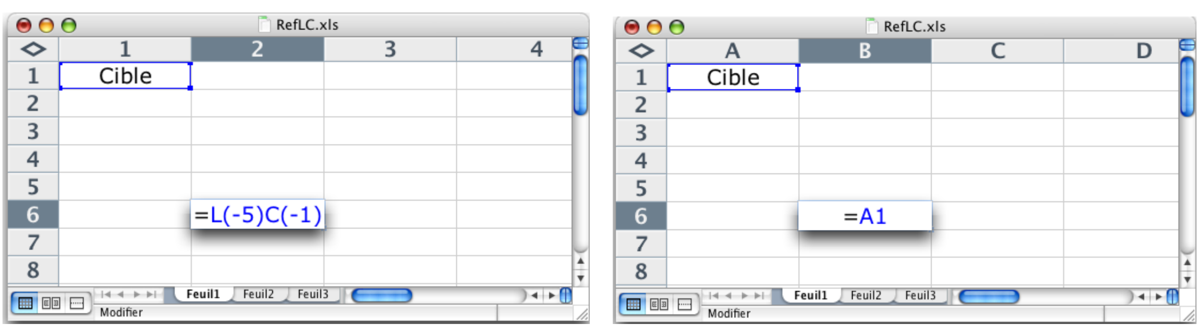


Figure 3.7.1 Comparaison des styles de références dans le cas relatif.

Voyons comment cela fonctionne dans le cas du style L1C1 (voir figure 3.7.1, à gauche). La référence relative L(-5)C(-1) est une indication des opérations à effectuer pour obtenir l'adresse de la cellule cible : il faut ajouter -5 à la ligne (6+[-5]=1) et -1 à la colonne (2+[-1]=1). La référence relative est donc une description des opérations à effectuer pour passer de la cellule source (celle contenant la formule) à la cellule cible. Dans le cas du style A1 (voir figure 3.7.1, à droite), la référence relative est tout simplement une description de la cellule cible (colonne et ligne), et non pas des opérations pour passer de la cellule source à la cellule cible.

Voyons maintenant ce qui se passe quand on copie cette cellule (L6C2 alias B6) une ligne et une colonne plus bas (soit en L7C3 alias C7). Nous obtenons la figure 3.7.2 :

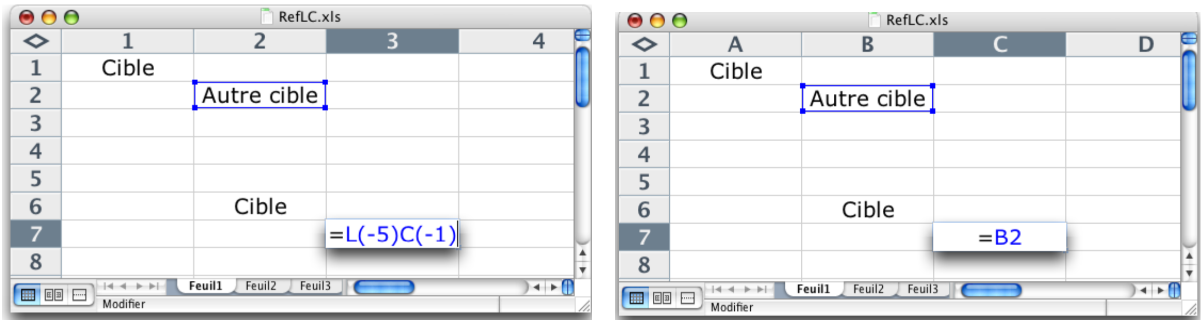


Figure 3.7.2 Copie d'une formule, selon le style de référence.

Dans le cas du style L1C1, la formule a été recopiée sans changement et désigne toujours une cellule 5 lignes plus haut et 1 colonne à gauche. Pour ce qui est du style A1, la formule n'est plus du tout la même : on peut imaginer qu'il a fallu calculer (dans le cas de la cellule où le « copier » a été fait) l'opérateur permettant de passer de la cellule source à la cellule cible; puis on a appliqué cet opérateur à la cellule où le « coller » a été fait, pour obtenir la nouvelle cellule cible, et donc la nouvelle formule.

Question sur la mécanique interne des tableurs

Quel que soit le style de référence employé, les tableurs doivent utiliser parfois les cellules cibles et parfois les opérateurs pour passer des cellules sources au cellules cibles. On peut se demander sous quelle forme sont stockées, à l'interne, les références relatives : cellules cibles, opérateurs pour passer des cellules sources au cellules cibles, ou coexistence des deux formes (avec un gaspillage de mémoire mais un gain de vitesse).

3.8 Exercices

1- *Extension du registre de notes*

Inspirez-vous de la section 3.2 pour constituer un registre de note comprenant 10 épreuves (dans les colonnes 2 à 11). Entrez des notes maximales à votre choix sur la ligne 3. Entrez des pondérations à votre choix sur la ligne 4 (la somme doit être 1).

Sans utiliser de noms :

a) Entrez dans la colonne 12 les formules permettant de calculer les notes finales.

b) Complétez la ligne 17 pour quelle affiche les moyennes de chaque épreuve et des notes finales.

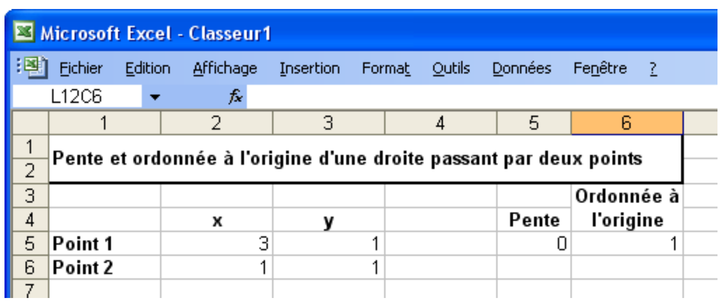
c) Puisque la somme des pondérations doit être égale à 1, lorsque 9 de celles-ci sont connues la 10^{e} peut être calculée. Entrez dans la cellule L4C11 une formule calculant la pondération de la 10^{e} épreuve en fonction des 9 autres (Vous pouvez utiliser la fonction *Excel* SOMME).

d) Il est possible que les pondérations entrées soient inconsistantes (< 0 ou > 1). On peut mettre ces cas en évidence en utilisant une « Mise en forme conditionnelle ». Aidez-vous de l'aide en ligne pour faire afficher en rouge les cellules L4C2 à L4C11 dont le contenu n'est pas admissible.
Remarque : lorsqu'on recopie une cellule le format aussi est copié y compris un format conditionnel.

e) Utilisez un format conditionnel pour que les cellules contenant une note finale inférieure à 60 soient aussi en rouge.

2- *Nommer les éléments du registre de notes*
Reprenez l'exercice 1 et inspirez-vous de la section 3.3 pour nommer les différents éléments de la feuille. Modifiez les formules pour utiliser les noms.

3- *Pente et ordonnée à l'origine d'une droite passant par deux points*
Créez une feuille Excel dans laquelle on peut entrer les abscisses et ordonnées de deux points et qui calcule la pente et l'ordonnée à l'origine de la droite passant par ces deux points.



	1	2	3	4	5	6
1	Pente et ordonnée à l'origine d'une droite passant par deux points					
2						
3						Ordonnée à
4		x	y		Pente	l'origine
5	Point 1	3	1		0	1
6	Point 2	1	1			
7						

a) Que se passe-t-il si les deux points ont la même abscisse?

b) Modifiez les formules calculant la pente et l'ordonnée à l'origine pour quelles affichent « indéterminé » lorsque les deux points ont la même abscisse (Utilisez la fonction SI).

4- *Les nombres de Fibonacci*
Les nombres de Fibonacci sont des entiers calculés de la façon suivante : les deux premiers sont égaux à 1 et, à partir du troisième, chacun est égal à la somme des deux précédents. Donc le troisième est 1+1=2, le quatrième 1+2=3, etc. La suite de ces nombres commence donc par 1, 1, 2, 3, 5, 8, 13, etc.

a) Construisez une feuille de calcul qui permettra d'obtenir les 100 premiers nombres de Fibonacci. Placez leur rang (de 1 à 100) sur la colonne 1 et le nombre de Fibonacci correspondant sur la colonne 2.

b) Sur la colonne 3, pour chaque nombre de Fibonacci à partir du second, faites calculer le rapport de ce nombre avec le précédent. Que constatez-vous?
Pour chaque nombre de Fibonacci, calculez sur la colonne 4, sa différence avec le nombre d'or $\dfrac{1+\sqrt{5}}{2}$. Que constatez-vous?

c) Dans la cellule L4C6 on va calculer le rang ***r*** du premier nombre de Fibonacci plus grand qu'un entier donné ***n***.

Commencez par entrer dans la cellule L3C6 le nombre entier positif ***n***.
Ensuite, dans la colonne 5, dans les cellules à droite de chaque nombre de Fibonacci entrez une formule qui écrit le rang du nombre si celui-ci est plus grand que ***n*** et "" sinon. Le minimum de la colonne 5 est alors le rang ***r*** cherché.

- Dans la cellule L3C7 utilisez la fonction MIN pour écrire une formule affichant le rang ***r*** (Pour utiliser MIN consultez l'aide en ligne si nécessaire).
- Dans la cellule L6C7 faites afficher le nombre de Fibonacci de rang ***r***.

Remarque : la fonction MIN retourne le minimum des nombres contenus dans la plage de données à laquelle elle est appliquée en négligeant les cellules qui ne contiennent pas des nombres.

d) Trouvez une méthode permettant de refaire la partie c) sans utiliser la fonction MIN?

e) Une des propriétés des nombres de Fibonacci concerne la somme des carrés des n premiers. On a découvert que cette somme peut s'exprimer en fonction de deux de ces nombres. Créez une feuille de calcul permettant d'explorer ce phénomène de façon à découvrir expérimentalement, par essais et erreurs, comment exprimer la somme en question en fonction de deux nombres de Fibonacci.

	1	2	3	4	5
1	**n**	**Le n ième nombre de Fibonacci**	**La somme des carrés**	**Essai de formule**	**Erreur**
2	1	1	1	1	0
3	2	1	2	2	0
4	3	2	6	6	0
5	4	3	15	15	0
6	5	5	40	40	0
7	6	8	104	104	0
8	7	13	273	273	0
9	8	21	714	714	0
10	9	34	1870	1870	0
11	10	55	4895	4895	0

5- *Le nombre à virgule (fini ou périodique) associé à une fraction*

Créez une feuille de calcul qui, étant donnée une fraction, calcule les 100 premières décimales du nombre à virgule associé.

Notez qu'on peut conclure du tableau ci-dessous que $\frac{1}{7} = 0{,}\overline{142857}$.

Utilisez ensuite cette feuille de calcul pour tenter de caractériser les fractions dont le nombre à virgule associé est fini.

Vous aurez peut-être besoin de fonctions telles ENT, TRONQUE, QUOTIENT ou MOD. Pour plus de détails, utilisez l'aide en ligne.

	1	2	3
1	Numérateur	1	
2	Dénominateur	7	
3			
4	Partie entière	0	
5			
6	Position	Décimale	Reste
7	1	1	3
8	2	4	2
9	3	2	6
10	4	8	4
11	5	5	5
12	6	7	1
13	7	1	3
14	8	4	2
15	9	2	6
16	10	8	4

6- *Amélioration de la feuille du calcul d'hypothèque*
Modifiez la feuille de calcul des hypothèques en ajoutant une cellule dans laquelle on entrera le taux d'intérêt annuel b affiché par les banques canadiennes. Le taux d'intérêt mensuel t (cellule L4C4) sera alors calculé à l'aide de la formule $(1+t)^{12} = \left(1+\frac{b}{2}\right)^2$. Pour ça vous pourrez utiliser la fonction PUISSANCE(nombre; exposant), qui permet d'élever un nombre décimal à une puissance décimale. Par exemple PUISSANCE(3,5 ; 1,3) = $3{,}5^{1,3}$.

7- *Les mathématiques sous-jacentes au calcul de la racine carré de la section* 3.5
Si a, b, h sont trois nombres tels que $0 < h < \sqrt{a} < b$ et $bh = a$ montrez que :

a) $\sqrt{a} < \dfrac{b+h}{2}$.

b) si on pose $b' = \dfrac{b+h}{2}$ et $h' = \dfrac{a}{b'}$ alors $h < h' < \sqrt{a} < b' < b$

c) avec les mêmes notations que ci-dessus $b' - h' < \dfrac{b-h}{2}$.

d) toujours avec les mêmes notations, si on appelle $e = \dfrac{b-\sqrt{a}}{\sqrt{a}}$ et $e' = \dfrac{b'-\sqrt{a}}{\sqrt{a}}$ deux erreurs relatives successives, on a $e' < \dfrac{e^2}{2}$.

8- *Approximation de la racine carré : calculs d'erreur*
Complétez la feuille de calcul de la racine carré pour implanter les calculs d'erreurs décrits à la page 68.

9- *Approximations de racines n-ième par la méthode de bissection*

Calculer $\sqrt[n]{a}$ c'est trouver une solution à l'équation $x^n - a = 0$.

a) Utilisez la méthode de bissection pour calculer $\sqrt{3}$, $\sqrt[3]{3}$, $\sqrt[7]{26}$, $\sqrt[0,3]{7}$.

b) Comparez vos résultats avec ceux donnés par la fonction PUISSANCE.

c) Pour calculer $\sqrt[n]{a}$, *n* et *a* étant variables, modifiez la feuille de calcul de la méthode de bissection de la façon suivante :

- la fonction sera toujours $x^n - a$
- ajoutez deux cellules qui contiendront les valeurs des paramètres *n* et *a*.
- modifiez les formules pour calculer $\sqrt[n]{a}$ en fonction des valeurs de *n* et *a* présentes dans les nouvelles cellules.

	1	2	3	4	5	6
1	**n=**	4	**a=**	16		
2	**f(x)=**	x^n - a	**zéro ≈**	2	**≈ racine n ième de a**	
3						
4	**gauche**	**droite**	**milieu**	**f(gauche)**	**f(droite)**	**f(milieu)**
5	0	17	8,5	-16	83505	5204,0625
6	0	8,5	4,25	-16	5204,0625	310,253906
7	0	4,25	2,125	-16	310,253906	4,39086914
8	0	2,125	1,0625	-16	4,39086914	-14,725571
9	1,0625	2,125	1,59375	-14,725571	4,39086914	-9,5482016
10	1,59375	2,125	1,859375	-9,5482016	4,39086914	-4,0472469
11	1,859375	2,125	1,9921875	-4,0472469	4,39086914	-0,248539
12	1,9921875	2,125	2,05859375	-0,248539	4,39086914	1,95901857
13	1.9921875	2.05859375	2.02539063	-0.248539	1.95901857	0.82810378

10- *Résolution approchée d'une équation par la méthode de bissection*

Utilisez la méthode de bissection pour calculer la solution positive de l'équation $tg(x) = x + 1$. (Dans *Excel* la fonction tangente s'écrit TAN).

3.9 Projets

1- *Calcul des* ***pgcd*** *et* ***ppcm*** *de deux nombres naturels par l'algorithme d'Euclide*

L'algorithme d'Euclide consiste à exécuter une suite de divisions avec reste:

dividende = diviseur · quotient + reste

en passant d'une division à la suivante en prenant respectivement le diviseur et le reste de la première pour dividende et diviseur de la suivante. Le dernier reste non nul est le *pgcd*.

Illustrons la méthode par un exemple, soit le calcul du *pgcd* de 127 et 19. On débute avec nos deux nombres 127 et 19 pour dividende et diviseur respectivement:

127 = 19 · 6 + 13	
19 = 13 · 1 + 6	***On passe d'une division à la suivante en prenant respectivement le diviseur et le reste de la première pour dividende et diviseur de la suivante.***
13 = 6 · 2 + 1	
6 = 1 · 6 + 0	***On termine lorsqu'on obtient un reste nul. Le pgcd est alors le dernier reste non nul, qui est aussi le diviseur de la dernière division.***

L'application de l'algorithme à nos deux nombres 127 et 19 nous permet de conclure que leur pgcd est 1.

	1	2	3	4
1				
2		Premier nombre	134	
3		Second nombre	180	
4				
5				
6	Dividende	Diviseur	Quotient	Reste
7	134	180	0	134
8	180	134	1	46
9	134	46	2	42
10	46	42	1	4
11	42	4	10	2
12	4	2	2	0
13	2	0	#DIV/0!	#DIV/0!
14	0	#DIV/0!	#DIV/0!	#DIV/0!

a) Créez une feuille de calcul qui réalise cet algorithme.
Remarque : dans la feuille de calcul illustrée dans la figure ci-dessus nous avons fait les divisions sans vérifier si les diviseurs étaient différents de 0. On peut voir qu'*Excel* nous indique les cellules où il y a des tentatives de division par 0 tout en continuant d'exécuter correctement les calculs possibles.

b) Ajoutez à la feuille précédente le calcul du *ppcm* des deux nombres, et des vérifications que
   - le *pgcd* trouvé divise bien les deux nombres donnés
   - les deux nombres donnés divisent bien le *ppcm*.

2- *Une méthode élémentaire pour approximer des fonctions trigonométriques*
Créez un classeur *Excel* permettant de calculer (via la méthode décrite ci-dessous) le cosinus de tout angle compris entre 0 et 90 degrés.

Méthode :
Dans le projet 2 du chapitre 1, nous avons déjà vu que

$$\cos(\alpha+\beta) = \cos\alpha\cos\beta - \sin\alpha\sin\beta.$$

Si, dans cette formule, on choisit $\alpha = \beta$, on obtient

$$\begin{aligned}\cos(2\alpha) &= \cos^2\alpha - \sin^2\alpha \\ &= \cos^2\alpha - (1-\cos^2\alpha) \\ &= 2\cos^2\alpha - 1\end{aligned}$$

En résolvant pour cos α, on obtient

$$\cos\alpha = \pm\sqrt{\frac{1+\cos(2\alpha)}{2}}$$

où l'on doit choisir le signe + quand α est compris entre 0 et 90 degrés.
Posons maintenant γ = 2α. On obtient alors

$$\cos\left(\frac{\gamma}{2}\right) = \sqrt{\frac{1+\cos\gamma}{2}}$$

De plus, si $\gamma = \frac{a+b}{2}$, on obtient finalement :

$$\cos\left(\frac{a+b}{2}\right) = \sqrt{\frac{1+\cos(a+b)}{2}} = \sqrt{\frac{1+\cos a\cos b - \sin a\sin b}{2}}$$
$$= \sqrt{\frac{1+\cos a\cos b - \sqrt{1-\cos^2 a}\sqrt{1-\cos^2 b}}{2}}$$

Pour être rigoureux, il faut noter que nous avons pu utiliser l'identité

$$\sin x = \sqrt{1-\cos^2 x}$$

parce que les angles x en présence sont tous supposés valoir entre 0 et 90 degrés, et que nous savons que la fonction sinus est positive dans cet intervalle.

Cette formule nous permet donc de calculer le cosinus de la moyenne de deux angles quand on connaît le cosinus de chacun de ces deux angles.
Nous allons maintenant appliquer la formule précédente pour calculer le cosinus d'un angle compris entre 0 et 90 degrés. Nous utiliserons aussi le fait que la fonction cosinus est décroissante entre 0 et 90 degrés, ce qui peut s'exprimer de la façon suivante:

$$\text{si } x \le y \text{ alors } \cos(x) \ge \cos(y).$$

Illustrons tout ceci par un exemple : un angle de 37 degrés.

Approximations de l'angle de 37°	**Approximations de cos(37)**
$0 \le 37 \le 90$ $0 \le 37 \le 45 = \dfrac{0+90}{2}$ $\dfrac{0+45}{2} = 22.5 \le 37 \le 45$ $\dfrac{22.5+45}{2} = 33.75 \le 37 \le 45$	$\cos(0) \ge \cos(37) \ge \cos(90)$ $\cos(0) \ge \cos(37) \ge \cos(45) = \cos\left(\dfrac{0+90}{2}\right)$ $\cos\left(\dfrac{0+45}{2}\right) = \cos(22.5) \ge \cos(37) \ge \cos(45)$ $\cos\left(\dfrac{22.5+45}{2}\right) = \cos(33.75) \ge \cos(37) \ge \cos(45)$

Comme on connaît cos(0)=1 et cos(90)=0, on voit qu'on peut calculer tous les cosinus ultérieurs, qui ont toujours comme arguments des moyennes d'angles dont on connaît déjà le cosinus. De cette façon, on peut encadrer cos(37) de façon de plus en plus précise.
Notons en passant que cette démarche n'est pas spécifique à l'exemple de 37 degrés choisi au départ. Elle fonctionnera pour tout angle compris entre 0 et 90 degrés.
Prolongements :

a) Modifiez la feuille pour l'adapter au calcul du cosinus d'un angle quelconque.
b) Modifiez la feuille pour l'adapter au calcul du sinus et des autres fonctions trigonométriques.

En conclusion
Traditionnellement, on utilise des méthodes issues du calcul différentiel et intégral, et donc beaucoup plus sophistiquées du point de vue mathématique, pour calculer le cosinus et les autres fonctions trigonométriques.
Ceci est un autre exemple illustrant la remarque faite au début de la section 3.4 concernant l'interchangeabilité possible entre la sophistication mathématique et l'utilisation d'outils technologiques.

3- *Approximation des fonctions exponentielles*
Faire une feuille de calcul permettant d'approximer les fonctions exponentielles de type a^x en employant des méthodes élémentaires. Les calculs ne devront utiliser que les opérations $+$, -, $\times$, $\div$ et $\sqrt{\ }$ (fonction RACINE dans *Excel*).
Indice : on peut s'inspirer librement de la méthode de bissection.

4- *Approximation des fonctions logarithmiques*
Faire une feuille de calcul permettant d'approximer les fonctions logarithmiques de type $\log_a x$ en employant des méthodes élémentaires. Les calculs ne devront utiliser que les opérations $+$, -, $\times$, $\div$ et $\sqrt{\ }$ (fonction RACINE dans *Excel*).
Indice : on peut s'inspirer librement de la méthode de bissection en utilisant la moyenne géométrique $\sqrt{ab}$ à la place de la moyenne arithmétique $\dfrac{a+b}{2}$.

Chapitre 4

Excel, graphiques et interactivité

Au chapitre précédent, nous avons pu apprécier les formidables capacités numériques d'*Excel*. Dans ce chapitre, nous allons voir que son potentiel graphique est tout aussi impressionnant. Signalons tout d'abord qu'*Excel* dispose des mêmes outils de dessin que *Word*. De même, il existe une adaptation de *Langage Graphique* pour *Excel*, et elle comporte les mêmes primitives : on comprendra mieux son utilité lorsque nous utiliserons le classeur *PixelsGraphiques.xls* au chapitre 5. Mais pour l'instant, nous nous contenterons de la version créée pour *Word*.

Ce qui distingue *Excel* est sa capacité d'associer des graphiques de types variés à des données contenues dans des plages de cellules. Soulignons que cette correspondance est dynamique, en ce sens que tout changement des données se traduit instantanément par une modification du graphique correspondant, et réciproquement. Les options disponibles pour représenter nos données étant trop nombreuses pour que nous en fassions une revue complète, nous nous limiterons à les illustrer dans le cadre de quelques exemples convenablement choisis. Dans un premier temps, nous décrirons une feuille de calcul permettant d'obtenir des graphes cartésiens de fonctions réelles à une variable, en utilisant des barres de défilement pour varier interactivement certains paramètres. Par la suite, nous verrons comment les opérations matricielles nous permettent d'expérimenter en temps réel avec les transformations géométriques. Puis nous simulerons un phénomène aléatoire, tout en nous familiarisant avec les graphiques statistiques et le mode de calcul itératif dans *Excel*. Enfin nous nous demanderons comment une machine déterministe (ordinateur ou calculatrice) peut arriver à simuler le hasard.

4.1 Graphiques cartésiens

Depuis quelques années, les programmes scolaires font beaucoup de place à l'étude de l'effet de la variation des divers paramètres entrant dans la définition de familles de fonctions $af(b(x-h))+k$ basées sur une fonction $f(x)$. Comme cette étude se déroule, la plupart du temps, à un niveau simplement descriptif, on peut se demander si on n'est pas ici en présence d'un cas où la disponibilité de la technologie, qui permet très facilement une étude expérimentale de tels phénomènes, n'a pas influencé les programmes.

Quoi qu'il en soit, nous allons chercher à faire une feuille de calcul qui simulera certains aspects d'une calculatrice à affichage graphique. La figure 1.1.1 nous donne une idée du résultat final. Par rapport à ce type de calculatrice, cette feuille aura des avantages et des inconvénients. Côté avantages, signalons la présence de barres de défilement, qui permettent de changer interactivement les valeurs des quatre paramètres. Côté inconvénients, mentionnons la nécessité de faire un plus grand nombre de modifications manuelles si on veut changer la fonction étudiée, la région du plan dans laquelle se fera la représentation, ou le domaine de variation des divers paramètres.

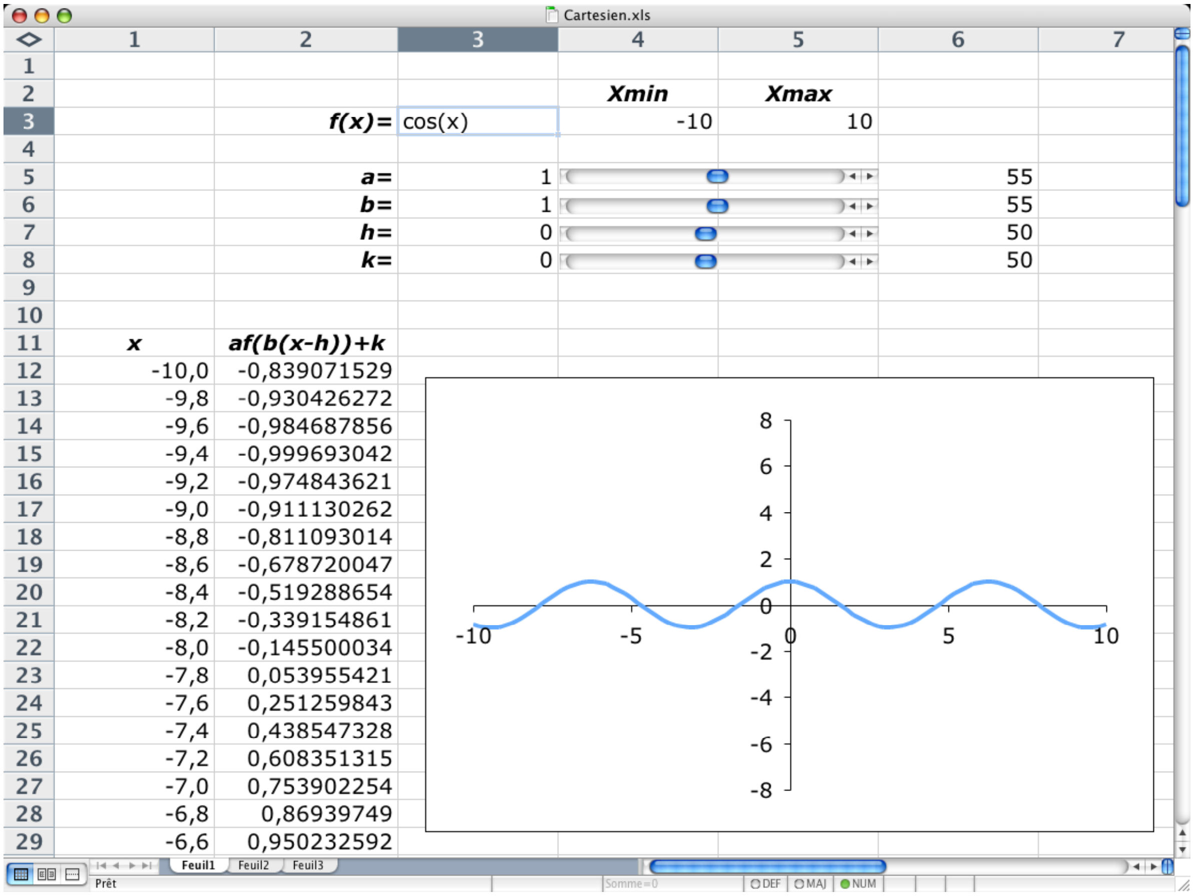


Figure 4.1.1 Représentation cartésienne d'une famille de fonctions dans *Excel*.

Comment apporter des changements à cette feuille de calcul?

Anticipons un peu et précisons comment nous pourrons changer les divers éléments quand nous aurons complété la feuille de calcul.

- Pour changer la fonction de base, on devra modifier non seulement la cellule L3C3 (qui contient une information pour l'utilisateur), mais aussi la formule indiquant comment calculer la valeur de $af(b(x-h))+k$ dans la colonne 2 (à partir de la ligne 12).
- Pour changer la région du plan utilisée, on doit modifier les propriétés correspondantes de chacun des deux axes (via un clic droit faisant apparaître le menu contextuel), mais aussi les cellules correspondant à **Xmin** (L3C4) et **Xmax** (L3C5).
- Pour changer le domaine de variation de chacun des paramètres, il faudra changer la formule de la cellule correspondante (dans la colonne 3).

On pourrait automatiser beaucoup plus le processus, en utilisant les ressources de programmation disponibles dans *Excel*. Mais ceci dépasserait le cadre de cet ouvrage.

Avant de décrire comment remplir les différentes cellules, ajouter les barres de défilement, et faire apparaître le graphique, il nous faut réfléchir aux représentations cartésiennes de fonctions. Mathématiquement, le graphe d'une fonction consiste en un ensemble infini de points $\{(x, f(x)) : x \in Dom(f)\}$. Comme nos outils informatiques ont une mémoire et une vitesse finies, on devra se restreindre à un nombre limité de points, quitte à les relier par des segments (ou, plus précisément, par des représentations informatiques de segments) pour compenser. Si nous choisissons d'utiliser trop peu de points, la représentation graphique résultante risque de ne pas être assez fidèle. Par contre, si le nombre de points choisis est trop grand, on risque d'utiliser inutilement beaucoup de mémoire et de ralentir indûment les calculs. Pour illustrer ceci, nous avons choisi d'utiliser 101 points (on verra plus tard pourquoi 101 et non 100), ce qui s'avèrera suffisant dans certains cas, mais pas dans tous. Quoi qu'il en soit, nous pourrions toujours modifier le nombre de points utilisés en redéfinissant (manuellement ou par programmation) certains éléments de notre feuille de calcul.

Oublions pour l'instant les barres de défilement, le graphique et les nombres de la colonne 6. Forts de l'expérience acquise au chapitre précédent, nous pouvons facilement définir et nommer les diverses cellules. On devine aisément les cellules devant recevoir les noms ***Xmin***, ***Xmax***, ***a***, ***b***, ***h*** et ***k***.

Voyons comment définir les cellules numériques de la colonne 1 : comme on veut utiliser 101 points, ceux-ci seront placés de la ligne 12 à la ligne 112. La cellule L12C1 sera définie par la formule « =Xmin », et la valeur de chaque cellule subséquente sera obtenue en ajoutant à la valeur précédente la longueur de l'intervalle entre deux points. Comme on a choisi 101 points, on aura donc 100 intervalles, chacun de longueur $\frac{Xmax - Xmin}{100}$: d'où l'utilisation de la formule

« =L(-1)C+(Xmax-Xmin)/100 »

pour les cellules des lignes 13 à 111 de la colonne 1. On voit maintenant pourquoi on a choisi 101 points et non 100 : pour avoir 100 intervalles, et donc des nombres avec peu de décimales (si, bien entendu, ***Xmin*** et ***Xmax*** ont eux-mêmes peu de décimales).

Une approche alternative que nous avons choisi d'éviter

Si on veut changer la fonction de base dans notre feuille de calcul, on doit inscrire cette nouvelle fonction *f(x)* dans la cellule L3C3, puis inscrire la formule « =a**f* (b*(x-h))+k » dans la colonne 2. Or ceci peut s'avérer assez pénible pour une fonction *f(x)* un peu complexe, comme $2x^4 - x^3 + 7x^2 - 13$. On pourrait alors être tenté de procéder autrement pour n'avoir qu'une formule « =f(x) » à recopier. Il suffirait de faire le calcul de *a*f(b*(x-h))+k* en trois étapes comme suit :

- Pas de changements dans la colonne 1, mais on ne donne pas le nom « x » à cette colonne.
- Dans la colonne 2, on calcule *b*(x-h)* via la formule « =b*(LC(-1)-h) ». On donne à cette colonne le nom « x ».
- Dans la colonne 3, on calcule *f(b*(x-h))* via la formule « = *f* (x) ».
- Dans la colonne 4, on calcule *a*f(b*(x-h))+k* via la formule « =a*LC(-1)+k ».

Bien que techniquement correcte, la stratégie utilisée est confuse conceptuellement, puisqu'elle repose sur l'identification de *b*(x-h)* par le nom « x ».

Si on a pris soin d'attribuer le nom ***x*** à la première colonne, la formule définissant les cellules des lignes 12 à 112 de la colonne 2 prendra une forme simple, par exemple

« =a*cos(b*(x-h))+k »

dans le cas où la fonction de base est le cosinus. Comme nous l'avons mentionné précédemment, il faudra changer la formule de toutes ces cellules si la fonction de base change.

Insertion d'un graphique

Nous nous proposons maintenant d'insérer notre graphique. Même si ce n'est pas obligatoire, il est plus simple de sélectionner tout d'abord la plage de cellule L12C1:L112C2 contenant les données pour notre graphique, puis de commander l'insertion d'un graphique en faisant appel à l'assistant graphique.

Comment insérer un graphique dans une feuille de calcul

Excel 2004 pour Macintosh et Excel 2003 pour Windows : à partir du menu « **Insertion »**, choisir l'item « Graphique... » et spécifier les paramètres dans les quatre fenêtres affichées par l'assistant graphique.

Excel 2007 pour Windows : dans le ruban, cliquer sur l'onglet insertion, puis

- **Pour réaliser l'étape1** : dans le groupe « Graphiques », cliquer le graphique désiré. (S'il n'apparaît pas, cliquer sur la flèche, dans le coin en bas à droite et, dans la fenêtre qui apparaît, cliquer sur le graphique désiré.)
  Dans le ruban, les onglets « Création » et « Disposition » apparaissent.
- **Pour réaliser l'étape 2** : cliquer l'onglet « Création » puis, dans le groupe « Données », cliquer le bouton « Sélectionner les données ». La fenêtre qui s'ouvre donne accès aux réglages de l'étape 2.
- **Pour réaliser l'étape 3** : cliquer sur l'onglet « Disposition » : différents boutons donnent accès aux réglages de l'étape 3.
- **Pour réaliser l'étape 4** : cliquer l'onglet « Création » puis choisir « Déplacer le graphique », et la fenêtre de l'étape 4 s'ouvre.

Le nombre de choix proposés par l'assistant graphique nous donne une idée de la diversité des graphiques qu'*Excel* peut produire. Mais comme on pourra modifier ces choix ultérieurement, vous pouvez accepter plusieurs propositions par défaut, en tenant compte des choix illustrés à la figure **4.1.2**. Nous nous contenterons de commenter ici les aspects les plus importants.

Étape 1 La plupart des types de graphiques engendrés par *Excel* considèrent la première coordonnée comme une simple étiquette, qu'on peut même éviter de spécifier (*Excel* fournissant alors une numérotation automatique). Dans notre cas, même si les valeurs en ***x*** sont également espacées, nous voulons que celles-ci soient considérées comme des nombres et non de simples étiquettes. Le type de graphique approprié est donc ici « Nuages de points ». Comme nous voulons relier nos points par des « segments », nous choisissons le sous-type « ... reliés par une courbe sans marquage des données ».

Étape 2 Nous remarquons ensuite qu'*Excel* a bien tenu compte de la plage de cellules sélectionnée pour spécifier les données ou, pour employer le vocabulaire d'*Excel*, les « Valeurs X » et les « Valeurs » (en Y est sous-entendu) de la « Série1 » de données. Mentionnons cependant que, si nous voulions ajouter le graphe d'une autre fonction, il faudrait ajouter ici une nouvelle série de données et préciser les plages de données correspondantes des valeurs en x et en y.

Étape 3 On peut décocher « Afficher la légende ». À cette étape, on pourrait aussi déterminer plusieurs facteurs déterminant l'apparence générale de notre graphique, mais nous préférerons le faire après-coup, pour indiquer comment modifier un graphique déjà fait.

Étape 4 À la fin, on peut spécifier si on veut que notre graphique soit intégré à notre feuille de calcul, ou dessiné sur une nouvelle feuille.

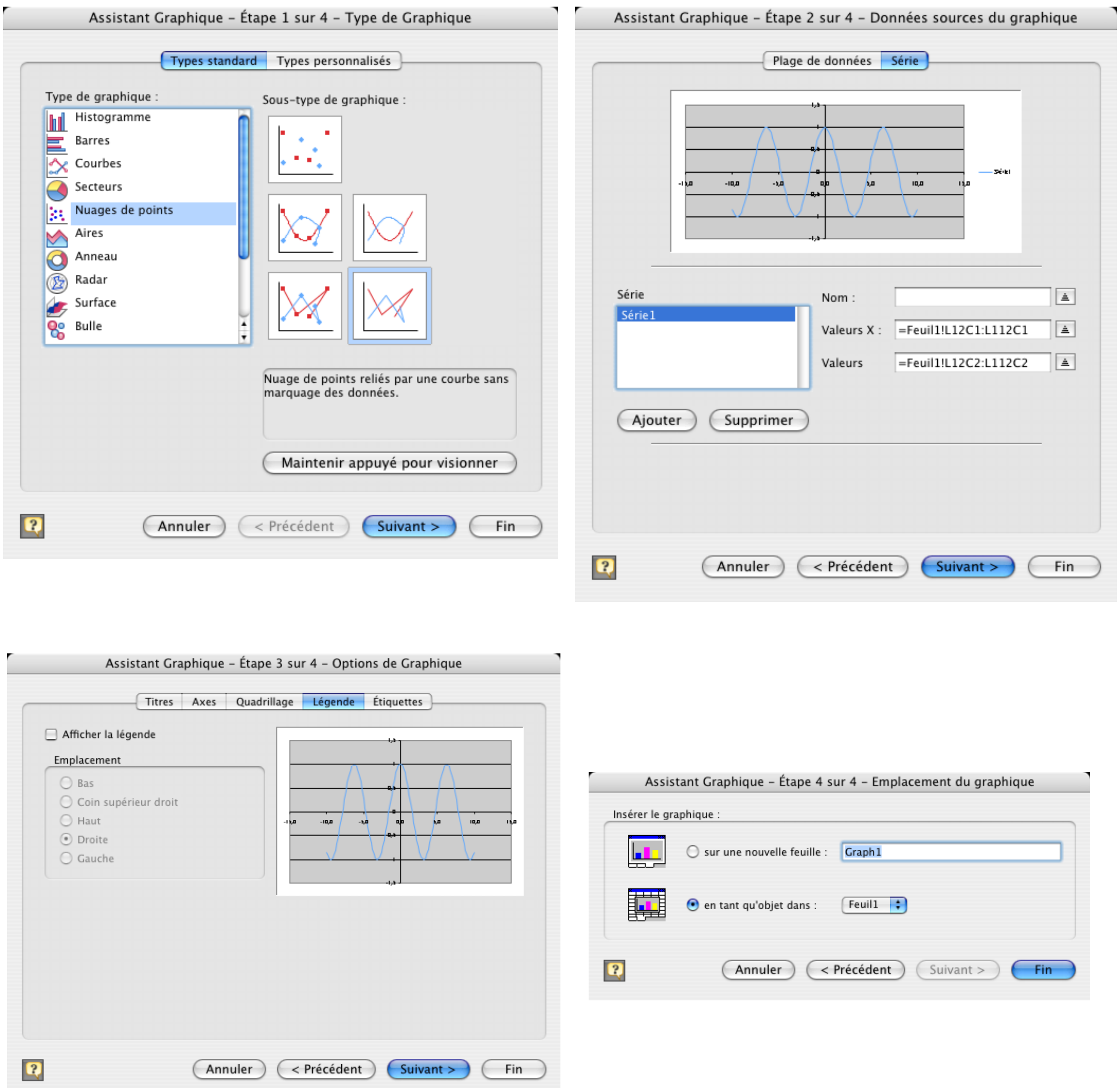


Figure 4.1.2 Les quatre étapes de la définition d'un graphique

À la fin des quatre étapes précédentes, un graphique s'affiche sur notre feuille de calcul. Il est automatiquement sélectionné, mais nous pouvons le désélectionner par un clic en dehors de celui-ci, sur une cellule quelconque de la feuille de calcul.

Quand nous voulons modifier un graphique déjà créé, ou un élément constituant de celui-ci, nous amenons notre curseur-souris au-dessus de l'élément en question, puis nous le sélectionnons par un clic. Soulignons au passage qu'*Excel* nous facilite la tâche en identifiant les éléments survolés par le curseur-souris.

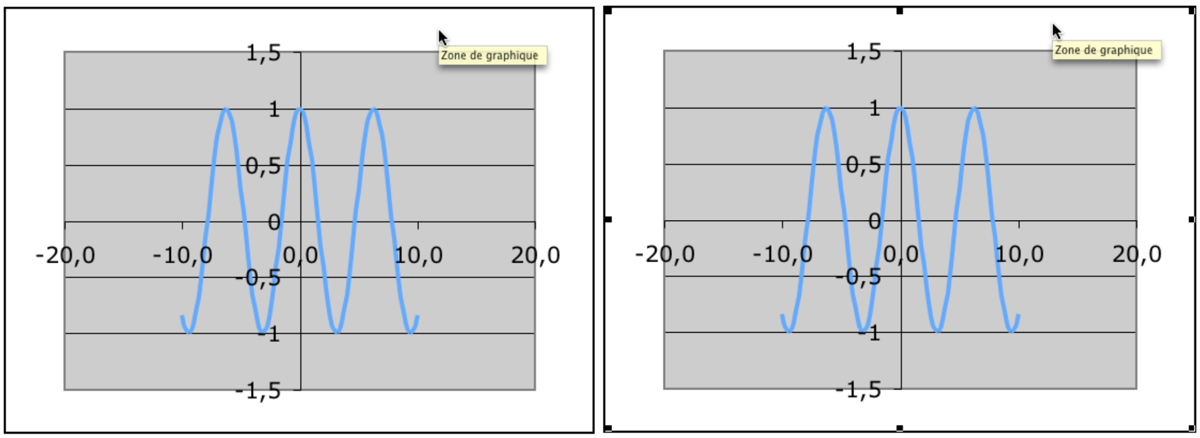


Figure 4.1.3 Survol des composantes d'un graphique, avant (à gauche) et après (à droite) sélection via un clic-souris.

Dans le cas illustré par la figure **4.1.3**, le curseur survole la « Zone du graphique » et un clic à ce moment nous permettra de la sélectionner. Par la suite, nous pourrons déplacer notre graphique ou redimensionner celui-ci par des glisser-souris, tel qu'illustré à la figure **4.1.4**, et le disposer comme nous le désirons dans la feuille.

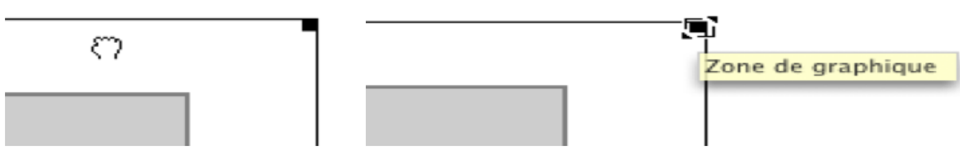


Figure 4.1.4 Une fois la zone de graphique sélectionnée, position du curseur-souris pour déplacer (à gauche) ou redimensionner (à droite).

Un mot sur le redimensionnement : comme nous l'avons constaté dans d'autres contextes, nous pouvons spécifier la contrainte de préserver le rapport entre les dimensions horizontale et verticale en maintenant enfoncée la touche *majuscule* pendant le glisser-souris.

Une fois une composante graphique sélectionnée, on peut faire apparaître un menu local via un *clic-droit*. Dans le cas d'une *Zone de graphique*, ceci donne le choix (entre autres possibilités) de retourner à l'une ou l'autre des quatre étapes nous ayant permis de créer le graphique, de changer son format, ou même de l'effacer.

Pour obtenir un graphique semblable à celui illustré à la figure **4.1.1**, nous pouvons sélectionner puis effacer (via un *clic-droit*) la *Zone de traçage*, ainsi que le *Quadrillage secondaire de l'axe des ordonnées (Y)*. Il ne nous restera plus qu'à spécifier le format des axes : un *clic* (pour sélectionner l'axe) suivi d'un *clic-droit* nous permettra de choisir l'item « Format de l'axe... »[1] dans le menu local, faisant apparaître la fenêtre de dialogue de la figure **4.1.5**

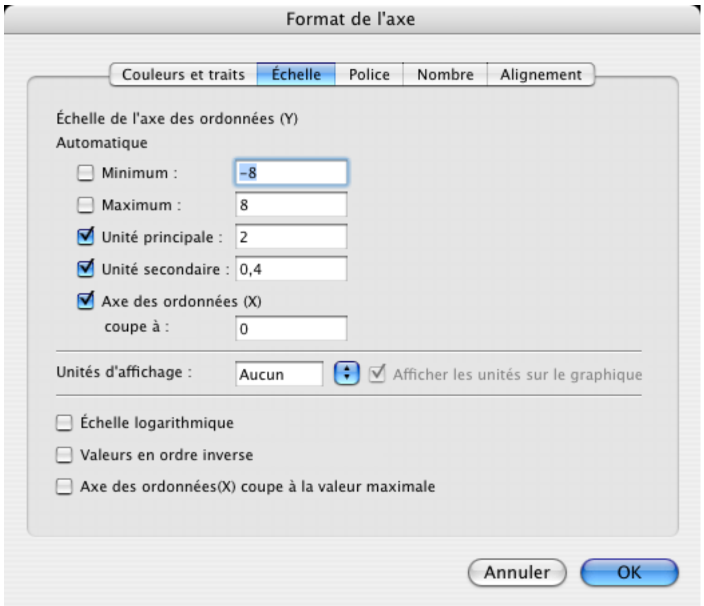


Figure 4.1.5 La fenêtre de dialogue permettant de spécifier le format de notre axe vertical.

Soulignons qu'il est important de bien décocher les deux cases correspondant à l'ajustement automatique du *Minimum* et du *Maximum* de l'axe vertical. Si nous ne le faisons pas, *Excel* pourra les modifier selon les valeurs en *y* obtenues dans la série de données, et la région du plan cartésien représentée dans le graphique pourra varier. Or, si nous voulons visualiser l'effet de la variation des divers paramètres sur les fonctions obtenues, il est important que cette région demeure inchangée pour que notre base de comparaison soit stable...

Jetons maintenant un coup d'œil au résultat obtenu : nous avons réussi à associer un graphique cartésien à la table de valeurs que nous avons construite à partir de notre fonction de base. Si nous tapons de nouvelles valeurs pour les paramètres (dans les cellules L5C3 à L8C3 qui leur correspondent), nous verrons le graphique se modifier presque instantanément. Le tout ressemble un peu à ce qu'on aurait pu obtenir avec une calculatrice graphique. Mais dans *Excel,* nous pouvons ajouter des éléments d'interactivité (les barres de défilement) qui permettront d'atteindre un plus grand dynamisme.

Utilisation des barres de défilement

Nous allons maintenant insérer une barre de défilement pour chacun des quatre paramètres de notre famille de fonctions. Il faut cependant savoir qu'Excel met à notre disposition deux types de barres de défilement (le type « Formulaires » et le type « Contrôles ») qui ont chacun leurs

[1] Nommé « Mise en forme de l'axe … » dans *Excel 2007*.

avantages et leurs inconvénients. Notre première décision sera donc de choisir le type qui nous convient le mieux.

Barres de défilement de type « Formulaires » ou « Contrôles » ?

On fait appel aux barres de défilement pour provoquer, dans une feuille de calcul, des changements interactifs suite aux déplacements gestuels d'un curseur. Malheureusement, ces barres de défilement ont un comportement fort différent selon les versions de Word utilisées :

- Les barres de défilement de type « Formulaires » sont interactives dans toutes les versions d'*Excel* sauf *Excel 2007* pour *Windows*. Dans cette dernière version, les barres de défilement peuvent faire varier interactivement la valeur de cellules, mais leur effet n'est ressenti sur les graphiques qu'une fois le bouton de la souris relâché : elles ne sont donc pas interactives pour les graphiques.
- Les barres de défilement de type « Contrôles » n'existent tout simplement pas dans les versions *Macintosh* d'*Excel*.

En conséquence, il n'est possible de réaliser des feuilles de calcul comportant des barres de défilement interactives fonctionnant dans toutes les versions d'Excel que lorsque ces feuilles ne comportent pas de graphiques interactifs. Dans le cas contraire, il faut choisir les barres de défilement de type

- « Formulaire » si on veut une compatibilité avec le *Macintosh*, en sacrifiant *Excel 2007*
- « Contrôles » si on veut une compatibilité avec toutes les versions *Windows* d'*Excel*, en sacrifiant le *Macintosh*.

C'est l'une des râres situations de ce livre où l'on ne peut réaliser des documents fonctionnant sous toutes les plates-formes : au moins deux versions seront nécessaires.

Pour faire apparaître une barre de défilement de type « Formulaires » dans notre feuille de calcul, il suffit de faire afficher la barre d'outils « Formulaires[1] » (si ce n'est déjà fait), de choisir par un clic l'outil « Barre de défilement » et de délimiter, par un glisser de la souris, le rectangle qui contiendra la barre en question.

Comment faire afficher la barre d'outils « Formulaires »

Excel 2004 pour Macintosh et Excel 2003 pour Windows : à partir du menu « **Affichage** », faire successivement les choix suivants : « Barres d'outils » ➤ « Formulaires ».
Excel 2007 pour Windows : si l'onglet « Développeur » n'apparaît pas dans le ruban, cliquer sur le « Bouton Microsoft Office » puis cliquer sur le bouton « Option Excel ». Cocher « Afficher l'onglet développeur dans le ruban ».
Lorsque l'onglet « Développeur » est visible, cliquer dessus puis dans le groupe « Contrôles » cliquer sur « Insérer » et sélectionner l'outil « Barre de défilement » dans la partie « Formulaires ».

Il nous faut ensuite configurer la barre de défilement que nous venons de créer. Un clic-droit sur la barre fera apparaître un menu local où nous choisirons l'item « Format de contrôle... », ce qui

[1] Nous faisons ce choix afin d'assurer une compatibilité entre les versions Macintosh et Windows. Si vous prévoyez que votre feuille de calcul ne sera utilisée qu'avec la version Windows, l'utilisation d'objets de type « Contrôles » serait un choix plus performant.

affichera la boîte de dialogue de la figure 4.1.6. Nous invitons le lecteur à jeter un coup d'œil aux divers onglets (il découvrira, entre autres choses, comment s'assurer d'imprimer la barre avec la feuille de calcul). Pour notre part, nous nous limiterons ici à l'onglet « Contrôle ». Comme nous pouvons le voir, chaque barre de défilement peut être liée à une cellule de notre classeur, que nous pouvons déterminer par le clavier (en tapant directement sa référence dans la zone appropriée) ou par la souris (par un clic sur l'icône à droite de la zone d'entrée de la cellule liée [ce qui transforme notre fenêtre de dialogue en la figure **4.1.7**], suivi d'une sélection de la cellule voulue, et en terminant par un clic sur l'icône à droite de la zone d'entrée).

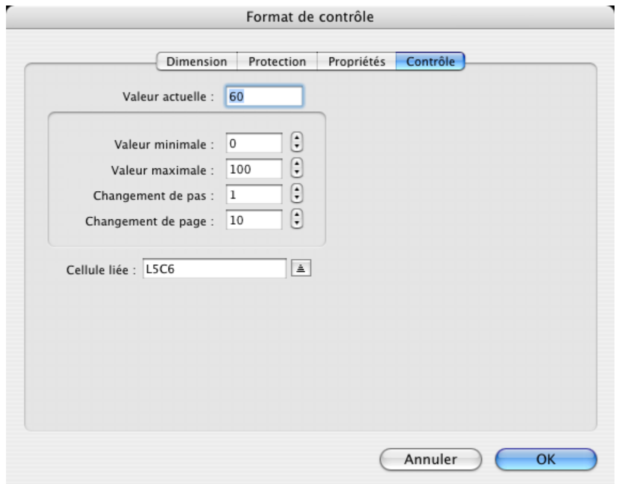


Figure 4.1.6 Configuration d'une barre de défilement.

Une fois ce lien entre la barre de défilement et une cellule établi, *Excel* se chargera de mettre à jour la valeur de la cellule pour chaque changement de l'état de cette barre. Et réciproquement, *Excel* modifiera l'état de cette barre si on change la valeur de la cellule liée.

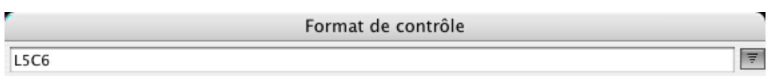
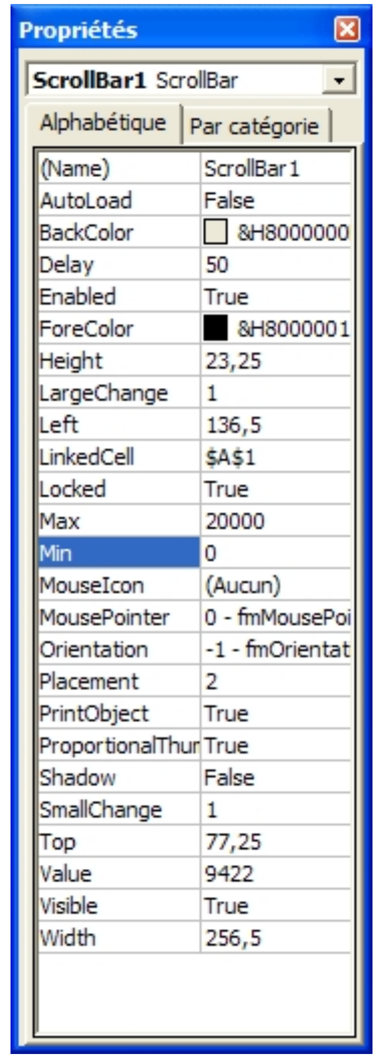


Figure 4.1.7 Pour spécifier la cellule liée avec la souris.

Voyons maintenant comment procéder pour utiliser une barre de défilement de type « Contrôles ». La première étape est de cliquer sur le bouton correspondant, comme indiqué ci-dessous.

Utilisation du bouton pour créer une barre de défilement de type « Contrôles »
Excel 2003 pour Windows : menu « Affichage » ➤ item « Barres d'outils... » ➤ sous-item « Boîte à outils Contrôles » ➤ clic sur le bouton « Barre de défilement ». ***Excel 2007 pour Windows*** : onglet « Développeur » ➤ bouton « Insérer » ➤ clic sur l'icône « Barre de défilement » dans le bloc « Contrôles ActiveX »



Après avoir cliqué sur ce bouton, on remarque que le bouton « Mode Création » (situé dans la barre d'outils de *Word 2003* et dans l'onglet « Développeur » de *Word 2007*) est activé. À l'aide d'un *glisser* de la souris, on détermine le rectangle correspondant à la barre de défilement désirée.
Lorsque la barre est dessinée, on fait un clic droit sur celle-ci pour faire apparaître un menu local, dans lequel on choisit l'item « Propriétés », ce qui fait apparaître la fenêtre ci-contre à droite. Il faut alors donner des valeurs aux paramètres suivants (les noms des paramètres correspondants pour les barres de type « Formulaires » apparaissent entre parenthèses) :

- « LinkedCell » (« Cellule liée »)
  Il n'est pas possible ici de désigner par un clic la cellule visée : il nous faut donc taper la référence (absolue) de la cellule en question. Notons aussi que, dans l'exemple ci-contre, nous avons tapé « L1C1 », mais qu'*Excel* a automatiquement changé le style de référence en « A1 ».
- « Min » (« Valeur minimale »)
- « Max » (« Valeur maximale »)
- « SmallChange » (« Changement de pas »)
- « LargeChange » (« Changement de page »).

Nous ne dirons rien des autres paramètres, dont nous ne ferons pas usage ici. Le lecteur intéressé pourra consulter l'aide d'*Excel* pour obtenir plus de renseignements. Quand tous les paramètres ont reçu les valeurs appropriées, nous fermons la fenêtre « Propriétés » et nous quittons le mode création par un clic sur le bouton « Mode Création » (dont le nom est devenu « Désactiver le mode création » dans *Word 2003*)[1].

À partir de ce moment, *Excel* pourra utiliser la barre de défilement pour modifier dynamiquement la valeur de la cellule liée.

Quel que soit le type de barre de défilement utilisé, les valeurs associées à celle-ci seront en général des entiers compris entre 0 et 30000, et on peut restreindre encore ces valeurs via les cases « Valeur minimale » et « Valeur maximale ». Cette restriction à des valeurs entières non-négatives ne nous convient pas toujours. Ainsi, dans notre cas, supposons que nous voulions laisser varier les valeurs de nos paramètres entre -10 et 10, tout en permettant de les contrôler via une barre de défilement. Il suffira alors de *calculer* la valeur du paramètre à partir de la valeur associée à la barre de défilement : par exemple, si la valeur associée à la barre est comprise entre 0 et 100 et est placée dans une cellule liée nommée « barre », la formule « =-10+barre/5 » produira des valeurs entre -10 et 10. (Pourquoi?)

[1] Si, plus tard, on veut apporter des modifications à une barre de défilement de type « Contrôles », il faudra en tout premier lieu se remettre en mode création, par un clic sur le bouton correspondant, avant de pouvoir faire un clic droit sur celle-ci pour obtenir le menu local.

De cette façon (voir figure **4.1.1**), on pourra lier les cellules des lignes 5 à 8 de la colonne 6 aux barres de défilement correspondantes, tandis que les cellules des lignes 5 à 8 de la colonne 3 seront calculées à partir des cellules correspondantes de la colonne 6. Si on le désire, on peut ensuite « cacher » les cellules de la colonne 6 : il suffit de spécifier que la couleur du caractère de ces cellules doit coïncider avec la couleur du fond. On pourrait aussi masquer toute la colonne 6.

La feuille de calcul que nous venons de compléter ajoute une dimension dynamique au résultat obtenu précédemment. Le graphique peut maintenant être contrôlé interactivement via les quatre barres de défilement, ce qui nous permet de visualiser de façon très convaincante le rôle de chacun des paramètres de notre famille de fonctions : amplitude, fréquence, déphasage, décalage vertical. Cependant, en explorant la situation avec toutes les variations permises par la feuille de calcul, vous pourrez constater l'apparition occasionnelle de phénomènes problématiques, que nous allons discuter dans la prochaine section.

4.2 Les graphiques dans Excel : problèmes et solutions

Nous avons donc pu constater la présence de divers problèmes des représentations graphiques fournies par notre feuille de calcul. Certains de ces problèmes sont purement techniques et liés au fonctionnement particulier d'*Excel*. D'autres, au contraire, sont de nature conceptuelle et il faudra chercher leurs sources dans le mécanisme général de représentation d'objets mathématiques (ayant souvent un caractère infini) dans un contexte technologique (où les ressources disponibles sont bien évidemment finies).

Un problème technique d'*Excel* et sa surprenante solution[1]

Chose curieuse, ce problème ne se manifestera que pour certaines positions de notre graphique dans la feuille de calcul. La figure **4.2.8** illustre l'une de ses manifestations.

Nous déplaçons le curseur de la glissière associée au paramètre h, et la fonction se modifie interactivement sous nos yeux, à l'exception de la tranche du haut qui demeure inchangée jusqu'au moment où nous relâchons le curseur. (Chose curieuse, dans la version *Macintosh* d'Excel, le curseur lui-même ne se déplace pas avec la souris. Mais nous ne connaissons pas le moyen de « réparer » ce comportement.)

De plus, si nous utilisons la barre de défilement horizontale de la fenêtre de notre classeur *Excel* pour modifier la région visible, de façon à cacher les deux premières colonnes, nous aurons une autre surprise : quand nous ferons glisser le curseur des barres de défilement des paramètres, nous constaterons que tout le graphique demeure inchangé jusqu'au moment où nous relâchons le curseur. Comment expliquer, et surtout corriger, ce comportement?

[1] Rappelons que, si vous prévoyez que votre feuille de calcul ne sera utilisée qu'avec *Windows*, il est préférable d'utiliser les barres de défilement de type « Contrôles », qui n'éprouvent pas le problème de mise à jour décrit ici.

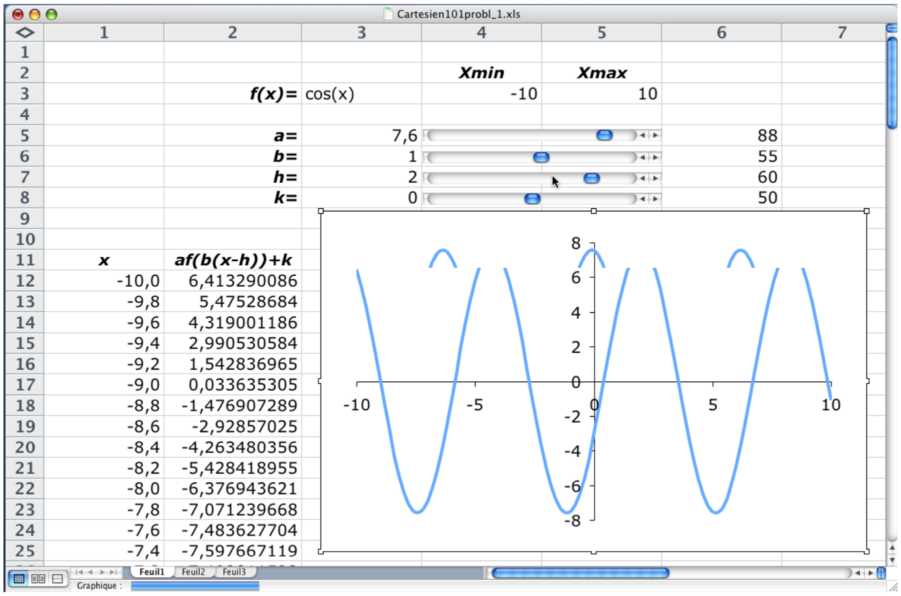


Figure 4.2.8 Problème de mise à jour dans *Excel*.

D'autres expérimentations nous amènent à émettre la conjecture suivante : *Excel* ne met à jour que les portions du graphique qui se situent au même niveau (horizontalement) que des cellules **visibles** qui sont mises à jour. Si cette hypothèse est vérifiée, le seul moyen de s'assurer que notre graphique soit toujours mis à jour est de s'assurer que toutes les cellules sous celui-ci soient continuellement mises à jour. Ce sera le cas, par exemple, si elles contiennent la formule « =alea() », qui place un nombre aléatoire entre 0 et 1 dans la cellule. De plus, si on veut éviter que ces nombres soient visibles et distraient l'attention de l'utilisateur, on peut les écrire de la même couleur que celle du fond de la cellule.

On peut réaliser le plan décrit au paragraphe précédent et constater que tous nos problèmes de mise à jour sont alors résolus. Prenons maintenant un peu de recul : notre mésaventure souligne le fait que, peu importe les théories qu'on peut avoir sur le fonctionnement de nos progiciels, nous devons nous adapter à leurs comportements réels et agir en conséquence.

Un problème de représentation informatique d'objets mathématiques

En variant les divers paramètres de notre famille de fonctions, nous arrivons à la figure **4.2.9**, qui semble indiquer à première vue que les maxima successifs atteints par la fonction ne sont pas tous égaux. Or nos connaissances mathématiques nous disent que ce n'est pas le cas : tous les maxima de la fonction $f(x) = 7{,}8\cos(10x)$ sont égaux à 7,8; de même, tous les minima sont égaux à -7,8. Est-ce qu'*Excel*, encore une fois, est dans l'erreur?

Cette fois, *Excel* n'y est pour rien : la faute nous en revient entièrement. Il faut remonter au moment où nous avons choisi d'utiliser 101 points, reliés les uns aux autres, pour représenter le graphe de nos fonctions. Dans l'intervalle [-10 , 10] et avec la fonction de base $f(x) = \cos(x)$, cela suffit largement : dans cet intervalle de longueur 20, notre fonction aura $\dfrac{20}{2\pi} \approx 3{,}2$ périodes, et nous disposerons d'environ $\dfrac{101}{3{,}2} \approx 31{,}6$ points pour représenter chaque période. Mais la fonction $f(x) = 7{,}8\cos(10x)$ en aura dix fois plus, soit près de 32 périodes : une moyenne d'environ $\dfrac{101}{32} \approx 3{,}2$ points par période semble nettement insuffisant pour obtenir une représentation graphique adéquate...

La solution est évidente : il nous faut utiliser un plus grand nombre de points pour tracer notre graphique. Dans l'exercice 5, nous vous demanderons de modifier notre feuille de calcul pour tracer nos graphiques avec 1001 points plutôt que 101 : la figure **4.2.10** montre le résultat obtenu. Cette figure montre aussi les 101 points utilisés pour tracer le graphe de la figure **4.2.9** et nous aide à mieux comprendre ce qui s'est passé dans ce cas.

En général, il n'existe pas de façon automatique de déterminer le nombre de points adéquat pour obtenir une bonne représentation graphique des fonctions. Les meilleurs logiciels permettent à l'utilisateur de spécifier le nombre de points : en fin de compte, c'est l'utilisateur qui doit avoir le discernement et l'expérience mathématique nécessaire pour faire de bons choix.

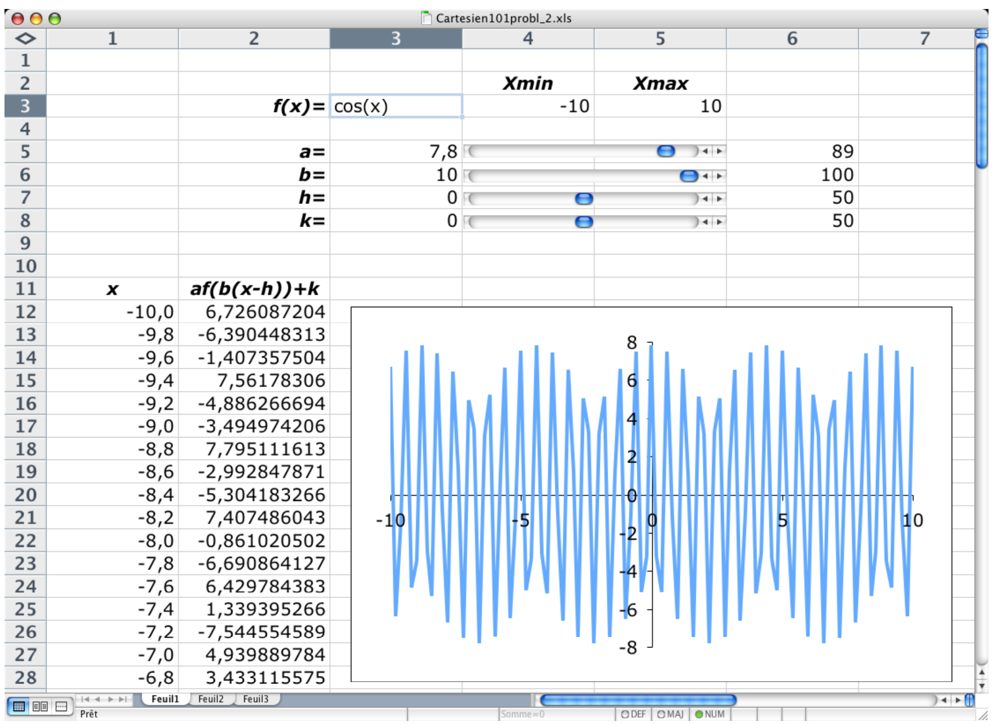


Figure 4.2.9 Problème avec le graphe de la fonction f(x)=7,8 cos(10*x).

En terminant notre discussion de ce traceur de graphes cartésiens, soulignons un phénomène qui semble assez universel en informatique : une versatilité accrue s'accompagne souvent d'une complexité accrue. La versatilité dont nous parlons ici, en comparaison avec les calculatrices graphiques, réside dans la double possibilité d'ajouter une dimension dynamique par l'utilisation de barres de défilement, et de s'assurer de la fidélité des graphiques obtenus par le contrôle du nombre de points utilisés pour tracer. Mais il faut avouer que cette versatilité s'accompagne d'un prix à payer : la création et même l'utilisation de cette feuille de calcul s'avère plus complexe que l'utilisation correspondante d'une calculatrice graphique, par exemple, si on veut changer le domaine de la fonction ou le nombre de points utilisés pour tracer.

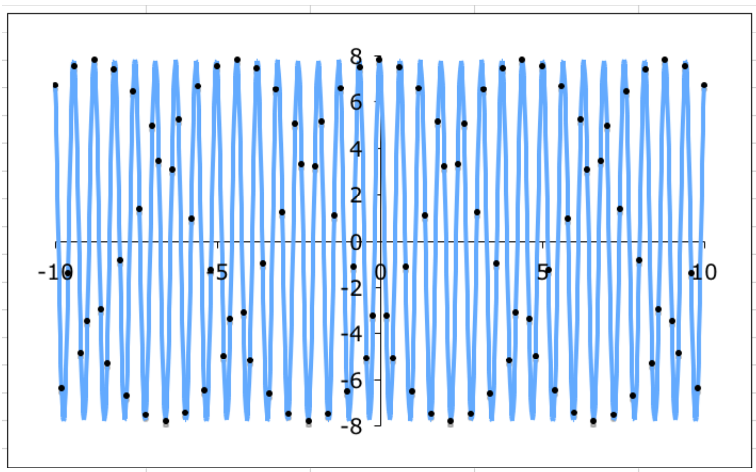


Figure 4.2.10 La même fonction qu'à la figure 4.2.9, mais tracée en utilisant plus de points. Les points ajoutés correspondent aux données utilisées pour tracer la figure 4.2.9.

En utilisant la programmation dans *Excel*, on pourrait automatiser tout ce processus. On pourrait alors envisager le mode de fonctionnement suivant : un expert (par exemple, un professeur) crée une feuille de calcul à la fois versatile et simple à utiliser par des novices (des élèves ou d'autres professeurs, dans notre exemple).

L'impact de la technologie sur les programmes de mathématiques

Comme nous le mentionnions en début de chapitre, l'arrivée des calculatrices graphiques et des ordinateurs dans les écoles secondaires semble avoir coïncidé avec une plus grande place faite à l'étude des paramètres des familles de fonctions. Mais on n'a pas jugé bon d'expliquer aux élèves une partie du fonctionnement des outils technologiques utilisés, ce qui semble pourtant nécessaire pour comprendre des phénomènes tel celui évoqué ci-dessus.

4.3 Produits matriciels et transformations géométriques

Quand on parle de « transformation » dans le contexte de l'enseignement des mathématiques au secondaire, on se réfère à une fonction qui s'applique à une figure géométrique initiale pour en produire une autre, dite figure transformée. Certains sont parfois surpris de constater l'absence d'étapes intermédiaires entre la figure de départ et la figure d'arrivée. Nous allons utiliser *Excel* pour exhiber à quoi pourraient ressembler ces étapes intermédiaires, dans le cas particulièrement simple de rotations autour de l'origine.

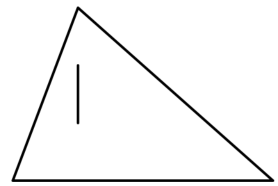
Commençons par un bref rappel mathématique. L'image d'un point *P*, de coordonnées (x, y), par une rotation d'angle θ autour de l'origine est un point *P'* dont coordonnées (x', y') sont données par le produit matriciel suivant :

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} (\cos\theta)x + (-\sin\theta)y \\ (\sin\theta)x + (\cos\theta)y \end{pmatrix}.$$

On peut ainsi trouver l'image de toute figure composée d'une collection de segments : l'image de chaque segment $\overline{PQ}$ sera le segment $\overline{P'Q'}$, où *P'* est l'image de *P* et *Q'* est l'image de *Q*. Notons au passage que si nous voulons transformer plusieurs points de coordonnées (x_1,y_1), … , (x_n,y_n), il suffit de faire une seule multiplication matricielle :

$$\begin{pmatrix} x_1' & \cdots & x_n' \\ y_1' & \cdots & y_n' \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_1 & \cdots & x_n \\ y_1 & \cdots & y_n \end{pmatrix}.$$

Nous avons choisi d'illustrer notre propos en utilisant une figure très simple (voir ci-contre) : un triangle auquel nous avons ajouté un segment pour illustrer comment traiter le cas où l'on doit « lever le crayon » pour tracer la figure. Libre à vous de la remplacer par une figure qui vous conviendrait mieux.



Comme nous le mentionnions précédemment, il est d'usage d'utiliser ces transformations avec un angle θ fixé (par exemple $\theta = 30°$) : on obtient ainsi la figure transformée à partir de la figure initiale. Mais comment obtenir les figures intermédiaires? Tout simplement en permettant à l'angle θ de prendre des valeurs intermédiaires (par exemple de 0° à 30°). Dans *Excel*, ces valeurs intermédiaires pourront être contrôlées par une barre de défilement.

Nous allons donc réaliser la feuille de calcul illustrée à la figure **4.3.11**. Pour ce faire, nous apprendrons comment utiliser les matrices dans *Excel* (nous verrons qu'il est utile dans ce contexte de nommer des plages de cellules), comment dire de « lever le crayon » dans une série de données, et comment déterminer précisément les dimensions de nos graphiques.

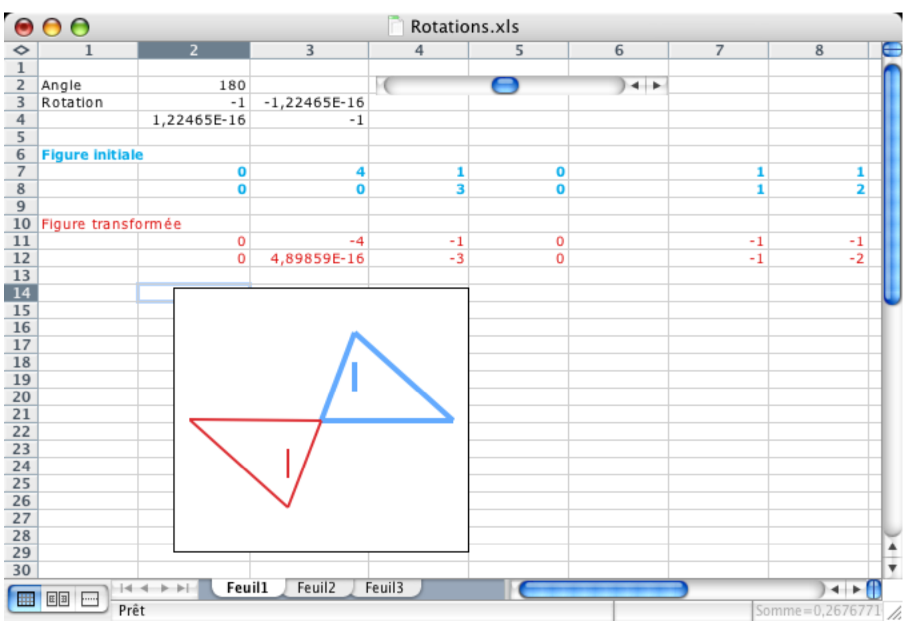


Figure 4.3.11 La figure initiale (en bleu, traits épais et série de données en caractères gras) et la figure transformée (en rouge, traits minces et série de données en caractères normaux) via une rotation contrôlée interactivement par une barre de défilement.

Commençons tout d'abord par ce que nous savons déjà faire. Insérons une barre de défilement qui contrôlera l'angle de rotation (de 0º à 360º) et qui sera associée à la cellule L2C2, que nous nommerons « theta ». La matrice de rotation sera placée dans la plage L3C2 :L4C3 en utilisant les formules montrées à la figure **4.3.12**.

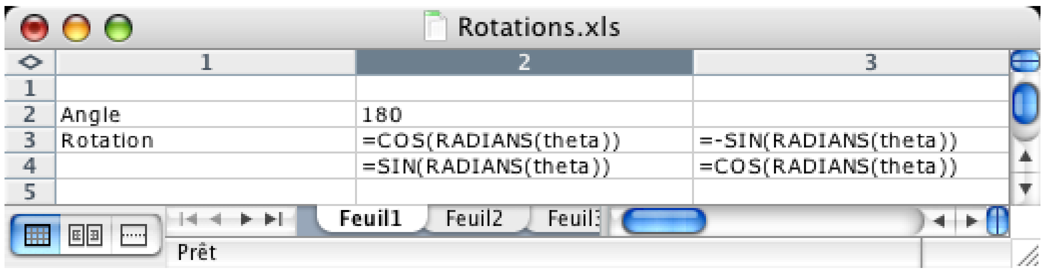


Figure 4.3.12 Les formules servant à définir la matrice de rotation d'un angle theta.

Soulignons que, dans *Excel*, les fonctions trigonométriques supposent que leurs arguments sont exprimés en radians. D'où l'utilisation de la fonction « RADIANS » qui, comme son nom l'indique, fait la conversion des degrés en radians. Nommons « Rot » la plage de cellules contenant cette matrice de rotation.

Les lignes 7 et 8 serviront à la description de la figure initiale, la ligne 7 contenant les coordonnées en x et la ligne 8 les coordonnées en y. Chaque intersection d'une colonne avec ces deux lignes correspondra donc à l'un des sommets qui seront joints pour tracer la figure. Notez la présence d'une colonne vide (la 6) qui sera interprétée comme une indication de « lever le crayon » (ou de cesser de relier les points de part et d'autre de cette colonne vide).

Il serait tentant de nommer « FigureInitiale » la plage L7C2:L8C8 qui lui correspond. Mais nous verrons que la fonction *Excel* qui fait le produit de matrices ne tolère pas de cellules vides comme celles de la colonne 6. Nous diviserons donc notre figure en deux parties : la plage L7C2:L8C5, que nous nommerons « Partie1 », et la plage L7C7:L8C8, que nous nommerons « Partie2 ».

Nous définirons donc la figure transformée via les formules « =ProduitMat(Rot; Partie1) » et « =ProduitMat(Rot; Partie2) ». Mais ce ne sont pas là des formules ordinaires, en ce sens qu'elles ne déterminent pas le contenu d'une cellule individuelle mais bien le contenu d'une plage de cellules. La façon de procéder sera donc différente de ce que nous connaissons. Voyons donc comment procéder pour la première formule :

- On sélectionne d'abord la plage de cellules dont le contenu sera calculé par la formule matricielle (soit L11C2:L12C5 dans ce cas).
- On tape le texte « =ProduitMat(Rot; Partie1) » mais **on ne confirme pas via une des touches habituelles** (« retour » ou « entrée »).
- On confirme via
    - « commande+ retour » ou « commande+ entrée » (Macintosh)
    - « contrôle+majuscule+ retour » ou « contrôle+majuscule+ entrée » (Windows).

Le caractère spécial des formules matricielles est souligné par le fait que, lorsqu'on sélectionne une cellule définie par une telle formule, la formule apparaît entre accolades : dans notre cas, ce sera « {=ProduitMat(Rot; Partie1)} » plutôt que « =ProduitMat(Rot; Partie1) ».

Après avoir fait de même pour la seconde partie de la figure, nous pouvons créer le graphique asssocié aux deux figures. On peut commencer par sélectionner la plage L7C2:L8C8 correspondant à la figure initiale et faire appel à l'assistant graphique (voir figure **4.1.2**) pour créer un graphique de type « Nuages de points » et de sous-type « ... reliés par une courbe sans marquage des données » : ceci nous assure que les points de notre série de données seront reliés, en tenant compte d'occasionnels « levers de crayon ». Nous pouvons ensuite ajouter la seconde figure en tant que nouvelle série de données. Comme nous l'avons vu précédemment, nous pouvons aussi choisir les options de présentation désirées : cacher la légende, zone de traçage, quadrillage et axes (après avoir spécifié une échelle non automatique appropriée).

Notons que nous choisissons la même échelle pour les deux axes, car nous visons à obtenir un graphique montrant une figure qui ne se déformera pas quand on lui fera subir une rotation. Par exemple, si nos deux axes ont un minimum de -4 et un maximum de 4, l'échelle sera géométriquement identique si et seulement si le rectangle contenant notre graphique est un carré. La figure **4.3.13** nous montre ce qu'il advient de nos rotations quand notre zone graphique s'éloigne (exagérément ici) du carré : la figure semble se transformer et non subir une simple rotation.

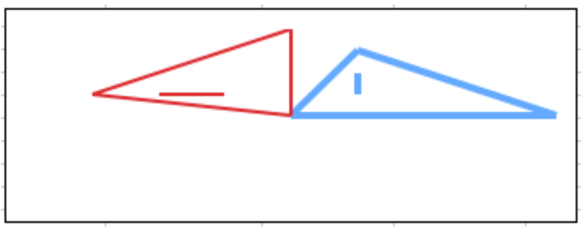


Figure 4.3.13 « Rotation » d'une figure quand les échelles horizontale et verticale sont identiques mais que la zone graphique n'est pas carrée.

La solution de notre problème est bien simple : il faut nous assurer que notre zone graphique soit bien carrée. On pourrait, bien sûr, ajuster ladite zone approximativement, à l'aide d'un glisser-souris. Mais nous visons une précision toute mathématique : comment dire à *Excel* qui nous souhaitons exactement un carré? La solution viendra du fait que notre **graphique** est aussi un **dessin**... Cet énoncé vous paraît peut-être une évidence inutile, mais il ne faut pas oublier que ces deux termes ont un sens précis pour *Excel* : un **graphique** est un type de représentation d'une ou de plusieurs séries de données contenues dans des plages de cellules, tandis qu'un **dessin** est obtenu via une utilisation des outils de dessin (tout comme dans *Word*).

Ce n'est pas encore clair? Pour mieux comprendre, faisons l'expérience suivante (*Excel* 2004 pour *Macintosh* et *Excel* 2003 pour *Windows*) . Sélectionnons notre graphique par un clic-souris puis faisons afficher (par un clic-droit) le menu local associé. Le résultat obtenu apparaît dans la partie gauche de la figure **4.3.14** : le menu local affiché nous propose ici une liste d'actions que nous pouvons appliquer à notre sélection, considérée ici comme un **graphique Excel**. Par un ou deux clics ailleurs dans la feuille de calcul, désélectionnons notre graphique et faisons afficher la barre des outils de dessin (via le menu « Affichage » ➤ « Barres d'outils » ➤ « Dessin »). Faisons d'abord un clic sur l'outil de sélection des objets de cette barre, puis un clic sur notre graphique, suivi d'un clic-droit. Le résultat obtenu apparaît dans la partie droite de la figure **4.3.14** : le menu local affiché nous propose ici une liste d'actions que nous pouvons appliquer à notre sélection, considérée ici comme un **dessin Excel**.

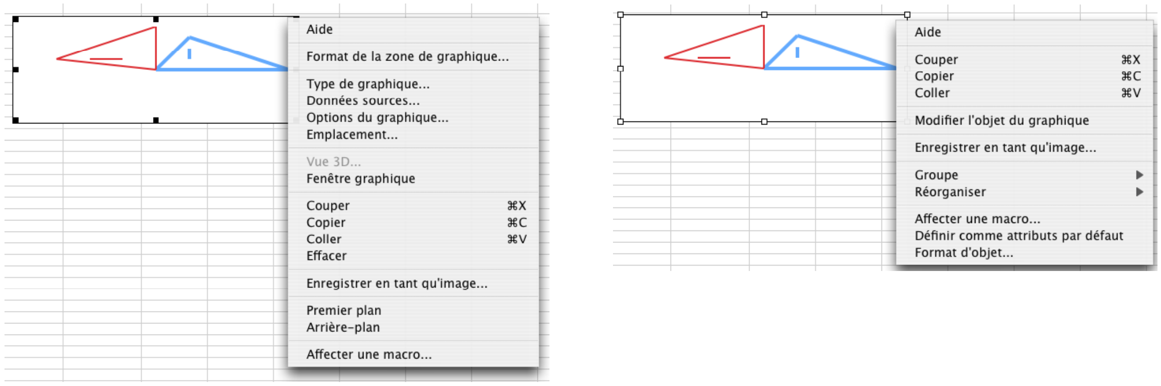


Figure 4.3.14 La double nature « graphique Excel » et « dessin Excel » de nos représentations picturales de séries de données.

Obtenir la zone graphique souhaitée est maintenant facile : à partir du menu local du **dessin Excel**, choisir « Format d'objet » ➤ « Dimensions », puis spécifier une *Hauteur* et une *Largeur* identiques. La dimension choisie importe peu car vous pourrez par la suite redimensionner avec des glisser-souris, mais en vous assurant de bien avoir la touche « majuscule » enfoncée pour que le rapport des dimensions (égal à 1 dans ce cas) soit conservé. À la fin, ne pas oublier de cliquer à nouveau sur l'outil de sélection des objets de la barre des outils de dessin pour le désélectionner.
Nous voici presque arrivés au terme de la construction de notre feuille sur la rotation. Nous pouvons peaufiner un peu la présentation de l'ensemble pour nous assurer que la correspondance entre les séries de données et leurs représentations géométriques soit bien claire. Le graphique étant sélectionné, à partir du menu « Format » ➤ « Cellule...» ➤ « Police »[1], vous pouvez changer la présentation (couleur, style, etc.) des coordonnées d'une figure. D'autre part, par un clic-droit sur la figure elle-même, vous pouvez faire apparaître le menu local associé à une série de données et aller harmoniser ses caractéristiques graphiques (couleur, épaisseur, etc.). Veuillez noter que, bien que la couleur semble une façon efficace pour bien faire ressortir la liaison entre données et figures, certains individus ont de la difficulté à percevoir ce type d'information : d'où l'intérêt d'utiliser aussi d'autres caractéristiques. Dans le cas présent, l'utilisation dynamique de la barre de défilement suffit amplement à distinguer entre la figure initiale (données et graphique restant inchangés) et la figure transformée (données et graphique changeants).

Comment ajuster la taille d'un graphique dans Excel 2007 pour Windows :
Cliquer d'abord sur le graphique pour le sélectionner, cliquer ensuite sur l'onglet « Mise en forme » qui apparaît dans le ruban. Dans le groupe « Taille » on peut maintenant ajuster la hauteur et la largeur comme dans les autres versions d'Excel.

Une dernière remarque. Soulignons que si on modifie les valeurs contenues dans des cellules correspondant à la figure initiale, les figures (initiale et transformée) du graphique s'ajustent immédiatement. Réciproquement, on peut sélectionner (par un premier clic sur la série de données, puis un second sur le point particulier) puis déplacer (horizontalement ou verticalement)

[1] Pour *Excel* 2007 pour *Windows,* à partir du ruban, faire: onglet « Accueil » ➤ groupe « Police ».

un sommet de la figure initiale et voir la cellule de la coordonnée correspondant à la direction sélectionnée se modifier, de même que le point associé de la figure transformée. On peut même tenter de déplacer un sommet de la figure transformée : *Excel* tentera alors (avec notre aide, voir figure **4.3.15**) de déterminer une valeur de la coordonnée correspondante de la figure initiale produisant le déplacement voulu. Dans le cas présent, *Excel* nous confirmera par la suite qu'il a trouvé une solution.

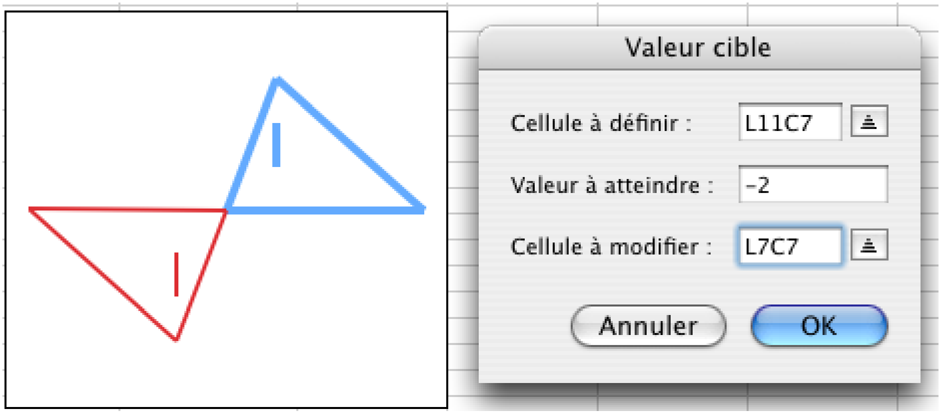
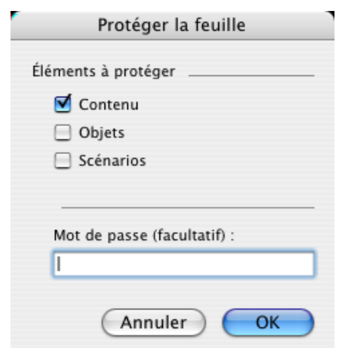


Figure 4.3.15 Quand on essaie de déplacer un point correspondant à une cellule définie par une formule, *Excel* nous demande de désigner une cellule indépendante à modifier pour provoquer le déplacement désiré.

Supposons que nous désirions éviter qu'*Excel* permette de déplacer directement les points de la figure transformée[1]. Il suffirait alors de « protéger » certains éléments de notre feuille de calcul. Si on demande à *Excel* de protéger la feuille de calcul, par défaut tous les éléments seront protégés, et tout changement de l'angle de rotation ou de la figure initiale sera interdit. Aussi notre première étape sera de désigner les cellules ne devant pas être protégées : la cellule L2C2 (correspondant à l'angle de rotation) et la plage L7C2:L8C8 (correspondant à la figure initiale). Pour ce faire, après avoir sélectionné la ou les cellules visées, on fait appel au menu « Format » ➤ « Cellules... » ➤ « Protection » ➤ décocher « Verrouillée »[2]. Nous devons maintenant demander à *Excel* de protéger la feuille de calcul, en faisant appel au menu « Outils » ➤ « Protection » ➤ « Protéger la feuille... »[3], et spécifier ensuite, comme ci-contre, de protéger le contenu mais non les objets : de cette façon, seules les cellules désignées pourront être changées, et on pourra encore manipuler les points de la figure initiale (correspondant à des cellules non protégées),



[1] Comme nous le verrons plus tard, c'est l'usage dans plusieurs logiciels, dont *Cabri-géomètre* : les objets libres peuvent être manipulés, mais on ne peut agir directement sur les objets liés ou dépendants. Ces derniers ne peuvent être modifiés que via une modification des objets libres dont ils dépendent.

[2] Dans Excel 2007 pour Windows, à partir du ruban faire : onglet « Accueil » ➤ « Cellules » ➤ « Format » ➤ « cellule » ➤ « Protection » ➤ décocher « Verrouillée ».

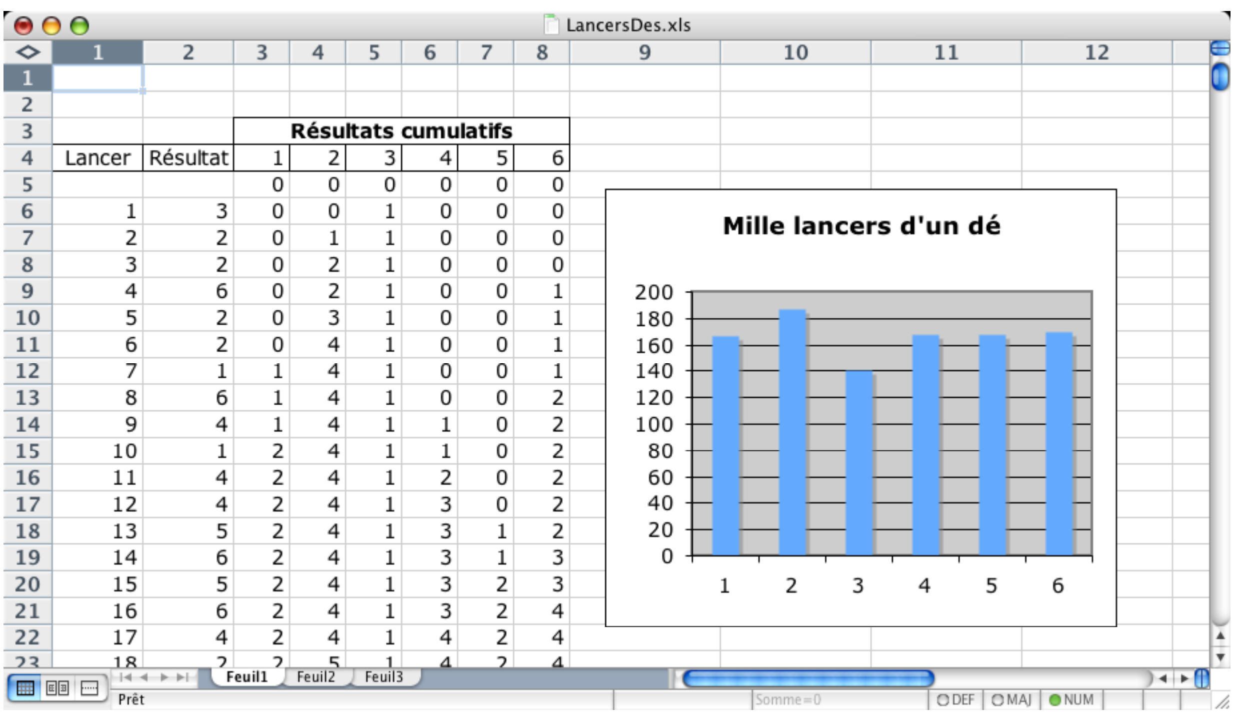
[3] Dans Excel 2007 pour Windows, à partir du ruban faire : onglet « Accueil » ➤ « Cellules » ➤ « Format » ➤ « Protéger la feuille » ➤ cocher seulement « Sélectionner les cellules déverrouillées ».

mais non ceux de la figure transformée (dont les cellules correspondantes sont protégées). Si vous n'avez pas de bonnes raisons de le faire, nous vous suggérons d'éviter d'utiliser des mots de passe : vous pourriez les oublier, et vous empêcheriez les autres d'apprendre en examinant votre travail.

Nous voici donc arrivés au terme de notre exemple sur les rotations. Il vous semble peut-être que nous avons déployé beaucoup d'effort pour un résultat somme toutes assez simple. Mais, chemin faisant, nous avons appris plusieurs techniques (opérations matricielles, calculs trigonométriques en degrés, obtention d'échelles horizontales et verticales identiques, correspondance données-graphiques, protection sélective d'une feuille de calcul) qui pourront être réinvesties dans d'autres contextes. Vous aurez d'ailleurs l'occasion d'appliquer ces techniques dans plusieurs exercices à la fin de ce chapitre.

4.4 Simulations du lancer d'un dé

Nous allons maintenant utiliser *Excel* pour simuler un phénomène aléatoire simple : le lancer répété d'un dé. Notez que nous disposons déjà de toutes les connaissances techniques pour créer une feuille de calcul comme celle illustrée à la figure **4.4.16**, où chaque lancer du dé correspond à une ligne, et où le nombre de lancers (1000 dans le cas présent) est déterminé au moment de la création de la feuille. La réalisation d'une telle feuille de calcul fera d'ailleurs l'objet de l'exercice 12. Dans ce cas, si nous demandons à *Excel* de recalculer la feuille (et nous verrons plus tard comment faire cette demande), les 1000 lancers précédents seront oubliés pour laisser place à 1000 nouveaux lancers, qui seront automatiquement représentés graphiquement.



		Résultats cumulatifs					
Lancer	Résultat	1	2	3	4	5	6
		0	0	0	0	0	0
1	3	0	0	1	0	0	0
2	2	0	1	1	0	0	0
3	2	0	2	1	0	0	0
4	6	0	2	1	0	0	1
5	2	0	3	1	0	0	1
6	2	0	4	1	0	0	1
7	1	1	4	1	0	0	1
8	6	1	4	1	0	0	2
9	4	1	4	1	1	0	2
10	1	2	4	1	1	0	2
11	4	2	4	1	2	0	2
12	4	2	4	1	3	0	2
13	5	2	4	1	3	1	2
14	6	2	4	1	3	1	3
15	5	2	4	1	3	2	3
16	6	2	4	1	3	2	4
17	4	2	4	1	4	2	4
18	2	2	5	1	4	2	4

Figure 4.4.16 Feuille de calcul simulant 1000 lancers d'un dé.

Nous allons plutôt tenter de faire une œuvre plus interactive, comme illustré à la figure **4.4.17**, où le dé lancé sera représenté « graphiquement[1] » et où nous pourrons commander autant de lancers que nous le voudrons. Cette feuille pourrait ressembler à une ressource qu'un professeur voudrait mettre à la disposition de ses élèves. Pendant la création de cette feuille, nous apprendrons

- comment utiliser les outils de dessin dans *Excel* (semblable à ce que nous faisions dans *Word*)
- comment utiliser les cases d'options et les boutons (semblable aux barres de défilement)
- comment écrire une macro en *Visual Basic* (comparaison avec *LangageGraphique*)
- comment *Excel* effectue les calculs nécessaires pour mettre à jour ses feuilles de calcul

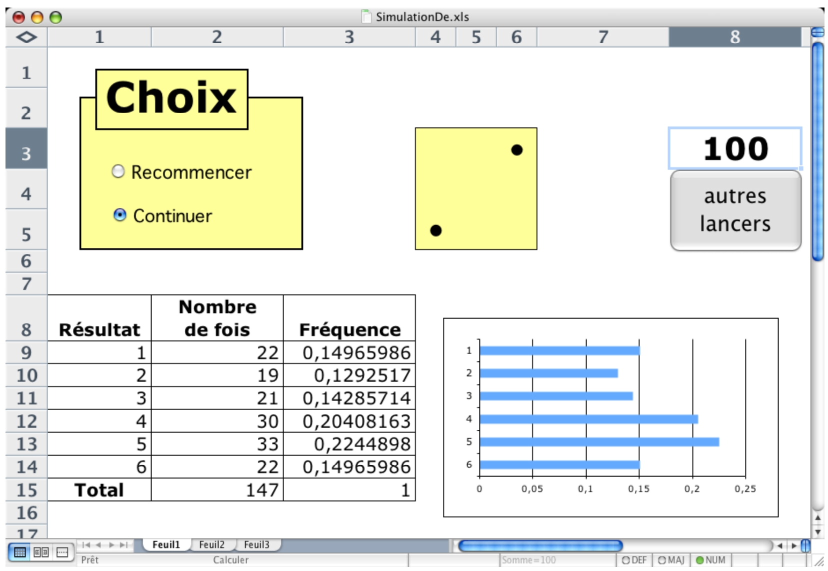


Figure 4.4.17 Feuille de calcul permettant de lancer interactivement un dé.

Au départ, il faut savoir lancer un dé, c'est-à-dire pouvoir engendrer aléatoirement un entier compris entre 1 et 6. Nous connaissons déjà la fonction ALEA(), qui retourne un nombre aléatoire compris entre zéro (inclus) et un (exclus), et la fonction TRONQUE(x) , qui supprime la partie décimale d'un nombre x. Il ne nous en faut pas plus : la formule

« =TRONQUE(6*ALEA()+1) »

calculera le résultat du lancer d'un dé (Pourquoi?). On la place dans la cellule L3C3, que l'on nomme « de ». Notez que cette valeur n'est pas visible dans la figure **4.4.17** car nous avons attribué à cette cellule une couleur de police blanche.

Passons maintenant à la représentation du dé. Pour ce faire, nous avons arbitrairement choisi d'utiliser la plage de cellules L3C4:L5C6, dont nous avons délimité le contour par une bordure et dont nous avons coloré l'intérieur. Nous avons aussi redimensionné les trois colonnes de façon à obtenir un « dé » carré. Il ne nous reste plus qu'à disposer les « points » sur le dé, en fonction du contenu de la cellule « de ». Il y aurait plusieurs façons de procéder : par exemple, on pourrait représenter le résultat obtenu à la figure **4.4.17** en utilisant l'autre diagonale du dé. Nous avons choisi d'utiliser la diagonale de pente positive, et de représenter le résultat « six » par deux colonnes de trois points. Il nous faut donc trouver des formules appropriées pour sept des neuf

[1] En fait, le dé sera composé d'un bloc de neuf cellules dont le contour est encadré par une bordure. Chaque cellule pourra rester vide ou contenir le caractère « • ».

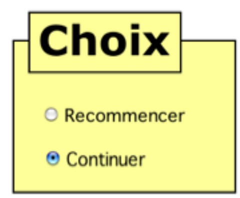
cellules. (Lesquelles?) Nous donnerons deux exemples, en laissant au lecteur le soin de déterminer les cinq autres.

- La cellule L3C6 comportera un point « • » quand le dé vaudra 2, 3, 4, 5 ou 6, et devra rester vide quand le dé vaudra 1 : d'où la formule « =SI(de>1;"•";"") ».
- La cellule L4C5 comportera un point « • » quand le dé vaudra 1, 3 ou 5, et devra rester vide quand le dé vaudra 2, 4 ou 6. On peut utiliser plusieurs formules pour obtenir ce résultat :
  - « =SI(de=1;"•";SI(de=3; "•";SI(de=5; "•";""))) »
  - « =SI(OU(de=1;de=3;de=5);"•";"") »
    Notez la position du « OU », à l'avant et non entre les arguments.
  - « =SI(MOD(de;2)=1;"•";"") »
    Rappelons que « MOD(*a*;*b*) » désigne le reste après division entière de *a* par *b*.
  - « =SI(EST.IMPAIR(de);"•";"") » ou encore « =SI(NON(EST.PAIR(de));"•";"") »
    Notez que les fonctions « EST.PAIR » et « EST.IMPAIR » nécessitent que la macro complémentaire « Utilitaire d'analyse » soit chargée. Pour plus de détails, consultez l'aide *d'Excel*.

On peut maintenant tester le fonctionnement de notre dé en demandant à *Excel* de recalculer plusieurs fois la feuille de calcul. À chaque recalcul, un nouveau nombre aléatoire sera produit par la fonction « ALEA », et toutes les cellules en dépendant seront mises à jour. La représentation visuelle de notre dé sera donc adaptée en conséquence.

Comment commander un recalcul dans *Excel*

Excel 2004 pour Macintosh : par la combinaison de touches « Commande » + « = ».
Excel 2003 et Excel 2007 pour Windows : par la touche « F9 » ou la combinaison de touches « Contrôle » + « = ».



Voyons maintenant comment réaliser la portion de la feuille représentée ci-contre. À l'aide de la barre d'outils « Dessin »[1], dont le fonctionnement est presque identique à celui de la barre correspondante dans *Word*, nous dessinons d'abord un rectangle coloré, puis nous ajoutons une zone de texte qui contiendra le mot « Choix »[2]. Ensuite, utilisant cette fois la barre d'outils « Formulaires »[3], nous ajoutons successivement deux « cases d'options », que nous renommons « Recommencer » et « Continuer ».

Quand nous cliquons en dehors de nos cases d'option pour indiquer que nous avons fini de les définir, nous constatons que celles-ci sont fonctionnelles. Mais il nous reste à compléter une

[1] Dans *Excel 2007*, on y accède via l'onglet « Insertion » ➤ groupe « Illustrations » ➤ bouton « Formes ».
[2] De façon alternative, nous aurions pu remplacer ce rectangle et cette zone de texte par une « Zone de groupe », obtenue par la barre d'outils « Formulaires ». Nous avons choisi une autre voie, qui nous permet plus de variété pour la présentation visuelle. Mais les zones de groupes deviennent obligatoires quand nous voulons utiliser plusieurs groupes indépendants de cases d'options.
[3] Dans *Excel 2007*, on y accède via l'onglet « Développeur » ➤ groupe « Contrôles » ➤ bouton « Insérer » ➤ choix dans « Contrôles de formulaire ».

étape, qui permettra d'informer la feuille de calcul de la case d'option choisie via une cellule qui sera liée à toutes nos cases d'options (puisque nous n'avons ici qu'un groupe de deux cases d'options). La procédure est semblable à celle que nous avons vue pour les barres de défilement : clic-droit sur l'objet, choix de l'item « Format de contrôle... », et désignation (dans l'onglet « Contrôle ») de la cellule liée. Cette dernière contiendra la valeur 1 si la première case d'option (« Recommencer » dans notre cas) est sélectionnée, et 2 dans le cas contraire. Nous avons désigné L5C3 comme cellule liée, lui avons donné le nom de « choix », et spécifié que sa couleur de police devait être blanche. Nous pourrons donc répercuter l'option spécifiée dans notre feuille de calcul à l'aide de formules pouvant ressembler à

« =SI(choix=1; *valeur_si_Recommencer*; *valeur_si_Continuer*) ».

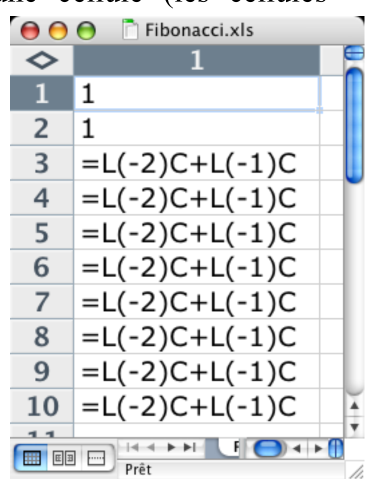
Résultat	Nombre de fois	Fréquence
1	22	0,14965986
2	19	0,1292517
3	21	0,14285714
4	30	0,20408163
5	33	0,2244898
6	22	0,14965986
Total	147	1

Nous voulons maintenant compléter le tableau ci-contre, occupant la plage L8C1:L15C3 de notre feuille de calcul. Cela nous conduira à la mise en évidence du concept de référence circulaire dans un tableur. Mais avant, voyons un peu comment calcule *Excel*.

Comment *Excel* parvient à s'y retrouver dans ses calculs

À priori, on peut distinguer deux types de cellules dans *Excel* : les cellules indépendantes, et les cellules dépendantes. Les cellules indépendantes peuvent contenir des constantes (nombres, textes, etc.) ou être définies par des formules ne faisant pas appel à d'autres cellules (comme « =1/3 », « =PI() » ou « =ALEA() »). Les cellules dépendantes, de leur côté, sont forcément définies par des formules faisant appel à une ou plusieurs autres cellules : « =L(-2)C+L(-1)C » et « =COS(x) » (où « x » est le nom d'une autre cellule) par exemple.

Nous allons raffiner quelque peu ce concept de cellules indépendantes ou dépendantes en définissant ce que nous appellerons le *niveau de dépendance* d'une cellule (les cellules indépendantes ayant un niveau de dépendance égal à zéro). Pour rendre ceci plus clair, nous procéderons tout d'abord sur un exemple : le calcul des nombres de Fibonacci. Voyons comment attribuer un niveau de dépendance aux cellules visibles ci-contre :



- les cellules L1C1 et L2C1 seront de niveau 0
  (car elles sont indépendantes)
- la cellule L3C1 sera de niveau 1
  (car elle dépend de cellules de niveau 0)
- la cellule L4C1 sera de niveau 2
  (car elle dépend d'une cellule de niveau 0 et d'une cellule de niveau 1)
- la cellule L5C1 sera de niveau 3
  (car elle dépend d'une cellule de niveau 1 et d'une cellule de niveau 2)
- etc.

En général, le niveau de dépendance d'une cellule sera d'un de plus que le maximum des niveaux des cellules dont elle dépend. Quand vient le moment de recalculer une feuille, *Excel* doit obligatoirement recalculer une cellule donnée *après* avoir recalculé toutes les cellules dont celle-ci dépend. (Pourquoi?) Bien que nous ne sachions pas exactement dans quel ordre *Excel* recalcule ses cellules (car on peut imaginer plusieurs variations de la procédure décrite ci-dessous), nous pouvons affirmer que l'algorithme suivant produit les mêmes résultats finaux :

- on recalcule d'abord les cellules dont le niveau de dépendance est de 0
- puis on recalcule les cellules dont le niveau de dépendance est de 1
- puis on recalcule les cellules dont le niveau de dépendance est de 2
- puis on recalcule les cellules dont le niveau de dépendance est de 3
- etc.

Les références circulaires

Dans ce qui précède, nous avons supposé qu'on pouvait attribuer un niveau de dépendance à toutes les cellules : or ce n'est pas toujours le cas. Illustrons ceci en modifiant légèrement l'exemple des nombres de Fibonacci : supposons maintenant que le contenu de la cellule L2C1 soit défini par la formule « =L2C1+1 ». En appliquant notre définition de *niveau de dépendance* (dénoté ici par une fonction que nous appellerons *nd*) à la cellule L2C1, comme celle-ci dépend d'elle-même, on devrait avoir

$$nd\,(\mathrm{L2C1}) < nd\,(\mathrm{L2C1})$$

ce qui est évidemment impossible.

De même, si la cellule L2C1 était définie par la formule « =L3C1+1 », on devrait avoir :

$$nd\,(\mathrm{L2C1}) < nd\,(\mathrm{L3C1}) \quad \text{(car L3C1 dépend de L2C1)}$$
$$nd\,(\mathrm{L3C1}) < nd\,(\mathrm{L2C1}) \quad \text{(car L2C1 dépend de L3C1)}$$

ce qui produit une chaîne d'inégalités contradictoires

$$nd\,(\mathrm{L2C1}) < nd\,(\mathrm{L3C1}) < nd\,(\mathrm{L2C1}).$$

Plus généralement, si la cellule L2C1 était définie par la formule « =L*n*C1+1 », où *n* est un entier supérieur à 2, on devrait avoir la suite contradictoire d'inégalités

$$nd\,(\mathrm{L2C1}) < nd\,(\mathrm{L3C1}) < nd\,(\mathrm{L4C1}) < \ldots < nd\,(\mathrm{L}n\mathrm{C1}) < nd\,(\mathrm{L2C1})$$

puisque chaque cellule dépend de la cellule la précédent dans la chaîne d'inégalités. On en conclut donc qu'on ne peut attribuer un niveau de dépendance à chacune des cellules en cause dans les exemples précédents.

De telles situations, où nous avons *m* cellules (que nous désignerons par $C_1, \ldots, C_m$) telles que C_{i+1} dépend de C_i ($i=1, \ldots, m-1$) et C_1 dépend de C_m, sont appelées des références circulaires. Habituellement, la présence de références circulaires est signe d'une erreur de l'utilisateur, et *Excel* nous avertit et refuse de telles références. En effet, sa procédure usuelle de calcul (décrite ci-dessus) ne peut être utilisée dans de tels cas car on ne peut définir le niveau de dépendance de toutes les cellules, puisque cela conduirait aux inégalités contradictoires suivantes :

$$nd\,(C_1) < nd\,(C_2) < \ldots < nd\,(C_m) < nd\,(C_1).$$

Mais il peut arriver qu'on veuille permettre certaines références circulaires. Dans l'exemple de la simulation d'un dé, il est tout à fait raisonnable de souhaiter décrire le contenu de la cellule L9C2 par « si le résultat du lancer du dé est 1, alors augmenter de 1 la valeur de la cellule, sinon garder la même valeur », ce qui peut s'exprimer par la formule « =SI(de=1;LC+1;LC) », où la référence relative « LC » (qui équivaut, rappelons-le, à « L(0)C(0) ») indique clairement qu'il s'agit d'une référence circulaire.

Bien évidemment, si on demande à *Excel* d'accepter la présence de références circulaires, il faudra que celui-ci utilise une autre procédure de calcul : on recalculera tout d'abord les cellules ayant un niveau de dépendance bien défini, puis ensuite les cellules impliquées dans une référence circulaire, dans l'ordre suivant

- on recalcule d'abord toutes les cellules de la ligne 1, en procédant de la gauche vers la droite
- puis on recalcule toutes les cellules de la ligne 2, en procédant de la gauche vers la droite
- puis on recalcule toutes les cellules de la ligne 3, en procédant de la gauche vers la droite
- puis on recalcule toutes les cellules de la ligne 4, en procédant de la gauche vers la droite
- etc.

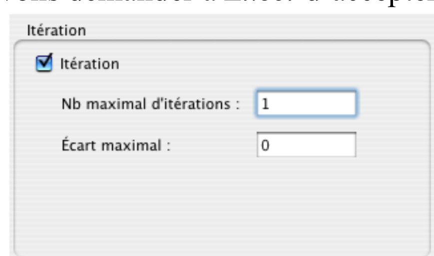
Notons au passage une conséquence dérangeante de l'utilisation de références circulaires : on ne peut plus être certain que les valeurs des cellules soient conformes aux formules qui les définissent. Ainsi, considérons encore la cellule L9C2 de l'exemple des lancers de dés, dans le cas où le résultat obtenu en lançant le dé est 1 : si la formule décrivant la valeur de L9C2 était vérifiée, on devrait avoir

$$valeur(\text{L9C2}) = valeur(\text{L9C2}) + 1$$

ce qui est évidemment impossible. Il est donc recommandé d'être plus attentif que d'habitude quand on utilise des références circulaires : il faut bien connaître toutes les conséquences de ce qu'on fait. Nous verrons d'ailleurs que nous ne sommes pas à l'abri de surprises quand nous poursuivrons notre feuille de calcul simulant des lancers d'un dé.

Retour à la simulation du lancer d'un dé

Suite à la discussion précédente, nous comprenons que nous devons demander à *Excel* d'accepter les références circulaires (voir encadré). Il suffit de cocher la case « Itération » du panneau ci-contre. Mais il faut encore préciser deux paramètres : le nombre maximal d'itérations et l'écart maximal. Nous avons vu qu'en mode itération, *Excel* recalcule toutes les cellules ligne par ligne, de gauche à droite. Et comme les valeurs des cellules ne sont pas toujours conformes aux formules utilisées pour les définir, il est parfois utile de demander à *Excel* de faire plusieurs itérations (c'est-à-dire de recommencer plusieurs fois le recalcul de toutes les cellules de la feuille). Les deux paramètres ci-dessus indiquent à *Excel* quand arrêter : dès qu'au moins un des deux événements suivants se produit



- le nombre maximal d'itération est atteint
- l'écart entre les valeurs des cellules avant et après le recalcul reste (pour toutes les cellules) inférieur à l'écart maximal spécifié.

Comment demander à *Excel* de permettre les références circulaires

Excel 2004 pour Macintosh : à partir du menu « Excel » choisir l'item « Préférences... », puis « Calcul » et compléter le panneau « Itération » comme expliqué dans le texte.

Excel 2003 pour Windows : à partir du menu « Outils » choisir l'item « Options... », cliquer l'onglet « Calcul », cocher « Itération » et renseigner les champs « Nb maximal d'itérations » et « Écart maximal » comme expliqué dans le texte.

Excel 2007 pour Windows : cliquer le bouton « Office » ➤ cliquer le bouton « Options Excel » ➤ choisir « Formules » puis cocher « Activer le calcul itératif» et renseigner les champs « Nb maximal d'itérations » et « Écart maximal » comme expliqué dans le texte.

Dans le cas de notre simulation, nous donnons au nombre maximal d'itérations la valeur 1, et à l'écart maximal la valeur 0 : à chaque fois qu'on demandera un recalcul à *Excel*, il y aura donc exactement une itération. On pourrait être tentés de spécifier un grand nombre d'itérations, 1000 par exemple, pour simuler 1000 lancers de notre dé[1]. Mais il faut savoir qu'*Excel* ne recalcule les fonctions « ALEA() » que lors de la première itération (peut-être pour pouvoir mieux tenir compte des écarts) : dans notre cas, ceci reviendrait à compter 1000 fois un seul lancer de dé, ce qui n'est pas du tout le résultat désiré. Si nous voulons lancer notre dé 1000 fois, il faudra donc commander à *Excel* 1000 recalculs de la feuille, ce qui n'est pas très agréable si nous devons procéder manuellement. Heureusement, on pourra utiliser des méthodes alternatives.

Un problème avec les références circulaires, et sa solution

Quand on enregistre un classeur *Excel*, les préférences (mode itération activé ou non, style de références A1 ou L1C1, affichage des valeurs ou des formules, etc) sont aussi sauvegardées avec le document. Par la suite, quand on ouvre le classeur, les préférences qui lui sont associées sont automatiquement réactivées, du moins dans la plupart des cas. Dans sa version pour *Macintosh*, il arrive parfois qu'*Excel* omette de réactiver certaines préférences, dont le mode *itération*, avec pour conséquence un mauvais fonctionnement (avec message d'erreur). Il suffit alors d'aller cocher manuellement la case « Itération » pour régler le problème.

La figure **4.4.18** montre les formules utilisées pour définir le tableau. Notons tout d'abord que nous avons attribué les noms « resultat » à la colonne 1, « nombre » à la colonne 2, et « total » à la cellule L15C2. Soulignons aussi que le cas « choix=1 » correspond au choix de la case d'option « Recommencer » : les cellules des colonnes 2 et 3 du tableau prennent alors toutes la valeur 0. Remarquons au passage les références circulaires utilisées pour définir les six cellules de la colonne 2 : ceci est conforme à notre discussion précédente. La seule surprise, très troublante de prime abord, vient de l'utilisation de l'expression « nombre / (total+1) » pour calculer la fréquence : on se serait plutôt attendu à la formule traditionnelle « nombre / total ». Que se passe-t-il?

SimulationDe.xls

	1	2	3
8	**Résultat**	**Nombre de fois**	**Fréquence**
9	1	=SI(choix=1;0;SI(de=resultat;nombre+1;nombre))	=SI(choix=1;0;nombre/(total+1))
10	=1+L(-1)C	=SI(choix=1;0;SI(de=resultat;nombre+1;nombre))	=SI(choix=1;0;nombre/(total+1))
11	=1+L(-1)C	=SI(choix=1;0;SI(de=resultat;nombre+1;nombre))	=SI(choix=1;0;nombre/(total+1))
12	=1+L(-1)C	=SI(choix=1;0;SI(de=resultat;nombre+1;nombre))	=SI(choix=1;0;nombre/(total+1))
13	=1+L(-1)C	=SI(choix=1;0;SI(de=resultat;nombre+1;nombre))	=SI(choix=1;0;nombre/(total+1))
14	=1+L(-1)C	=SI(choix=1;0;SI(de=resultat;nombre+1;nombre))	=SI(choix=1;0;nombre/(total+1))
15	**Total**	=SI(choix=1;0;SOMME(L(-6)C:L(-1)C))	=SI(choix=1;0;SOMME(L(-6)C:L(-1)C))

Feuil1 Feuil2 Feuil3

Figure 4.4.18 Formules utilisées pour définir le tableau de la simulation d'un dé.

[1] Comme la valeur de la cellule indiquant le nombre de simulations augmenterait alors de 1 à chaque itération, l'écart maximal serait toujours d'au moins 1, et le nombre d'itérations atteindrait la valeur maximale spécifiée.

L'explication de ce phénomène fait appel à l'algorithme utilisé par *Excel* pour recalculer ses feuilles lorsque des références circulaires sont permises (c'est-à-dire quand la case « Itération » est cochée). Quand un nouveau lancer est effectué (c'est-à-dire quand un recalcul est déclenché), *Excel* refait ses calculs ligne par ligne, du haut vers le bas. En particulier, les calculs ci-dessous sont faits dans l'ordre suivant :

- choix du nouvel état du dé (cellule L3C3)
- mise à jour du nombre de fois, puis de la fréquence, où le résultat 1 a été obtenu (ligne 9)
- mise à jour du nombre de fois, puis de la fréquence, où le résultat 2 a été obtenu (ligne 10)
- ...
- mise à jour du nombre de fois, puis de la fréquence, où le résultat 6 a été obtenu (ligne 14)
- mise à jour du total et de la somme des fréquences

On voit donc que le total est mis à jour **après** que toutes les fréquences aient été calculées, ce qui est incorrect : il faudrait que le total soit mis à jour **avant** le calcul des fréquences. Et comme on sait que le total augmentera de 1 à chaque lancer, le véritable total **au moment de calculer les fréquences** sera de (total+1), d'où les formules utilisées à la colonne 3.

Nous laissons au lecteur le soin de réaliser le graphique associé aux diverses fréquences obtenues. Mentionnons simplement que nous avons voulu que les barres se placent, grosso-modo, vis-à-vis les données correspondantes du tableau : nous avons donc spécifié que les abscisses devaient être en mode inverse, et que l'axe des ordonnées devait couper à l'abscisse maximale.

Nous pouvons maintenant lancer le dé en commandant un recalcul de la feuille autant de fois que désiré. Cependant, même en laissant les doigts enfoncer la combinaison de touches provoquant ce recalcul pendant un long moment, profitant ainsi du mode de répétition automatique du clavier, il faut avouer qu'il reste assez pénible d'obtenir un grand nombre de lancers. D'où l'intérêt de l'ajout d'un bouton commandant autant de recalculs que spécifié dans une cellule donnée.

Voici comment procéder : après avoir choisi l'outil « Bouton » de la barre d'outils « Formulaires », vous dessinez le bouton désiré. Dès que vous relâchez le bouton de la souris, *Excel* vous propose de lui associer une macro. Vous pouvez conserver le nom suggéré par *Excel*, et vous cliquez sur le bouton « Nouvelle » pour aller dans l'éditeur *Visual Basic for Applications*. Vous ne serez pas dépaysés car il ressemble beaucoup à celui de *Word*, que vous avez utilisé avec *Langage Graphique*.

```
Sub Bouton1_QuandClic()
   For n = 1 To Cells(3, 8).Value
      Calculate
   Next n
End Sub
```

Vous tapez ensuite les instructions nécessaires pour obtenir le petit programme ci-contre. Voyons comment ça fonctionne. Remarquons tout d'abord deux instructions propres à *Excel* : « Calculate », qui provoque un recalcul de la feuille, et « Cells(3,8).Value », qui retourne la valeur contenue dans la cellule L3C8. Ces instructions sont utilisées dans le cadre d'une instruction « For ... Next », que nous avons déjà utilisée avec *Langage Graphique*, dont l'effet global est ici de commander autant de recalculs que le nombre placé dans la cellule L3C8. Quand nous avons terminé, on quitte l'éditeur *Visual Basic* pour retourner à *Excel* : on peut alors modifier le nom, la taille et la position du bouton. On termine le tout par un clic hors du bouton, dans une cellule quelconque de la feuille de calcul. Comme d'habitude, si nous voulons plus tard apporter des modifications à notre bouton, nous procédons via un clic-droit sur celui-ci.

À partir de maintenant, pour lancer le dé n fois, il suffira de placer la valeur n dans la cellule L3C8 puis de faire un clic sur le bouton que nous venons de définir. Si la valeur de n est grande, le temps pris pour exécuter les recalculs peut être très long : libre à nous de laisser rouler notre ordinateur toute la nuit, ou d'interrompre le programme en pressant sur la touche « ESC ». Soulignons ici la présence d'un phénomène que nous rencontrerons souvent : pour obtenir une fonctionnalité très utile, il suffit d'ajouter un petit programme fort simple.

Nous voici donc arrivé au terme de la création de notre feuille de calcul. Il faut avouer que le résultat obtenu est assez élémentaire, mais soulignons cependant que les techniques employées peuvent être réinvesties pour simuler des phénomènes moins triviaux (voir par exemple l'exercice 13 et le projet 5 de ce chapitre). À titre d'exemple (voir projet 3), on pourra créer une feuille *Excel* pour donner une réponse approximative à la question suivante : si on veut réunir une collection d'objets placés dans des boîtes de céréales, combien faudra-t-il acheter de boîtes (en moyenne) pour compléter notre collection? Notons que la réponse donnée par *Excel* sera approximative et expérimentale : on simulera plusieurs fois la complétion d'une telle collection, et on calculera en moyenne le nombre d'achats qui auront été nécessaires. On pourra constater qu'une telle approche est, somme toute, assez simple, surtout si on la compare à une approche plus théorique, qui nous apporterait une réponse exacte mais au prix de l'utilisation de mathématiques et de raisonnements plus sophistiqués.

4.5 Simulation déterministe du hasard

Dans la section précédente, nous avons utilisé la fonction ALEA() qui, avons-nous affirmé, retourne un « nombre aléatoire » dans l'intervalle [0,1). Mais on peut se demander comment une machine déterministe comme l'ordinateur peut choisir des nombres au hasard. Y a-t-il à l'intérieur un petit lutin qui joue à la roulette pour déterminer, une à une, les décimales de ces nombres? Ou, plus sérieusement, a-t-on installé des dispositifs physiques, basés sur des phénomènes non déterministes comme la désintégration nucléaire, dont diverses mesures alimentent un générateur de nombres aléatoires?

Pas du tout! En fait, tous les nombres dits « aléatoires » qui sont produits par l'ordinateur sont le résultat de calculs répétés où le hasard n'intervient en aucune façon, mais dont les résultats successifs forment une suite de nombres qui « ressemble[1] » à une suite aléatoire. De tels nombres sont parfois appelés « pseudo-aléatoires ». Tout ceci vous paraît-il très abstrait? Illustrons le tout en donnant un exemple d'un tel calcul...

Supposons donc que nous voulions obtenir une suite de nombres « aléatoires » entre 0 et 1, avec une partie décimale de 4 chiffres. Nous choisissons tout d'abord un entier naturel m_0 de 4 chiffres qui agira à titre de « générateur de nombres aléatoires » : par exemple m_0 = 8767. Notre premier nombre pseudo-aléatoire x_0 sera obtenu en « plaçant une virgule devant m_0 » : nous obtiendrons ainsi 0, 8767 dans notre exemple.

---

[1] On discutera plus tard des façons de vérifier une telle « ressemblance ».

Notre second nombre entier m_1 sera obtenu de la façon suivante : on multiplie m_0 par m_0 , ce qui nous donne en général un entier de 8 chiffres[1], et l'on ne conserve que les 4 chiffres du milieu. Dans notre exemple, on aura

$$8767 \times 8767 = 76860289$$

et notre second entier m_1, composé des 4 chiffres du milieu de ce produit, sera donc 8602. Et nous obtiendrons notre second nombre aléatoire x_1 en divisant m_1 par 10^4 : dans notre cas, on aura $x_1 = 0,\ 8602$.

Plus généralement, nous obtiendrons m_{k+1} en prenant les 4 chiffres du milieu du produit $m_0 \times m_k$, puis $x_{k+1} = \dfrac{m_{k+1}}{10^4}$. Mais comment dire à *Excel* de prendre les 4 chiffres du milieu d'un nombre n de huit chiffres? Il suffit d'abord de prendre le reste après division de n par 10^6, pour éliminer les deux premiers chiffres, puis de diviser par 10^2 et d'éliminer la partie décimale pour éliminer les deux derniers chiffres : « =TRONQUE(MOD(*n*;10^6)/10^2) ».

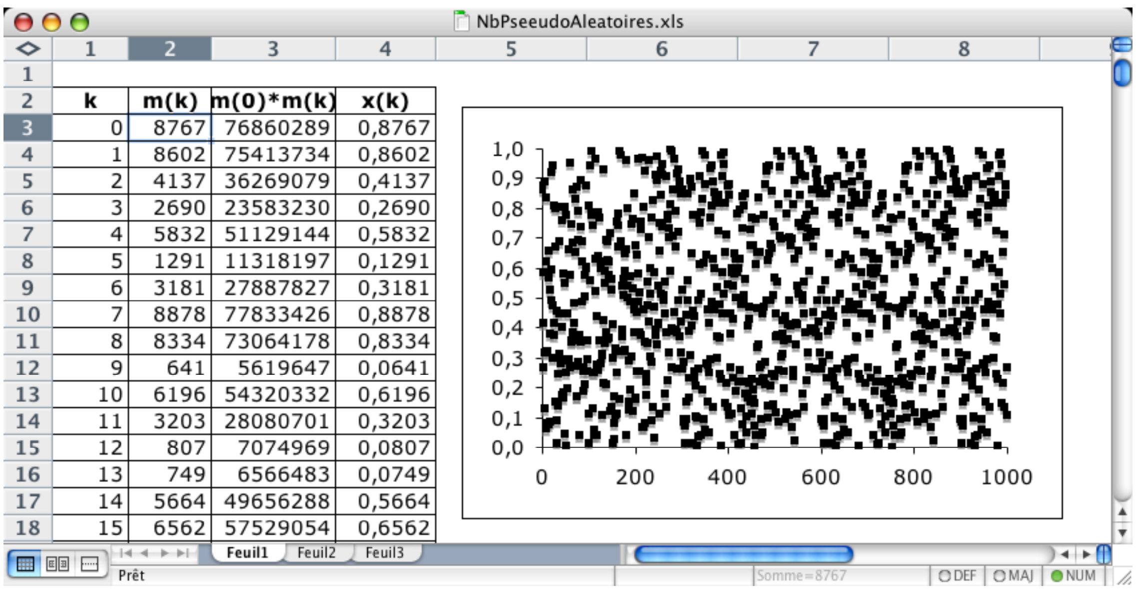


Figure 4.5.19 Feuille de calcul engendrant des nombres pseudo-aléatoires.

En mettant tout ceci en pratique, nous obtenons la feuille de calcul illustrée à la figure **4.5.19**, qui calcule 1000 nombres pseudo-aléatoires et les représente ensuite à l'aide d'un graphique en nuage de points non reliés. Un tel graphique nous permet de faire un premier test visuel pour vérifier que la suite ainsi obtenue ressemble bien à une suite aléatoire. La figure **4.5.20** nous donne un aperçu de ce qui peut se passer quand on change notre générateur m_0 : les suites résultantes ont parfois des régularités bien peu aléatoires.

[1] Si le résultat comporte moins de 8 chiffres, on convient d'ajouter à sa gauche assez de « 0 » pour obtenir 8 chiffres au total.

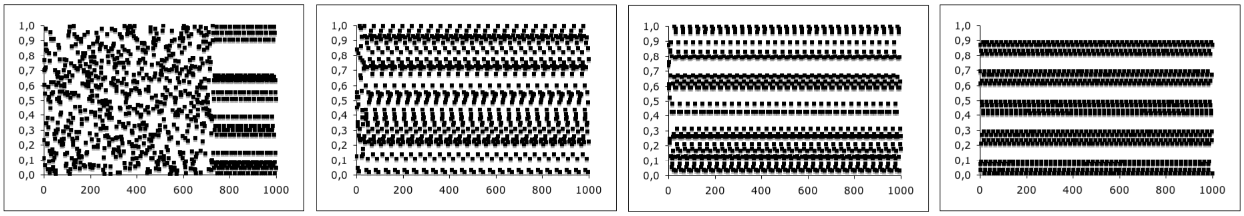


Figure 4.5.20 Graphiques associés à des suites obtenues avec d'autres générateurs (de gauche à droite : 1111, 1234, 8760 et 8700).

Mais une simple inspection visuelle ne suffit pas à nous assurer que notre suite a toutes les caractéristiques d'une suite aléatoire. Il nous faut effectuer des tests plus précis, dont le nombre et l'exigence dépendent de l'application visée : veut-on simuler des phénomènes aléatoires simples dans un contexte scolaire, ou veut-on effectuer des simulations pour concevoir un réacteur nucléaire? Une discussion élaborée du processus dépasse le niveau de cet ouvrage[1], mais nous allons cependant esquisser quelques tests allant dans ce sens.

Intéressons-nous à la suite des chiffres que l'on retrouve à une position décimale donnée, la position des centièmes par exemples : si la suite des nombres est « aléatoire », il devrait en être de même de ces suites de chiffres. Dans le cas où le générateur est 8767, nous obtenons la suite de 1000 chiffres suivante :

7, 6, 1, 6, 8, 2, 1, 8, 3, 6, 1, 2, 8, 7, 6, 5, ...

Si cette suite ressemble bien à une suite aléatoire, il faut s'attendre à ce que chacun des 10 chiffres décimaux apparaisse environ une fois sur 10. Comme premier test, on peut donc compter le nombre d'occurrences de chaque chiffre et voir si les résultats obtenus sont près de 100.

Mais ce test n'est pas suffisant en soi, puisqu'il serait couronné de succès si on avait d'abord 100 chiffres 0 consécutifs, puis 100 chiffres 1 consécutifs, etc. Pour détecter un tel biais, on peut faire appel à un second test, portant sur la suite des blocs de deux chiffres :

76, 16, 82, 18, 36, 12, 87, 65, ...

Si notre suite de nombres était aléatoire, il faudrait s'attendre à ce que les deux chiffres d'un bloc donné soient différents 10×9 fois sur 10×10 (Pourquoi?), et donc qu'ils soient égaux 10 fois sur 100 (Pourquoi?). On peut donc vérifier si ces proportions sont vérifiées approximativement par notre suite de blocs de deux chiffres.

De même, on peut regrouper notre suite de chiffres en blocs de trois chiffres

761, 682, 183, 612, 876, , ...

et tenter de vérifier si on obtient en pratique une bonne approximation des valeurs théoriques suivantes :

- la probabilité que les trois chiffres soient tous distincts est de $\dfrac{10\times9\times8}{10^3}=0,72$ (Pourquoi?)
- la probabilité que les trois chiffres soient tous pareils est de 0,01 (Pourquoi?)
- la probabilité qu'exactement deux des trois chiffres soient égaux est de 0,27 (Pourquoi?).

---

[1] Le lecteur intéressé pourrait consulter Donald Knuth, *The Art of Computer Programming*, Volume 2, Chapter 3, Addison-Wesley, 1997 (third edition).

On vous proposera dans le projet 6 de réaliser les trois tests décrits ci-dessus, et ce pour chacune des quatre positions décimales des nombres de nos suites pseudo-aléatoires. La liste des tests qu'on pourrait ainsi appliquer à ces nombres est loin d'être complète, mais nous allons nous arrêter là. On peut aussi imaginer d'autres calculs permettant de produire des suites pseudo-aléatoires (voir par exemple l'exercice 14), mais nous ne poursuivrons pas plus avant dans cette direction.

Terminons cette brève incursion dans le monde des nombres pseudo-aléatoires en mentionnant un problème, et une solution possible. Nous savons maintenant comment engendrer **une** suite de tels nombres, mais il ne faudrait pas que la même suite soit toujours utilisée car cela risquerait de biaiser les résultats obtenus. Doit-on avoir recours à plusieurs suites pseudo-aléatoires, engendrées par des calculs (ou, à tout le moins, par des générateurs) différents? En fait ce n'est pas absolument nécessaire : on peut toujours utiliser la même suite, mais faire varier la position à partir de laquelle on commence, par exemple en se servant de l'horloge interne de l'ordinateur.

Nous voilà donc arrivés au terme de notre brève étude des tableurs. D'une certaine façon, nous n'avons fait qu'effleurer le sujet. Entre autres choses, nous n'avons pas parlé du **solveur**[1], qui est une composante mathématique importante d'*Excel* (car elle permet de résoudre de vastes classes de systèmes d'équations), mais dont le caractère de généralité et de « boîte noire » nous semble moins utile au niveau secondaire. Mais nous pensons que ces deux chapitres constituent une base solide, sur laquelle on peut s'appuyer pour poursuivre une étude autonome d'*Excel*.

4.6 Exercices

1- *Contrôler les coordonnées d'un point d'un graphique à l'aide de barres de défilement*
Réalisez une feuille de calcul contenant deux cellules nommées x et y et un graphique contenant le point de coordonnées (x, y). Ajoutez deux barres de défilement, l'une horizontale et l'autre verticale, et complétez la feuille de la façon suivante :
   a) La barre de défilement horizontale contrôle la valeur de la cellule x et la fait varier entre -100 à 100.
   b) La barre de défilement verticale contrôle la valeur de la cellule y et la fait varier entre -100 à 100.
   c) Dans le graphique fixez les bornes des deux axes à -100 et 100.
   d) Placez le graphique et les barres de défilement pour que le point soit toujours approximativement à la verticale du curseur de la barre horizontale et à la même hauteur que le curseur de la barre verticale.

   Remarque : si le point ne se déplace pas en même temps que les curseurs des barres de défilement revoyez le début de la section 4.2.

[1] Le lecteur intéressé pourra consulter l'aide en ligne d'*Excel*, et en particulier la rubrique « À propos du solveur » qui identifie les méthodes numériques utilisées. Veuillez noter que, dans la version *Macintosh* d'*Excel*, le solveur ne fonctionne correctement qu'avec le style de référence A1.

2- *Superposer des graphiques avec des caractéristiques diverses*

On apprend en géométrie élémentaire que les médianes d'un triangle sont concourantes et que le point d'intersection se situe aux $\frac{2}{3}$ des médianes à partir des sommets.

Réalisez une feuille de calcul *Excel* dans laquelle :

a) on peut entrer les coordonnées des trois sommets du triangle sur deux colonnes, une pour les abscisses et une pour les ordonnées.
Nommez individuellement chacune des coordonnées, mais nommez aussi chacune des colonnes contenant les abscisses et les ordonnées.

b) pour chaque sommet, sur la même ligne, placez deux cellules contenant les coordonnées du point milieu du côté opposé.
Nommez aussi les colonnes contenant les abscisses et les ordonnées de ces points.

c) pour chaque sommet, sur la même ligne, placez deux cellules, contenant les coordonnées du point situé au $\frac{2}{3}$ de la médiane associée. (Écrivez les formules en utilisant les noms)

	1	2	3	4	5	6	7	8
1	**Vérification que les médianes d'un triangle sont concourantes**							
2								
3								
4	Sommets du triangle			Milieu du coté opposé			Points aux 2/3 des médianes	
5	x	y		mx	my		gx	gy
6	1	4		2,5	6		2	5,333333333
7	2	5		2	5,5		2	5,333333333
8	3	7		1,5	4,5		2	5,333333333
9								

Vous devriez constater numériquement que les trois derniers points calculés coïncident (aux approximations d'arrondissement près).

d) Insérez un graphique montrant :

1. le triangle en bleu
2. les médianes en noir
3. les milieux des côtés en vert et de forme carrée
4. le point d'intersection des médianes en rouge et de forme ronde.

3- *Parcours d'un intervalle donné à l'aide d'une barre de défilement*

On a vu dans la section 4.1 comment, à l'aide d'une barre de défilement, on peut faire varier la valeur d'une cellule entre deux bornes données.

On se donne a et b les bornes (des nombres décimaux) d'un intervalle et n un nombre d'étapes pour le parcourir et on veut associer une barre de défilement et une cellule de façon que le déplacement de la barre de défilement permette au contenu de la cellule de prendre n+1 valeurs de a à b inclusivement.

a) Réalisez une feuille de calcul avec des cellules contenant des valeurs pour a, b et n. Ajoutez une barre de défilement et une cellule avec une formule adéquate pour que, lorsque la barre est munie des réglages adéquats, la cellule prenne bien les valeurs désirées. Est-il possible de choisir la formule pour que ça fonctionne indifféremment pour $a < b$ et $b < a$?

b) Refaites le même travail si, au lieu de se donner n (le nombre d'étapes), on se donne e qui sera l'écart en deux valeurs successives.

Remarque : On veut donc une cellule qui prend les valeurs a, $a+e$, $a+2e$, etc. En général il n'existera pas d'entier m tel que $b = a+me$. Pour cet exercice vous choisirez pour borne le $b'=a+me$ qui est le plus proche de b.

4- *Droite de pente variable*
Réalisez la feuille de calcul décrite dans l'avant-propos. Cette feuille montre sur un graphique :

- une droite de pente m modifiable avec une barre de défilement
- le point d'abscisse 1 situé sur la droite.

5- *Influence du nombre de points calculés dans un graphique*
Reprenez la feuille de calcul implantant l'exemple de la section 4.1 et modifiez-la pour tracer le graphe avec 1001 points plutôt que 101.

6- *Varier les paramètres pour faire coïncider af(bx+h)+k avec une fonction cible*
Construisez une feuille de calcul *Excel* comportant les éléments suivants :

- Quatre barres de défilement faisant varier des paramètres :
  - a de -3 à 3
  - b de -4 à 4
  - h de -4π à 4π
  - k de -π à π

Un graphique contenant les tracés de $a\cos(bx+h)+k$ et de $\sin(x)$.

L'objectif est de réaliser une feuille de calcul qui permet de faire varier les paramètres pour obtenir la superposition des courbes.
Apportez une attention particulière aux points suivants :

- Le nombre de points où les fonctions sont calculées doit être suffisamment grand pour permettre d'obtenir de bons tracés, mais pas trop grand au point de ralentir indûment les calculs.
- Les valeurs spécifiées comme paramètres des barres de défilements doivent permettre d'obtenir effectivement les valeurs permettant la superposition des courbes.

7- *Organisation des données pour utiliser le produit matriciel*
Nous avons vu dans la section 4.3 comment appliquer une matrice de rotation à une série de points pour trouver leurs transformés :

$$\begin{pmatrix} x_1' & \cdots & x_n' \\ y_1' & \cdots & y_n' \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_1 & \cdots & x_n \\ y_1 & \cdots & y_n \end{pmatrix}$$

Dans *Excel* nous avons réalisé ce calcul grâce à la formule matricielle
{=ProduitMat(Rot; Partie1)}
où Rot est le nom donné à la plage de cellules représentant la matrice de rotation et Partie1 est le nom donné à la plage de cellules représentant la matrice $\begin{pmatrix} x_1 & \cdots & x_n \\ y_1 & \cdots & y_n \end{pmatrix}$.

Mais, en général, il n'est pas toujours possible de multiplier deux matrices. Pour pouvoir faire la multiplication AB de deux matrices A et B (dans cet ordre) il faut absolument que le nombre de colonnes de A soit égal au nombre de lignes de B.
Ainsi si nous avions organisé « Partie1 » avec les coordonnées x et les y en colonnes $\begin{pmatrix} x_1 & y_1 \\ . & . & . \\ x_n & y_n \end{pmatrix}$ la multiplication ne serait plus possible.
Heureusement, *Excel* nous fournit une fonction qui transforme les lignes en colonnes et vice versa, c'est la fonction TRANSPOSE. Ainsi $TRANSPOSE\left(\begin{pmatrix} x_1 & y_1 \\ . & . & . \\ x_n & y_n \end{pmatrix}\right) = \begin{pmatrix} x_1 & \dots & x_n \\ y_1 & \dots & y_n \end{pmatrix}$ et la multiplication {=ProduitMat(Rot; TRANSPOSE(Partie1))} devient possible. Cependant le résultat est une matrice à 2 lignes et n colonnes qui ne peut occuper qu'une plage de 2 lignes et n colonnes. Si on veut l'écrire sur 2 colonnes et n lignes il faut évidemment transposer le résultat.
Reprenez l'exemple de la rotation mais organisez les « Partie1 » et « Partie2 » sur 2 colonnes.

a) Vérifiez qu'*Excel* n'accepte plus la formule {=ProduitMat(Rot; Partie1)}
b) Utilisez la fonction TRANSPOSE pour corriger le problème et construire une plage de 2 colonnes représentant la figure transformée.
c) Vérifiez que tout fonctionne correctement.

Remarque : la lecture d'un bon livre d'algèbre matricielle pourra vous permettre d'approfondir les aspects mathématiques de la transposition. Par exemple, vous pourrez y apprendre qu'on aurait aussi pu obtenir le même résultat en transposant la matrice de rotation à la condition d'inverser l'ordre du produit matriciel.

8- *Différentes transformations : isométries et homothétie.*
Parmi les différentes transformations du plan étudiées au secondaire on trouve les translations, les symétries et les homothéties. La translation n'est pas représentable via une multiplication par une matrice 2 x 2. Voici quelques matrices d'isométries :

$$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix}.$$

Créez trois feuilles de calcul identiques avec les coordonnées d'une figure initiale ne comportant pas de symétrie (par exemple la figure utilisée dans la section 4.3).

a) Dans une première feuille de calcul :
    1) Appliquez les quatre matrices ci-dessus à la figure initiale pour obtenir quatre figures transformées.
    2) Insérez un graphique contenant les cinq figures tracées chacune avec une couleur différente. Pouvez-vous identifier les quatre transformations? (Modifiez la figure initiale si nécessaire).
b) Une translation est déterminée par un vecteur (u, v) et n'est pas représentable via une multiplication par une matrice 2x2. Pour illustrer l'effet d'une translation, dans une deuxième feuille de calcul :

1) Inscrivez dans deux cellules les coordonnées u et v d'un vecteur de votre choix. Ce vecteur donnera la direction de la translation.
2) Ajoutez une barre de défilement dont la valeur t permettra de varier la longueur du vecteur de translation (tu, tv).
3) Créez un nouveau graphique contenant la figure initiale et sa transformée par la translation, tracées avec des couleurs différentes. Faites varier la longueur du vecteur de translation pour observer les étapes intermédiaires.

c) Une homothétie de rapport r centrée à l'origine à pour matrice $\begin{pmatrix} r & 0 \\ 0 & r \end{pmatrix}$. Dans la troisième feuille de calcul :
   1) Ajoutez une barre de défilement pour contrôler la valeur de r, utilisez une formule pour que r puisse prendre des valeurs positives et négatives.
   2) Créez un nouveau graphique comprenant la figure initiale et sa transformée par l'homothétie tracées avec des couleurs différentes. Faîtes varier le rapport d'homothétie r. Que se passe-t-il lorsque r devient négatif?

9- *Mise en évidence de la non commutativité de la composition des transformations du plan*
Nous avons vu comment représenter graphiquement l'image d'une figure après certaines transformations (rotations, homothéties) en utilisant le produit matriciel (fonction ProduitMat).
Nous allons maintenant considérer la réflexion par rapport à l'axe des x, dont la matrice associée est $\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$.
Nous allons utiliser *Excel* pour montrer que, lorsqu'on applique un certain nombre de transformations à une figure, le résultat final va dépendre, en général, de l'ordre d'application choisi.
En fait, nous allons étudier le cas particulier suivant : l'application d'une rotation suivie d'une réflexion (et aussi de la réflexion suivie de la rotation) à une figure.
Créez une feuille de calcul dans laquelle vous définissez une figure initiale (toute figure ne présentant pas de symétrie fera l'affaire; vous pouvez par exemple prendre la figure utilisée dans la section 4.3).
Ajoutez une barre de défilement pour contrôler l'angle de rotation.
Créez maintenant un graphique contenant les trois figures suivantes tracées avec trois couleurs différentes:
   a) la figure initiale
   b) la figure initiale après rotation puis réflexion
   c) la figure initiale après réflexion puis rotation.
Vous pouvez constater que les deux figures transformées sont, en général, distinctes.

10- *Graphique d'une courbe définie par des équations paramétriques*
On sait qu'un cercle de rayon r admet l'équation paramétrique
$$\alpha \to (r\cos(\alpha), r\sin(\alpha))$$
où le paramètre α varie dans l'intervalle $[0, 2\pi]$.
   a) Construire une feuille de calcul contenant un graphique du cercle tracé à l'aide de l'équation ci-dessus, en calculant 361 points, un pour chaque degré de 0 à 360.

(Attention : *Excel* suppose que les arguments des fonctions trigonométriques sont en radians)

b) Ajoutez une barre de défilement qui contrôle le rayon de 0 à 10 par sauts de 0,1.

c) Ajoutez une barre de défilement qui contrôle un point mobile sur le cercle (Pour une bonne visibilité tracez le point avec une couleur différente de celle du cercle).

Comment simuler une plage variable :

d) On voudrait maintenant tracer l'arc qui va du point $(r,0)$ jusqu'au point mobile.

Le problème, c'est que cet arc devra varier lorsque le point mobile se déplace. Une solution possible est d'avoir, comme pour le cercle, 361 points, un pour chaque degré de 0 à 360, mais d'utiliser une fonction « SI » pour que les points qui ne sont pas sur l'arc soient tous égaux au point mobile.
Implantez cette solution en traçant l'arc avec une couleur différente de celles du cercle et du point mobile.

11- *Graphique d'une courbe définie par des équations paramétriques*
On sait qu'une ellipse de demis axes a et b admet l'équation paramétrique

$$\alpha \to \left(a\cos(\alpha), b\sin(\alpha)\right)$$

où le paramètre α varie dans l'intervalle $[0,2\pi]$.

a) Construire une feuille de calcul contenant un graphique de l'ellipse tracé à l'aide de l'équation ci-dessus, en utilisant suffisamment de points.
(Attention : *Excel* suppose que les arguments des fonctions trigonométriques sont en radians)

b) Ajoutez deux barres de défilement qui contrôlent a et b de 0 à 10 par sauts de 0,1.
Ajoutez au graphique une seconde ellipse obtenue de la première par une rotation autour de l'origine dont l'angle est contrôlé par une barre de défilement variant de 0 à 360 degrés par sauts de 1º.

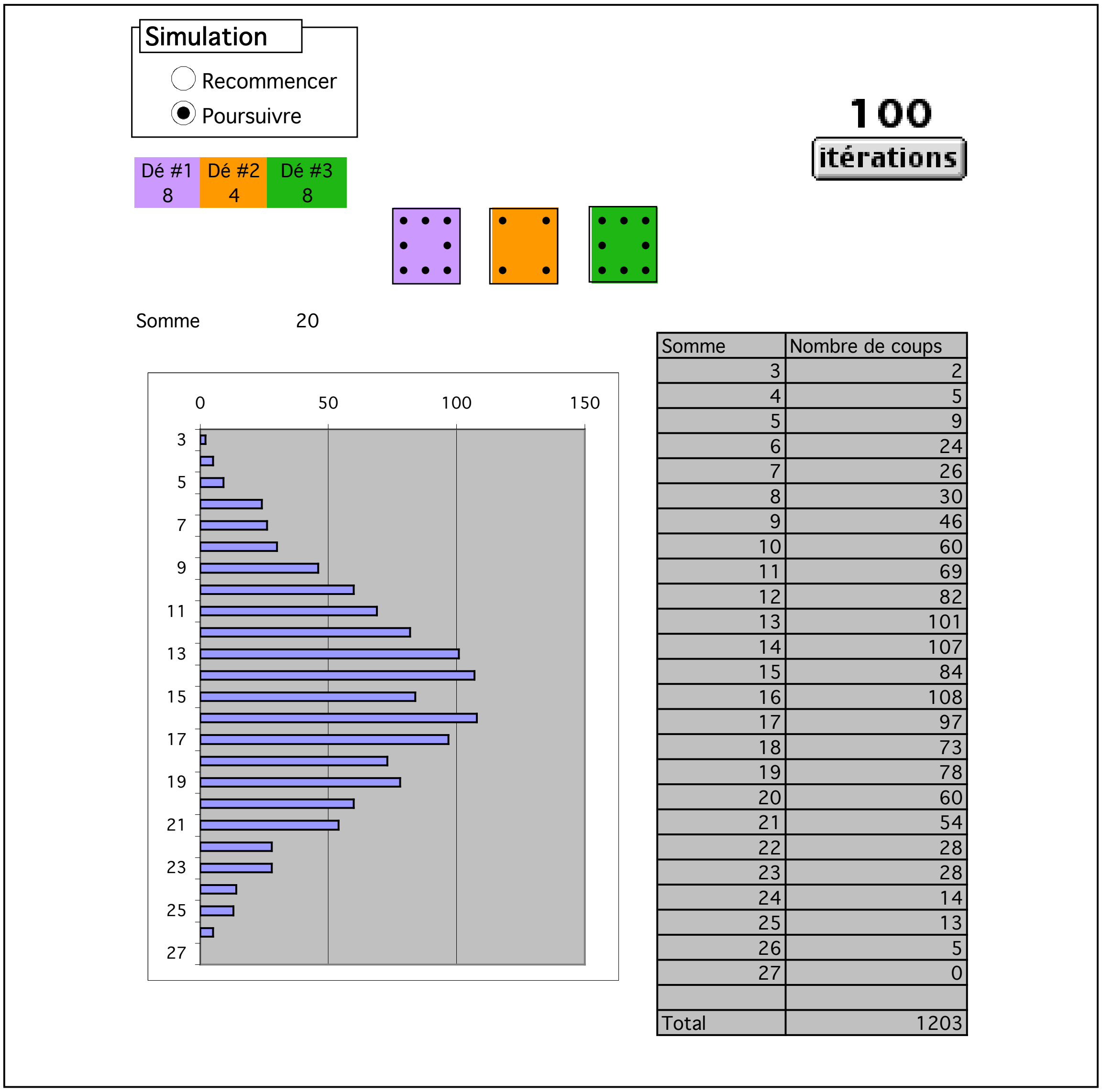
12- *Simulation du lancer d'un dé*
Sans utiliser les propriétés d'itération d'*Excel*, réalisez une feuille de calcul simulant le lancer d'un dé et représentant la distribution des résultats à l'aide d'un histogramme. Une façon de procéder est expliquée au début de la section 4.4.

13- *Utiliser les possibilités de calculs itérés d'Excel*
Dans la section 4.4 vous avez vu comment utiliser *Excel* et ses possibilités de calculs itérés pour simuler les lancers répétés d'un dé et construire un graphique statistique représentant la situation.
Dans cet exercice, vous allez généraliser cette démarche de deux façons :

- les « dés » utilisés produiront des résultats aléatoires allant de 1 à 9 (et non de 1 à 6 comme des dés « habituels »)
- il y aura trois « dés » au lieu d'un seul

et vous vous intéresserez à la somme des résultats obtenus pour en tracer un diagramme à barres.

Le dessin ci-dessous donne une idée de l'apparence que pourrait avoir la feuille de calcul.



Somme	Nombre de coups
3	2
4	5
5	9
6	24
7	26
8	30
9	46
10	60
11	69
12	82
13	101
14	107
15	84
16	108
17	97
18	73
19	78
20	60
21	54
22	28
23	28
24	14
25	13
26	5
27	0
Total	1203

14- *Génération d'une suite pseudo-aléatoire d'entiers*

On peut construire une suite pseudo-aléatoire d'entiers entre 0 et 9 de la façon suivante :

- on commence avec un nombre décimal x_0 entre 0 et 1, choisi arbitrairement, mais possédant au moins 5 décimales. Par exemple $x_0 = 0{,}379645937$.
- on génère ensuite un nouveau nombre décimal x_1 entre 0 et 1 en prenant la partie décimale du produit de x_0 par 147. Donc $x_1 = \textit{partie décimale de } 147 x_0$. Dans notre exemple $147 x_0 = 55{,}80795274$ et donc $x_1 = 0{,}80795274$.
- on continue de cette façon pour générer les nombres suivants $x_{n+1} = \textit{partie décimale de } 147 x_n$. Dans notre exemple

$$x_2 = \textit{partie décimale de } 147 x_1 = 0{,}7690528.$$

- la suite pseudo-aléatoire d'entiers est constituée des premières décimales des nombres $x_1, x_2, x_3, ...etc.$ Dans notre exemple 8, 7,…etc.

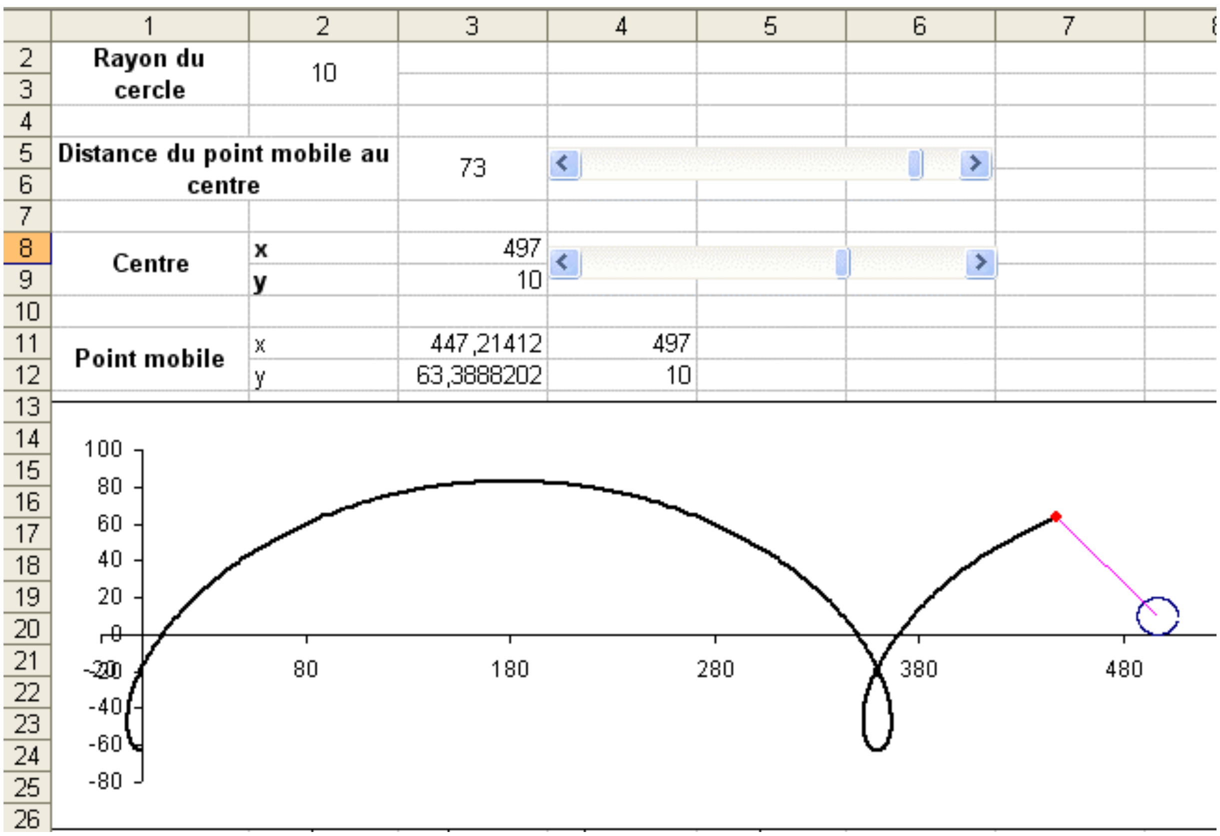
Le multiplicateur 147 donne un bon résultat et a été déterminé expérimentalement.

1. Réalisez un feuille de calcul qui implante cette méthode de construction d'une suite pseudo-aléatoire de chiffres. Construisez une suite assez longue, par exemple 1000 chiffres.
2. Ajoutez un histogramme qui montre le nombre de chaque chiffre obtenu.
3. Expérimentez avec des multiplicateurs différents de 147.

4.7 Projets

1- *Courbes paramétriques : variations sur la cycloïde*

La *cycloïde* est la courbe engendrée par un point fixé à un cercle qui roule en ligne droite sans glisser. On peut généraliser en fixant le point à un rayon du cercle ou à son prolongement (la courbe se nomme alors *trochoïde*).



Réalisez une feuille de calcul Excel comportant un graphique et deux barres de défilement de sorte que :

a) le graphique contienne un cercle, un point et un segment joignant le point au centre du cercle.

b) lorsqu'on fait varier une barre de défilement, le point s'éloigne ou se rapproche du centre du cercle

c) lorsqu'on fait varier la deuxième barre de défilement, le cercle roule sur l'axe des x et la trajectoire du point se trace depuis la position initiale du point jusqu'à sa position courante.

Rappel : sur un cercle de rayon *r*, l'arc intercepté par un angle au centre de α radians mesure $r\alpha$.

Vous pouvez généraliser en faisant rouler le cercle sur un cercle fixe, soit à l'extérieur ou à l'intérieur. Si le point est sur la circonférence du cercle mobile on obtient une épicycloïde dans le premier cas et une hypocycloïde dans le second.

2- *La méthode de bissection via les possibilités d'itération d'Excel*
Nous avons vu dans le chapitre précédent comment utiliser la méthode de bissection pour calculer les zéros d'une fonction. La méthode utilisée consistait à itérer un calcul et à construire une table à raison d'une ligne par itération.
Il est possible, maintenant, de réaliser le même travail sans construire de table : il suffit simplement d'utiliser les propriétés de calculs itérés d'*Excel*.
Construisez, sur ce principe, une feuille de calcul *Excel* qui, une fonction $f(x)$ étant donnée, permet de rechercher une solution de l'équation $f(x) = 0$.

Voici un exemple de feuille de calcul qui réalise ce travail pour la fonction $f(x) = x^2 - 2$:

	1	2	3	4	5	6	7	8
1								
2				Fonction	x^2-2	Ok		
3		Initialiser						
4			2					
5		Résoudre						
6								
7			Gauche	Droite	Milieu	f(gauche)	f(droite)	f(milieu)
8		Veleurs initiales	0	2	1	-2	2	-1
9		Calculs itérés	1,41421356	1,41421356	1,41421356	0	0	0
10								

Les valeurs initiales sont entrées dans les cellules L8C3 et L8C4.

a) Lorsque la fonction change de signe entre les bornes la cellule L2C6 affiche « Ok » et « Erreur » dans le cas contraire

Les cellules L9C3 et L9C4 contiennent des formules de façon à ce que :

b) lors d'un clic sur la case d'option « Initialiser » elle prennent les mêmes valeurs que L8C3 et L8C4.
c) lors d'un clic sur la case d'option « Résoudre », leurs valeurs changent par calcul itéré jusqu'à ce que le nombre maximal d'itérations soit atteint ou jusqu'à ce que les changements soient inférieurs à l'écart maximal spécifié.

Utilisez votre feuille pour calculer $\sqrt{2}$, la racine cubique de 5, la solution de $\log(x) = 3$, une solution de $\cos(x) = 2\sin(x)$, etc. Dans chaque cas expérimentez en modifiant les bornes, le nombre maximal d'itérations et l'écart maximal.

3- *Résolution de problème via une simulation probabiliste*
On se propose d'utiliser *Excel* afin de résoudre expérimentalement le problème suivant :

Un manufacturier insère une carte de hockey dans chacune de ses boîtes de céréales. Combien faut-il acheter de boîtes en moyenne pour obtenir la collection des 7 cartes ?

On peut simuler l'acquisition d'une collection de cartes comme suit :

a) On choisit une « carte » au hasard (un nombre entre 1 et 7) et on retient d'une façon ou d'une autre le résultat obtenu.
b) On continue de la sorte tant que toutes les cartes n'ont pas été obtenues.
c) Quand toutes les cartes ont été obtenues, on aura compté le nombre de « boites » (c'est-à-dire de choix de cartes) qui ont été nécessaires afin de compléter la collection.

	1	2	3	4	5	6	7	8	9	10	11	12
9	Essai	Carte obtenue	1	2	3	4	5	6	7	COLLECTION	Nb. boîtes	
10												
11	1	6	0	0	0	0	0	1	0		1	
12	2	2	0	1	0	0	0	1	0		2	
13	3	1	1	1	0	0	0	1	0		3	
14	4	3	1	1	1	0	0	1	0		4	
15	5	5	1	1	1	0	1	1	0		5	
16	6	3	1	1	1	0	1	1	0		6	
17	7	2	1	1	1	0	1	1	0		7	
18	8	4	1	1	1	1	1	1	0		8	
19	9	2	1	1	1	1	1	1	0		9	
20	10	1	1	1	1	1	1	1	0		10	
21	11	7	1	1	1	1	1	1	1	OUI	11	
22	12	5	1	1	1	1	1	1	1	OUI	11	
23	13	4	1	1	1	1	1	1	1	OUI	11	
106	96	5	1	1	1	1	1	1	1	OUI	11	
107	97	3	1	1	1	1	1	1	1	OUI	11	
108	98	2	1	1	1	1	1	1	1	OUI	11	
109	99	3	1	1	1	1	1	1	1	OUI	11	
110	100	5	1	1	1	1	1	1	1	OUI	11	
111												
112										Résultat courant		11
113										Nombre total d'expériences		72
114										Somme des résultats		1368
115										Moyenne des résultats		19,00

Mais une seule simulation ne donne pas nécessairement une idée juste du nombre moyen de boîtes qu'il faudrait acheter : on pourrait être très chanceux et obtenir la collection au bout de

7 achats, mais on pourrait aussi être très malchanceux et ne compléter la collection qu'au bout de 80 achats.

Réalisez une feuille de calcul qui permette non seulement de faire une simulation, mais aussi de cumuler les résultats de plusieurs simulations.

De façon plus précise, réalisez une feuille de calcul *Excel* comportant :

a) deux cases d'option permettant de « Recommencer », c'est-à-dire de tout remettre à zéro, ou d' « Accumuler » les résultats de toutes les séries de simulations.
b) une cellule affichant le nombre moyen d'achats nécessaires pour obtenir une collection complète.

Comme dans l'exemple du dé vous pouvez ajouter une cellule précisant le nombre de répétitions désiré et un bouton qui déclenche ces répétitions.

4- *Une autre façon de représenter les transformations du plan par des matrices*

Nous avons vu (section 4.3 et exercices 7, 8, 9) comment représenter certaines transformations du plan par des matrices 2 par 2 et comment utiliser ces matrices avec *Excel*. Nous avons aussi mentionné (exercice 8) que la translation n'est pas représentable via une multiplication par une matrice 2x2, en fait c'est aussi le cas de toute transformation qui ne préserve pas l'origine des coordonnées (par exemple rotation ou homothétie dont le centre n'est pas à l'origine, réflexion dont l'axe ne passe pas par l'origine). Dans ce projet, vous allez voir comment représenter toutes ces transformations par des matrices 3 par 3.
Pour les transformations représentables par une matrice 2 par 2, nous complétons la matrice en ajoutant une ligne et une colonne de 0, sauf sur la diagonale où nous plaçons un 1. Par exemple, considérons la matrice de rotation d'angle α :

$$\begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix} \text{ devient } \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Les vecteurs, eux, sont complétés en ajoutant le nombre 1 comme troisième coordonnée :

$$\begin{pmatrix} x \\ y \end{pmatrix} \text{ devient } \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}.$$

Ensuite, lorsqu'on transforme un vecteur par une matrice, on ne tient compte que des deux premières coordonnées. Par exemple :

$$\begin{pmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} x\cos(\alpha) - y\sin(\alpha) \\ x\sin(\alpha) + y\cos(\alpha) \\ 1 \end{pmatrix}$$

et les deux premières coordonnées donnent bien le transformé du point $\begin{pmatrix} x \\ y \end{pmatrix}$ par la rotation d'angle α.

Maintenant on peut représenter une translation de vecteur $\begin{pmatrix} u \\ v \end{pmatrix}$ par la matrice $\begin{pmatrix} 1 & 0 & u \\ 0 & 1 & v \\ 0 & 0 & 1 \end{pmatrix}$. En effet $\begin{pmatrix} 1 & 0 & u \\ 0 & 1 & v \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} x+u \\ y+v \\ 1 \end{pmatrix}$ et les deux premières coordonnées du transformé représentent bien le translaté de $\begin{pmatrix} x \\ y \end{pmatrix}$ par le vecteur $\begin{pmatrix} u \\ v \end{pmatrix}$.

(Le lecteur intrigué par cette façon de procéder pourra se rendre sur le site http://fr.wikipedia.org/wiki/Accueil et chercher « coordonnées homogènes »)

La représentation de la translation par une matrice permet maintenant de représenter beaucoup d'autres transformations, par exemple :

- Pour faire tourner une figure d'un angle α autour d'un centre $(a,b) \neq (0,0)$, on lui applique trois transformations successives : translation qui déplace le centre de rotation à l'origine, rotation d'angle α autour de l'origine et finalement la translation qui déplace l'origine au centre de rotation.
- Le même principe permet d'appliquer à une figure une homothétie de centre $(a,b) \neq (0,0)$.
- Pour appliquer une réflexion d'axe d'équation $y = ax + b$, on utilise une idée semblable : déplacer l'axe de symétrie pour le faire coïncider avec l'axe des *x*, en se souvenant que la droite $y = ax + b$ fait avec l'axe des *x* un angle dont la tangente est a.

Remarque : en faisant le produit de toutes les matrices concernées on peut constater que chacune de ces nouvelles transformations peut se représenter par une seule matrice 3 par 3.

Mettez maintenant tout ceci en application :
Commencez par créer une feuille *Excel* avec une figure de départ et son graphique.

a) Réservez 2 cellules pour les coordonnées (a,b) d'un centre de rotation et une autre cellule pour un angle α de rotation.
Utilisez ce que nous avons dit plus haut pour ajouter au graphique la figure transformée par la rotation d'angle α et de centre (a,b) en utilisant des matrices 3 par 3.
Ajoutez une barre de défilement qui permette de faire varier l'angle de rotation.
b) Dans une nouvelle, feuille répétez l'étape a) en remplaçant la rotation par une homothétie de rapport r. La barre de défilement fera varier r.
c) Dans une nouvelle feuille, illustrez, toujours en utilisant des matrices 3 par 3, la réflexion par rapport à un axe $y = ax + b$. Ajoutez des barres de défilement pour faire varier a et b.

Dans chacun des cas vous pouvez aussi rendre certaines coordonnées de la figure variables via des barres de défilement pour observer les variations des figures transformées.

5- *Promenades aléatoires*

Imaginons un point (x, y) dans un plan, qui se déplace par une succession de « pas » de direction et longueur aléatoires, c'est ce qu'on appelle une « marche aléatoire »[1].

Dans ce projet, vous allez utiliser *Excel* pour illustrer différentes variations du phénomène obtenues en plaçant diverses contraintes sur la partie « hasard ». Dans trois feuilles de calcul différentes, fixez d'abord un point de départ arbitraire $(x, y) = (x_0, y_0)$, puis représentez graphiquement les marches aléatoires suivantes :

a) Pour faire un pas on translate le point (x, y) par un vecteur (u, v) dont les coordonnées u et v sont choisies aléatoirement dans l'intervalle $[-d, d[$, où d est un nombre positif que vous choisirez.

b) On procède comme en a) mais les pas se font seulement dans 4 directions : haut, bas, gauche et droite.

   Remarque : il est parfois intéressant d'utiliser la fonction CHOISIR d'*Excel* pour remplacer l'imbrication de multiples SI.

c) On procède comme en b) mais les pas sont tous de longueur fixe.

Pour avoir une bonne visualisation du phénomène, prenez un grand nombre de pas (1000 semble être un nombre correct), expérimentez avec différents choix pour d et placez des points de couleurs différentes au point de départ et au point d'arrivée.

6- *Nombres pseudo-aléatoires*

Pour ce projet, vous allez compléter la feuille de calcul présentée à la section 4.5 pour réaliser la série de tests mentionnés dans cette même section.

Commencez par ajouter à la feuille une barre de défilement dont la cellule associée, nommée *n*, prendra des valeurs entières de 1 à 4. La valeur de n indiquera le rang de la décimale sur laquelle on travaille.

a) Complétez la feuille de façon à compter le nombre de fois où chaque chiffre 0, 1, …, 9 apparaît à la position décimale *n* dans la suite pseudo-aléatoire des 1000 nombres. Ajoutez un histogramme représentant, en bleu, la distribution obtenue et, en jaune, la distribution attendue si on pense que la suite est aléatoire.

b) Imaginez maintenant que les chiffres de la nième décimale soient regroupés en 500 paquets de 2 (premier et second, troisième et quatrième, etc.). Complétez la feuille de façon à compter le nombre de paquets contenant des paires de chiffres identiques et le nombre de paquets contenant des chiffres différents.

   Ajoutez un histogramme représentant, en bleu, la fréquence obtenue et, en jaune, la fréquence attendue si on pense que la suite est aléatoire.

[1] Un tel système est un cas particulier d'un modèle mathématique appelé « promenade aléatoire » ou quelquefois « marche au hasard ».
Pour plus d'informations voir http://fr.wikipedia.org/wiki/Marche_aléatoire .

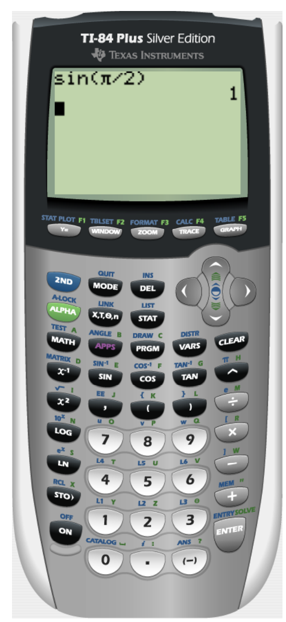
c) Imaginez maintenant que les chiffres de la nième décimale soient regroupés en 333 paquets de 3 (Négligez le premier chiffre pour en avoir 999 et regroupez second-troisième-quatrième, cinquième-sixième-septième, etc.). Complétez la feuille de façon à compter le nombre de paquets contenant 3 chiffres identiques, le nombre de paquets contenant 3 chiffres différents et le nombre de paquets « mixtes », c'est à dire ceux contenant exactement deux chiffres identiques.
Ajoutez un histogramme représentant, en bleu, la fréquence obtenue et, en jaune, la fréquence attendue si on pense que la suite est aléatoire.

Chapitre 5

La calculatrice graphique *TI-84Plus*

Par certains aspects, la calculatrice à affichage graphique *TI-84Plus* ressemble aux ordinateurs de la fin des années 70 : affichage monochrone de faible résolution, entrée difficile des lettres minuscules et des caractères accentués, interaction avec les applications limitée au clavier (ignorant souris et interface graphique), mémoire et vitesse extrêmement limitées, microprocesseur 8 bits utilisé, etc. Quand on a connu le confort d'utilisation des ordinateurs modernes, on ressent rapidement beaucoup de frustration au moment d'utiliser le tableur ou l'application de géométrie dynamique inclus dans ces petites machines.

Dans ces conditions, pourquoi consacrer un chapitre à l'étude de ces calculatrices ? Essentiellement parce que c'est la technologie la plus répandue, à l'heure actuelle, dans les écoles secondaires québécoises. Peut-être aussi parce que la comparaison entre ces calculatrices et nos ordinateurs peut nous aider à imaginer l'avenir, quand nos ordinateurs actuels seront devenus les calculatrices d'alors...



À moins de mention explicite du contraire, tout ce que nous dirons sera valable pour les calculatrices *TI-83Plus*, *TI-83Plus Silver Edition*, *TI-84Plus* et *TI-84Plus Silver Edition*, qui ne diffèrent les unes des autres que par leur vitesse et la taille de leur mémoire[1].

5.1 Vue d'ensemble

À première vue, notre calculatrice semble disposer des outils usuels de calcul : on remarque, par exemple, des touches permettant de calculer des fonctions trigonométriques, exponentielles et logarithmiques. De plus, on peut taper nos commandes sous une forme proche des mathématiques[2], et la réponse est affichée à la ligne suivante (voir ci-contre).

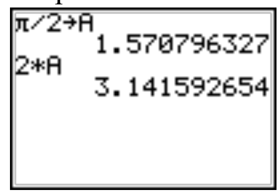
Par contre, certaines touches semblent manquer : on chercherait en vain les touches permettant de mettre en mémoire, puis de récupérer la valeur ainsi mémorisée. De même, on ne trouve pas de touche permettant de passer de *radians* à *degrés*, ou vice-versa. En fait, ces « absences » sont

[1] Pour être exact, il y a d'autres différences (comme la présence d'une horloge ou la qualité de l'affichage), mais celles-ci ne joueront aucun rôle dans nos propos.

[2] Par exemple, pour calculer sin(90), on presse successivement sur les touches **sin**, **9**, **0**, **)** et **ENTER**, et on voit apparaître *sin(90)* et une valeur approchée à l'écran. Sur certaines calculatrices, on taperait plutôt **9**, **0**, **sin** et =, et l'affichage *90* serait directement remplacé par le résultat du calcul.

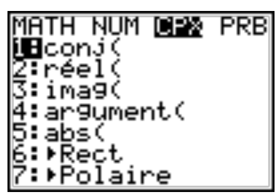
plutôt le signe d’une richesse : notre calculatrice dispose de plusieurs mémoires au lieu d’une seule, et ses fonctionnalités sont si nombreuses qu’elles ne peuvent être toutes activées au moyen d’une touche.

Voyons brièvement comment utiliser les diverses mémoires. Pour mettre une valeur en mémoire, il faudra taper le nombre en question (ou une expression permettant de le calculer), puis appuyer sur la touche **STO** (ce qui produira le caractère « → » à l’écran), et enfin indiquer le nom de la variable où stocker la valeur. Les noms permis sont en fait des lettres apparaissant en vert au-dessus de certaines touches. On les obtient en pressant d’abord sur la touche **ALPHA**, puis ensuite sur la touche correspondant à la lettre[1]. Par la suite, on pourra utiliser ces variables dans nos calculs, comme le montre l’exemple ci-contre.



Pour spécifier un calcul en *degrés* ou en *radians*, nous devons appuyer sur la touche **MODE**, puis amener le curseur clignotant (au moyen des touches fléchées) sur l’option désirée, en enfin confirmer notre choix en appuyant sur la touche **ENTER**. Nous voyons donc ici un premier exemple où une touche unique nous amène à un écran nous permettant de choisir entre plusieurs options. Notons en passant que l’affichage ci-contre est en français : nous verrons bientôt comment franciser (si désiré) les affichages de votre calculatrice. Bien évidemment, les noms des commandes sur ou au-dessus des touches resteront en anglais.

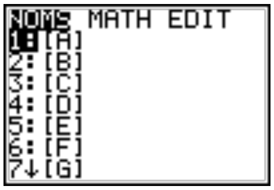
De la même façon, un appui sur la touche **MATH** provoque l’affichage du premier écran à gauche ci-dessous. Les flèches horizontales nous permettent de passer d’un menu à l’autre, tandis que les flèches verticales nous permettent de circuler parmis les items d’un menu donné. Pour sélectionner un item, il suffit de taper le caractère de gauche qui lui correspond, ou d’amener le curseur sur celui-ci et d’appuyer sur **ENTER**. Notons la dernière ligne des deux écrans de gauche, où le caractère identifiant le menu est suivi d’une flèche vers le bas plutôt que d’un deux-points : cela signifie qu’il y a d’autres items au-delà du septième, et qu’on peut y accéder au moyen des flèches verticales de notre calculatrice.

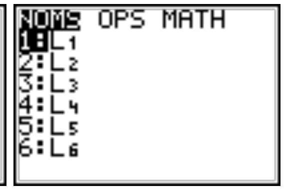


Plusieurs commandes de notre calculatrice, obtenues directement ou après appui sur la touche **2ND**, font apparaître de semblables écrans de menus. Mentionnons, par exemple, **MATRIX** (**2ND** + **x^{-1}**), **STAT**, **VARS** et **LIST** (**2ND** + **STAT**), dont on peut voir ci-dessous les écrans correspondants.

---

[1] Mentionnons également la présence de commandes supplémentaires apparaissant en bleu au dessus des touches. On y accède en appuyant au préalable sur la touche **2ND**. C’est ainsi qu’on a pu obtenir la constante π dans l’exemple ci-dessus. Une commande particulièrement importante est **QUIT** (**2ND**+**MODE**), qui nous permet en tout temps de revenir à l’écran de base.





Comme vous pouvez le voir, les commandes utilisables avec notre calculatrice sont nombreuses : on peut d'ailleurs le constater avec la commande **CATALOG** (**2ND** + **0**), qui énumère alphabétiquement toutes les commandes disponibles. Quand l'écran du catalogue est affiché, notre clavier est placé en mode alphabétique : un appui sur une touche nous amène à la première commande commençant par la lettre correspondante (en vert au-dessus), sans avoir besoin d'appuyer au préalable sur la touche **ALPHA**.

On peut même ajouter de nouvelles commandes à notre calculatrice, sous la forme de *programmes* ou d'*applications*. Les *programmes* sont écrits à l'aide de la calculatrice, dans un langage de programmation plutôt primitif, rappelant le *Basic*. On invoque la définition ou l'exécution d'un tel programme en appuyant sur la touche **PRGM**. Nous ne décrirons pas l'écriture de tels programmes dans ce livre.

Les *applications* sont des commandes que nous pouvons ajouter à nos calculatrices, mais qui doivent être créées par des programmeurs travaillant sur ordinateur[1]. On exécute une application en choisissant un item de l'écran invoqué par la touche **APPS**. L'écran ci-contre montre deux applications intéressantes disponibles : *Cabri Jr*, une version allégée du programme de géométrie dynamique que nous étudierons au chapitre 6, et *CelSheet*, un mini-tableur. Mentionnons aussi deux autres applications importantes : *Français*[2] et *CtlgHelp* (Catalog Help), qui permet d'obtenir de l'aide sur certains items de divers menus (dont le Catalogue) en appuyant sur la touche « + ».

On peut trouver des programmes et des applications sur le cédérom fourni à l'achat de la calculatrice et sur certains sites Internet, dont celui de *Texas Instruments* (www.education.ti.com - dont on obtient une version française en choisissant *Belgique*, *France* ou *Suisse* comme pays[3]). On verra à la section 5.3 comment les charger dans nos calculatrices.

Jetons maintenant un coup d'oeil aux capacités graphiques de notre calculatrice. Évoquons tout d'abord la diversité des représentations graphiques disponibles, en nous rappelant certaines options affichées à l'écran obtenu en appuyant sur le bouton **MODE** :

---

[1] Au moment d'écrire ces lignes, la création de telles *applications* devait se faire dans un environnement informatique sur plate-forme *Windows*.

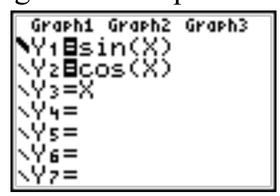
[2] **Pour franciser votre calculatrice**, il suffit d'exécuter l'application *Français* et de choisir l'option « *1:Français* ». Pour revenir à l'anglais, on choisira plutôt l'option « *2:English* ».

[3] Même si on nous promet que cela va venir, on chercherait en vain *Québec* ou *Canada français* dans les choix de pays. En fait, le *Canada* et les *États-Unis* se partagent présentement le même site (anglophone).

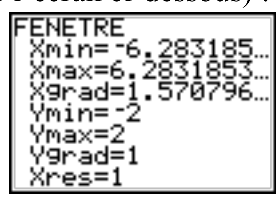
- La ligne 4 nous permet de choisir entre des représentations cartésiennes, paramétriques, polaires et des suites. À ceci il faudrait ajouter les possibilités supplémentaires rendues disponibles par la commande **DRAW** (obtenue via les touches **2ND** + **PRGM**).
- La ligne 5 nous permet de spécifier si nous voulons relier les points utilisés.
- La ligne 6 nous permet de spécifier l'ordre des tracés.

Choisissons pour l'instant de tracer séquentiellement des graphes cartésiens en reliant les points calculés. Pour ce faire, nous utiliserons les touches situées immédiatement au bas de l'écran.

La touche **Y=** permet de décrire les fonctions à considérer. Dans l'exemple ci-contre, on a spécifié que seules les deux premières seraient tracées (puisque le signe d'égalité correspondant est en blanc sur fond noir). Pour ce faire, en utilisant les touches fléchées, on a amené le curseur sur le signe d'égalité et appuyé sur **ENTER** pour en changer l'état. De même, on peut amener le curseur sur le petit dessin à gauche de chaque « Y » pour déterminer (en appuyant autant de fois que nécessaire sur la touche **ENTER**) le type de trait qui sera utilisé lors du tracé de la fonction



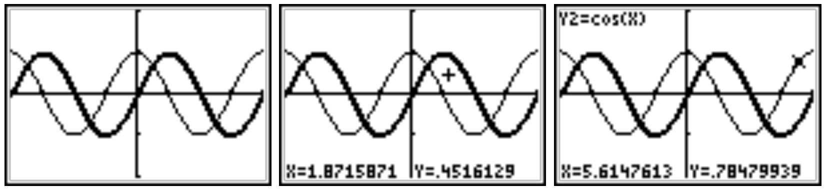
Il faut ensuite décrire la région où les fonctions seront tracées. On peut le faire en appuyant sur la touche **WINDOW**, qui fait apparaître un écran où l'on peut décrire la portion rectangulaire du plan à utiliser. Puisque nous étions en mode radians, nous avons choisi (voir l'écran ci-dessous) :



- Xmin = -2*π et Xmax = 2*π, avec des graduations à tous les π/2 (Xgrad = π/2). Notons ici que nous avons tapé les expressions ci-dessus (en utilisant le moins unaire [touche **(-)**]), et la calculatrice a calculé et affiché les valeurs numériques correspondantes.
- Ymin = -2 et Ymax = 2, avec des graduations à toutes les unités (Ygrad = 1).
- Xres = 1 : calculer un point à tous les pixels.
  Xres peut prendre des valeurs entières allant de 1 à 8.

Une autre façon de déterminer la région du tracé est d'utiliser la touche **ZOOM**, qui fait apparaître l'écran ci-contre, dont nous ne décrivons que la première commande. **Zboîte** permet de spécifier un sous-rectangle de l'écran graphique, qui occupera après coup tout l'écran graphique. Pour ce faire, on utilise les touches fléchées pour déterminer des sommets diagonaux de ce sous-rectangle (marqués via la touche **ENTER)**.

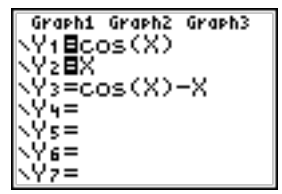
Pour tracer les graphiques, il suffit d'appuyer sur la touche **GRAPH** ou sur **TRACE**. Dans le cas de **GRAPH**, nos fonctions sont alors tracées, produisant un graphique semblable au graphique de gauche ci-dessous. Si nous utilisons alors les touches fléchées, nous ferons apparaître un curseur (ainsi que ses coordonnées - voir la figure du centre ci-dessous), que nous pourrons ensuite déplacer (toujours avec les touches fléchées). Par comparaison, la commande **TRACE** fait apparaître un écran comportant les courbes ainsi qu'un curseur que nous pouvons déplacer sur une des courbes avec les flèches horizontales, les flèches verticales permettant de « sauter » d’une courbe à l’autre (voir la figure de droite ci-dessous).



Comme on a pu le constater, ces calculatrices mettent à notre disposition de très nombreuses commandes que nous ne pouvons toutes expliquer ici. Heureusement, le cédérom accompagnant la calculatrice comporte une version électronique d’un manuel de plus de 700 pages (disponible en français), écrit à l’intention des enseignants de mathématiques et de sciences, où vous pourrez trouver réponse à la plupart de vos questions.

5.2 Résolution numérique d’équations

Dans la section 3.6, nous avons vu comment résoudre des équations comme $\cos(x) = x$ avec *Excel*, en utilisant la méthode de bissection. Nous allons maintenant décrire plusieurs approches pour résoudre une telle équation avec notre calculatrice. Commençons tout d’abord par définir les fonctions utiles (via la commande **Y=** [voir l’écran ci-contre]). Dans ce qui va suivre, nous supposerons que x est exprimé en *radians*.

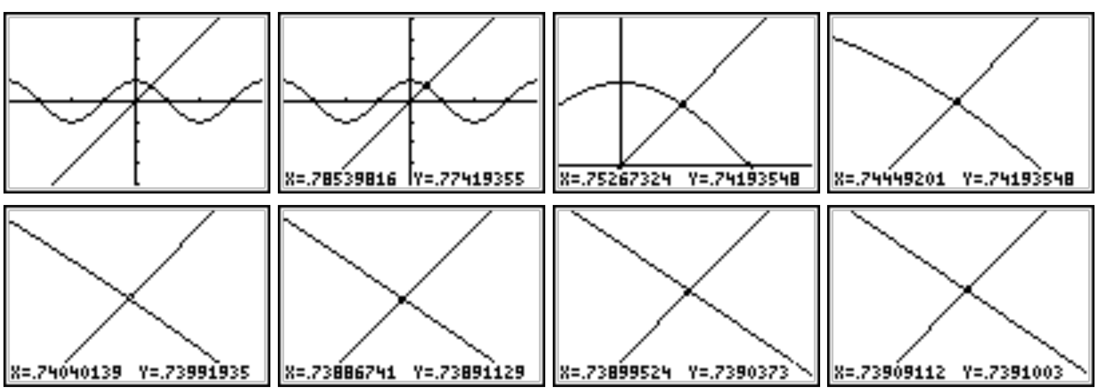


Résolution graphique manuelle

Nous traçons tout d’abord les deux premières fonctions dans une région déterminée via **ZOOM ➤ ZTrig**[1]. Nous obtenons alors le premier graphique ci-dessous, qui permet de nous faire une image globale de la situation. Nous constatons alors que notre équation n’a qu’une solution, Nous utilisons ensuite la commande **ZOOM ➤ Zoom +** pour nous approcher de plus en plus du point d’intersection par une succession d’étapes en deux temps :

- Au moyen des touches fléchées, nous approchons le curseur du point d’intersection.
- En appuyant sur ENTER, nous faisons un « zoom in » de facteur 4 (homothétie de rapport 4), centré sur la position du curseur.

De cette façon, nous obtenons la suite d’écrans ci-dessous.



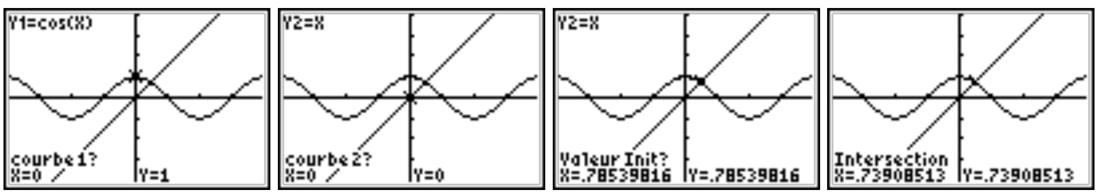
---

[1] La commande **ZTrig** détermine pour nous une région graphique bien adaptée aux fonctions trigonométriques de base. Les bornes de la région dépendent du mode (*degrés* ou *radians*) de la calculatrice, et peuvent être observées via la commande **WINDOW**.

En lisant les coordonnées du curseur, quand il est près du point d'intersection, on arrive à la conclusion que $\cos(x) = x \approx 0{,}739$. Pour obtenir une plus grande précision, il nous faudrait augmenter le nombre d'étapes. Mentionnons que, pour arrêter la commande **Zoom +**, il suffit d'appuyer sur la touche **GRAPH**. Notons en passant que nous aurions tout aussi bien pu rechercher le zéro de la fonction $\cos(x) - x$, toujours avec des « zoom in » successifs. Nous aurions aussi pu utiliser la commande **Zboîte** (qui permet de spécifier une région rectangulaire à agrandir) plutôt que **Zoom +**. Mais, quelle que soit la variante utilisée, notre démarche apparaît comme intuitivement claire.

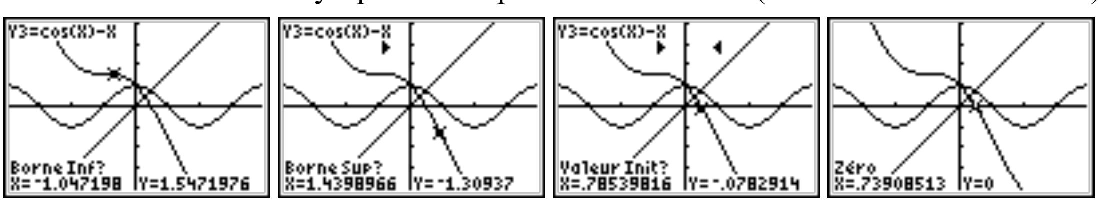
Résolution graphique automatique
Revenons tout d'abord à la région déterminée via **ZOOM ➤ ZTrig**. Puis faisons **CALC** (c.-à.-d. **2ND** + **TRACE**) ➤ **intersect**. Ceci nous conduira au premier écran graphique ci-dessous, où nous devrons successivement (voir les écrans ci-dessous)

- déterminer la première courbe à considérer pour le calcul du point d'intersection (identifiée via les flèches verticales et confirmée par un appui sur **ENTER**)
- déterminer la deuxième courbe à considérer pour le calcul du point d'intersection (identifiée via les flèches verticales et confirmée par un appui sur **ENTER**)
- déterminer une valeur initiale près du point d'intersection (identifié via les flèches horizontales et confirmé par un appui sur **ENTER**). Ceci est particulièrement important dans le cas où il y a plus d'un point d'intersection : nous devons alors indiquer lequel nous voulons obtenir.



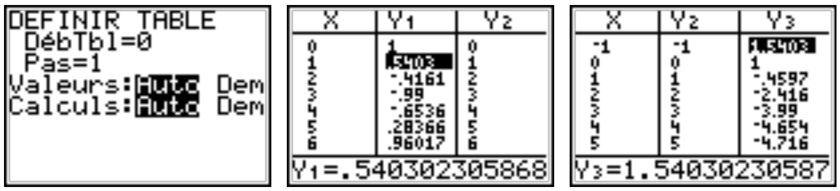
Par une méthode qui ne nous est pas révélée, la calculatrice détermine le point d'intersection et affiche sa position et ses coordonnées (voir l'écran de droite ci-dessus). Nous constatons ici que cette démarche est rapide et précise, mais que la calculatrice agit un peu comme une boîte noire ne fournissant que le résultat final à l'utilisateur.

Notons au passage une variante de cette méthode : si la troisième courbe était affichée, nous aurions pu utiliser la commande **CALC** (c.-à.-d. **2ND** + **TRACE**) ➤ **zéro** pour trouver un zéro de $\cos(x) - x$. Nous aurions alors dû spécifier la courbe, un intervalle de recherche et une valeur initiale (au cas où il y aurait plusieurs zéros dans l'intervalle) : la calculatrice aurait ensuite déterminé visuellement et analytiquement le point d'intersection (voir les écrans ci-dessous).

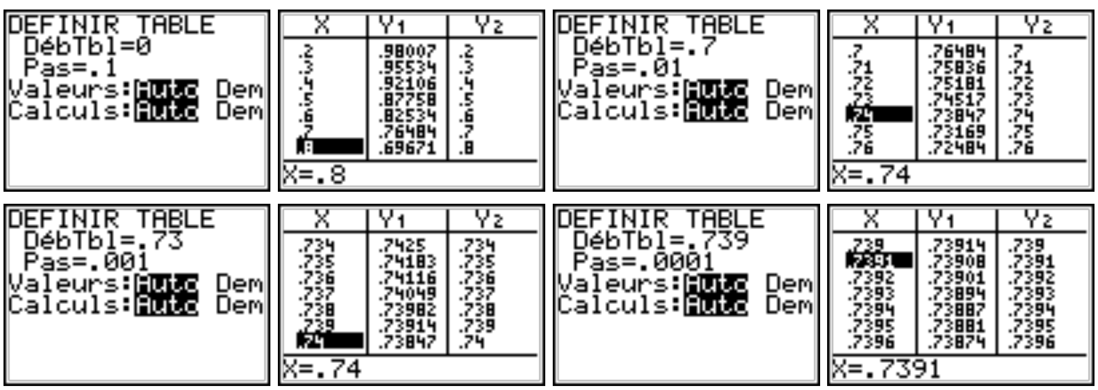


Résolution numérique manuelle
Nous allons maintenant trouver une valeur approchée du point d'intersection de nos deux premières courbes en utilisant plusieurs tables de valeurs. Chaque table de valeur se construit en deux temps.

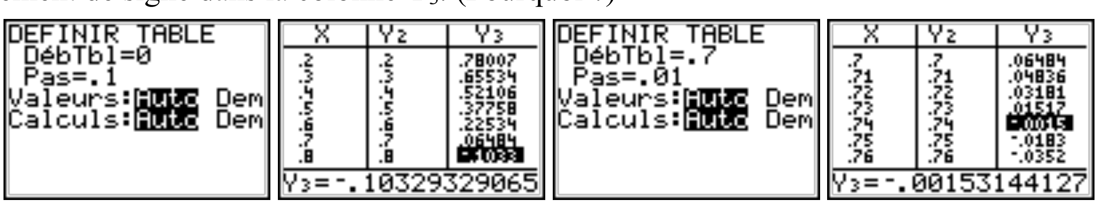
- Nous spécifions tout d'abord le point de départ et le pas utilisé dans notre table, à l'aide de la commande **TBLSET** (c'est-à-dire **2ND** + **WINDOW**).
- Nous affichons ensuite la table, à l'aide de la commande **TABLE** (c'est-à-dire **2ND** + **GRAPH**). Les touches fléchées permettent d'amener le curseur sur une valeur donnée (qui est alors affichée avec plus de précision en bas de l'écran). Elles permettent aussi de faire apparaître les portions non visibles (colonnes ou lignes) de la table. Veuillez noter que, pour faire apparaître la ligne située avant la première ligne de la table, il faut faire monter le curseur dans la première colonne.

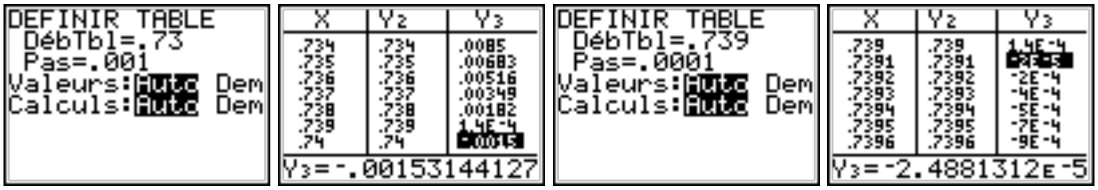


Voyons maintenant comment trouver une solution approximative de $\cos(x) = x$. Pour cela, nous examinerons les colonnes Y_1 et Y_2 et nous porterons plus particulièrement attention au cas où l'ordre numérique change d'une ligne à l'autre. Dans la table ci-dessus, c'est le cas de la première et de la deuxième ligne, où l'on a $Y_1(0) > Y_2(0)$ mais $Y_1(1) < Y_2(1)$: on aura donc une solution entre 0 et 1. (Pourquoi ?) On construira donc une nouvelle table de valeurs correspondant à l'intervalle [0,1] et on reprendra notre étude. Les copies d'écrans ci-dessous rendent compte de quelques étapes de cette démarche (la valeur **X=...** affichée sous la table correspondant à la ligne où le changement d'ordre a été constaté).



Soulignons qu'on peut, ici aussi, choisir de plutôt résoudre $\cos(x) - x = 0$. Dans ce cas, le zéro cherché sera toujours une valeur de la colonne des X entre deux lignes où l'on détecte un changement de signe dans la colonne Y_3. (Pourquoi ?)

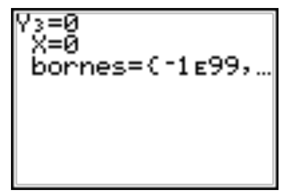


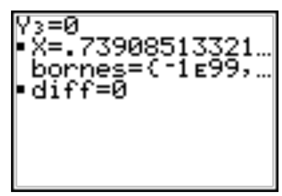


En y pensant bien, la méthode utilisée ci-dessus ressemble un peu à la méthode de bissection que nous avons utilisée avec *Excel*. Mais ici, chaque intervalle successif est subdivisé en 10 plutôt qu'en 2 pour trouver le prochain sous-intervalle. Contrastons au passage, d'une part, la simplicité conceptuelle mais aussi la lourdeur des manipulations requises pour la méthode manuelle sur calculatrice à, d'autre part, la facilité d'utilisation mais aussi la réflexion plus intense requise pour trouver les formules à mettre dans les cellules de notre feuille de calcul.

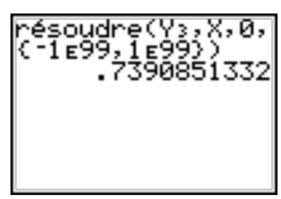
Résolution numérique automatique

Notre calculatrice comporte aussi une commande de résolution automatique d'équations. On l'invoque via **MATH ➤ Solveur...**, et l'écran ci-contre apparaît alors. Notons que l'équation à résoudre doit être de la forme $0 = f(x)$: dans notre cas, on pourra écrire « cos(X)-X » ou « Y_3 ». Soulignons ici que, pour la calculatrice, « cos(» est un caractère spécial, qui doit être engendré par la touche **COS** et non via quatre caractères séparés. De même pour le caractère « Y_3 », qui doit être produit via **VARS ➤ Y-VARS ➤ Fonction... ➤ Y_3**. Après avoir confirmé notre formule par **ENTER**, l'écran suivant apparaît. On doit alors entrer une valeur initiale pour X, à partir de laquelle se fera la recherche du zéro. Si on le désire, on peut aussi restreindre l'intervalle dans lequel se fera la recherche, qui est initialement « toute la droite réelle » $\left(\text{soit } \left[-10^{99}, 10^{99}\right] \text{ pour la calculatrice}\right)$. En remontant à la première ligne, on peut aussi corriger la fonction dont on recherche un zéro.





Quand nous sommes satisfaits des données affichées, **on amène notre curseur à la deuxième ligne** (*sinon ça ne fonctionnera pas*) et on utilise la commande **SOLVE** (c'est-à-dire **ALPHA** + **ENTER**) : après un court instant, l'écran ci-contre apparaît. La solution trouvée est affichée à droite du « X= », et les points de suspension signifient que la valeur comporte plus de chiffres que ceux qui sont affichés : on peut voir ces chiffres supplémentaires en utilisant les touches fléchées. Une valeur de *diff* qui serait différente de zéro indiquerait une imprécision dans la solution trouvée. Quand nous avons terminé, n'oublions pas de quitter le solveur, au moyen de la commande **QUIT** (**2ND** + **MODE**).



Une façon alternative de procéder est d'utiliser la commande à partir du catalogue : **CATALOG** (c'est-à-dire **2ND** + **0**) ➤ **résoudre(**. Cette commande a trois arguments obligatoires : l'expression dont on veut trouver un zéro, la variable par rapport à laquelle on veut résoudre, et une valeur initiale pour débuter les recherches. Le quatrième argument est facultatif : l'intervalle dans lequel se fera la recherche. L'écran ci-contre en montre l'utilisation dans notre cas.

Il ne faut pas croire que les méthodes de résolution de $\cos(x)=x$ exposées ci-dessus épuisent toutes les approches possibles avec notre calculatrice. On pourrait encore, par exemple, créer une feuille de calcul ad hoc dans l'application **CelSheet** (qui est un mini-tableur), ou même écrire un programme utilisant une méthode de résolution de notre choix, avec ou sans interactivité demandée à l'utilisateur. Notre but était plutôt de donner une idée de la diversité des approches à notre disposition quand on utilise une telle calculatrice.

5.3 Liens des calculatrices avec le monde extérieur

Comme nous le mentionnions à la section 5.1, nos calculatrices ne constituent pas des systèmes fermés : on peut notamment leur ajouter des programmes, des applications et des périphériques (voir le site de *Texas Instruments*). On peut aussi établir des liens entre

- deux calculatrices (commande **LINK**)
- une calculatrice et un ordinateur (avec le programme *TI-Connect*)
- une calculatrice et un dispositif de projection (tablette graphique *ViewScreen* ou interface vidéo *TI-Presenter*)
- une calculatrice et un appareil de mesure (CBR, CBL, etc.)
- plusieurs calculatrices et un ordinateur via un réseau local sans fil (*TI-Navigator*).

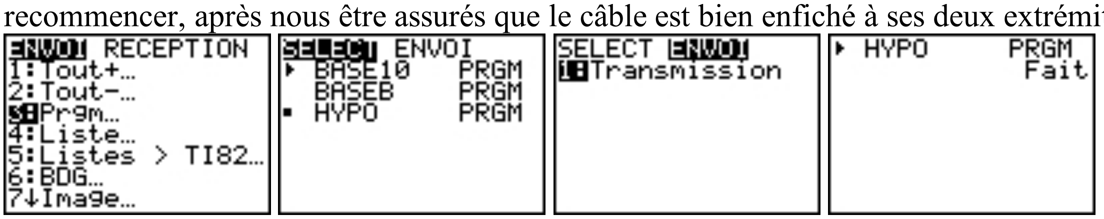
Figure 5.3.1 Les trois ports de communication d'une calculatrice TI-84 Plus Silver Edition (version professeur), de gauche à droite : le port USB, la sortie vidéo et le port I/O.

Pour réaliser ces divers branchements, nos calculatrices disposent de un à trois ports de communication (voir la figure 5.3.1). Toutes les calculatrices disposent d'un port I/O, permettant, avec les câbles appropriés (voir la figure 5.3.2), de relier une calculatrice à une calculatrice, un ordinateur, certains appareils de mesures, ou un réseau sans fil *TI Presenter*. Sur les modèles *TI-84Plus* et *TI-84Plus Silver Edition*, on retrouve aussi un port USB qui joue plus efficacement le même rôle. Sur les modèles pour enseignants, on dispose aussi d'une sortie vidéo permettant d'alimenter un dispositif de projection[1] : ceci est très utile pour le professeur voulant décrire certaines manipulations à sa classe[2]. Dans ce qui suit, nous nous contenterons de décrire les branchements calculatrice-calculatrice et calculatrice-ordinateur.

Figure 5.3.2 Les câbles de branchement I/O (en haut) et USB (en bas) : à droite, pour relier deux calculatrices; à gauche, pour relier calculatrice et ordinateur; au centre, pour transformer le port USB en sortie vidéo.

Voyons tout d'abord comment transférer des fichiers d'une calculatrice à une autre. Supposons que nous voulions transférer un petit programme (nommé HYPO) permettant de calculer des versements hypothécaires. Nous commençons par relier les deux calculatrices par le câble approprié (reliant les deux ports *I/O* ou les deux ports *USB*). Sur les deux calculatrices (émettrice et réceptrice), nous utilisons ensuite la commande **LINK** (**2ND** + **X,T,Θ,n**) :

- Sur la calculatrice émettrice, nous demeurons en mode **ENVOI** et nous choisissons d'abord **3 :Prgm...** Nous amenons ensuite le curseur face au programme à transmettre et nous le choisissons via un appui sur **ENTER**[3]. Nous passons ensuite en mode **ENVOI**, et nous choisissons **1 :Transmission**. Si tout se passe bien, nous obtiendrons alors l'écran de droite ci-dessous, qui confirme la bonne transmission du programme. Sinon, nous devrons recommencer, après nous être assurés que le câble est bien enfiché à ses deux extrémités.



[1] Sur les modèles pour élèves munis d'un port USB, on peut utiliser un adaptateur vidéo *Presentation Link* pour obtenir une sortie vidéo.

[2] Mentionnons également *TI Smartview*, un programme de simulation de la calculatrice pour ordinateur *Macintosh* ou *Windows*, qui permet de montrer non seulement l'écran mais aussi les touches de la calculatrice.

[3] Notons au passage que nous pourrions choisir de la sorte plusieurs programmes, et que la procédure serait la même pour tout fichier de notre calculatrice.

- Sur la calculatrice réceptrice, nous passons en mode **RECEPTION** et nous pressons sur **1 :Réception** au moment même où l'on déclenchera la transmission sur la calculatrice émettrice. Si un programme de même nom existe déjà, on nous demandera des instructions supplémentaires. Dans tous les cas, si tout se passe bien, nous recevrons confirmation du transfert.

Voyons maintenant comment échanger des fichiers entre une calculatrice et un ordinateur. Pour cela, nous allons utiliser le logiciel *TI Connect*[1], que nous supposerons avoir été installé au préalable. Nous devons tout d'abord relier les deux machines par le câble approprié (reliant le port *USB* de l'ordinateur au port *I/O* ou au port *USB* de la calculatrice), puis démarrer *TI Connect*. Comme nous le montre la figure 5.3.3, *TI Connect*[2] donne accès à une série de modules permettant notamment de mettre à jour le système d'exploitation de la calculatrice, de créer et de modifier diverses données avec l'ordinateur (en bénéficiant du confort accru de son clavier et de sa souris) plutôt qu'avec la calculatrice, ainsi que de transférer divers fichiers entre la calculatrice et l'ordinateur. C'est cette dernière possibilité qui retiendra notre attention.

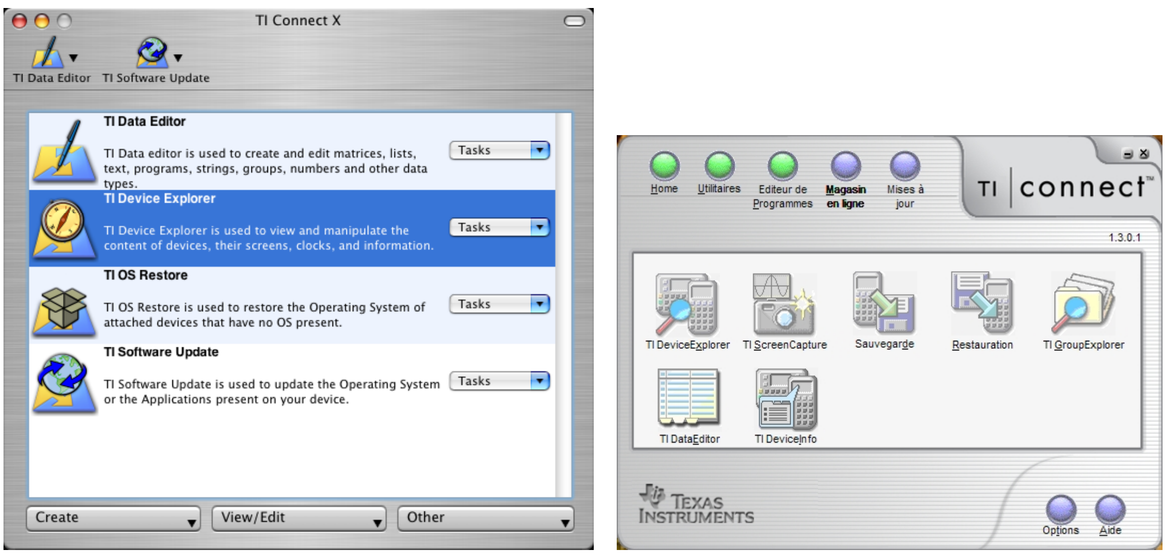


Figure 5.3.3 Les divers modules de *TI Connect*,
en version Macintosh (à gauche) et Windows (à droite).

Nous démarrons donc, par un double-clic, le module *TI Device Explorer*. Après avoir patienté quelques instants, le temps de permettre l'établissement de la communication entre la calculatrice

[1] Ce logiciel est fourni (en version *Macintosh* et *Windows*) sur le cédérom livré avec la calculatrice, et ses dernières versions sont disponibles sur le site de *Texas Instruments* (http://education.ti.com).

[2] Au moment d'écrire ces lignes, *TI Connect* n'était disponible qu'en version anglaise pour *Mac OS X*. Il est cependant disponible en français pour *Mac OS 9* et *Windows*.

et le module, nous obtenons une fenêtre semblable à l'une de celles de la figure 5.3.4, qui affiche tout le contenu (applications, programmes, variables, etc.) de la calculatrice, dans un ordre spécifiable (par nom, type ou taille, dans la version *Macintosh*) ou de façon hiérarchique (version *Windows*). Notons en passant qu'un clic sur l'entête d'une colonne modifiera en conséquence l'affichage des fichiers.

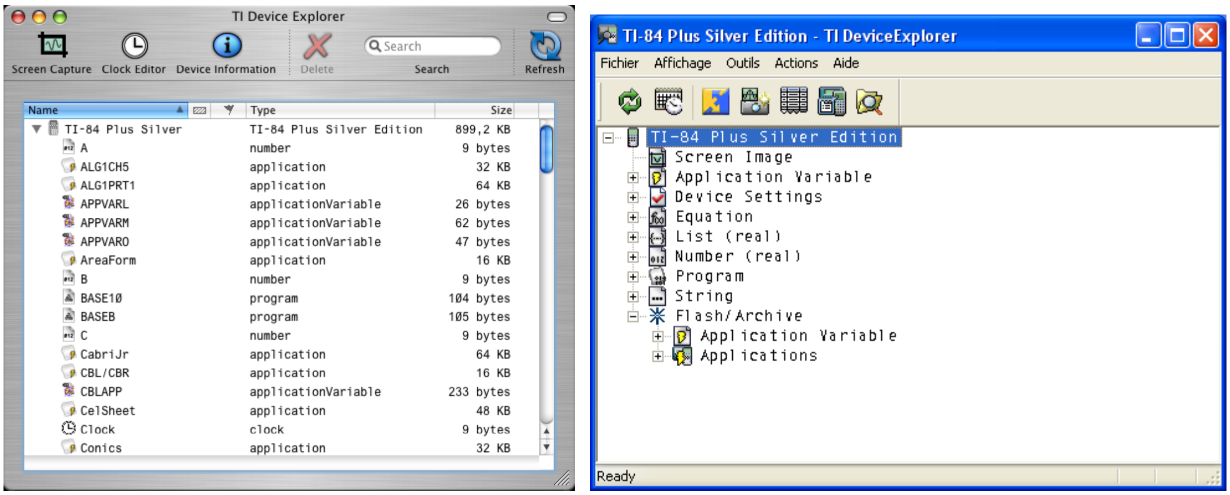


Figure 5.3.4 Le contenu d'une calculatrice, tel qu'affiché par le module *TI Device Explorer* de *TI Connect*, en version *Macintosh* (à gauche) et *Windows* (à droite).

Pour transférer un ou plusieurs fichiers de la calculatrice vers l'ordinateur, il suffit de les sélectionner dans la fenêtre de *TI Device Explorer*, puis de les faire glisser pour les déposer dans un dossier de l'ordinateur. De façon similaire, pour transférer un ou plusieurs fichiers de l'ordinateur vers la calculatrice, il suffit de les sélectionner puis de les faire glisser pour les déposer dans la fenêtre de *TI Device Explorer*. Soulignons ici qu'on ne peut transférer vers la calculatrice que des fichiers reconnus par celle-ci : applications, programmes, données dans un format standard (nombre, liste ou tableau de nombres, chaîne de caractères, etc.).

Cette possibilité de transfert de fichiers est intéressante à plus d'un égard. Bien sûr, elle nous permet de placer dans notre calculatrice des fichiers venant de l'extérieur (autres calculatrices, ordinateurs, cédéroms, Internet). Mais elle nous permet aussi de sauvegarder le contenu de notre calculatrice, que ce soit par souci de sécurité[1] ou pour faire place à de nouveaux fichiers.

Une autre caractéristique intéressante du module *TI Device Explorer* est de permettre de transférer une copie de l'écran de la calculatrice sous la forme d'une image, que l'on pourra, par exemple, insérer dans un document *Word* destiné aux élèves (via le menu « Insertion » ➤ « Image » ➤ « À partir d'un fichier... »). Pour ce faire, il suffit de cliquer sur le bouton « Screen Capture » (voir figure 5.3.4), qui fera apparaître la copie de l'écran actuel de la calculatrice dans une nouvelle fenêtre (voir figure 5.3.5). À partir de cette fenêtre de gestion des captures d'écran, nous pouvons sauvegarder les images dans des fichiers, encadrer l'image de l'écran, varier la

[1] Mentionnons que notre calculatrice comporte deux types de mémoires, qui réagissent différemment en cas de panne des piles : la mémoire vive (dont le contenu est perdu) et la mémoire d'archive (dont le contenu est conservé).

taille des images, demander de mettre à jour la capture d'écran, et même faire un film de l'écran de notre calculatrice pendant que nous l'utilisons (version *Macintosh* seulement).

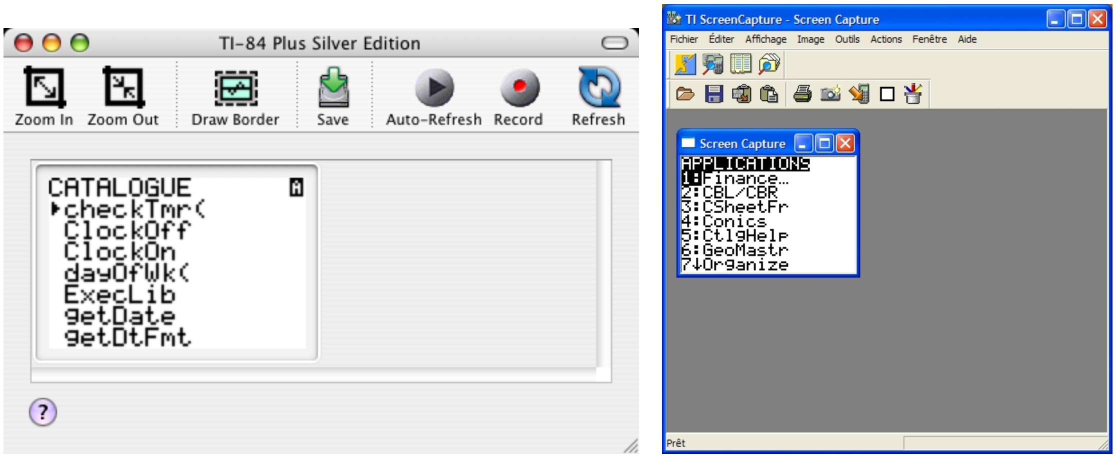
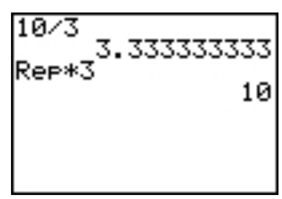


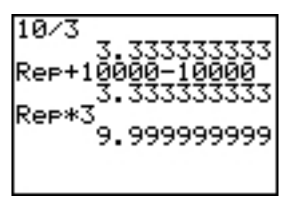
Figure 5.3.5 La fenêtre de gestion des captures d'écran, en version *Macintosh* (à gauche) et *Windows* (à droite).

Voila qui termine notre brève description de *TI Connect* et de son module *TI Device Explorer*. Comme toujours, nous vous référons à l'aide en ligne de *TI Connect* et de ses modules pour obtenir des renseignements supplémentaires.

5.4 Calculs numériques dans nos outils technologiques

Les calculatrices et les logiciels étudiés précédemment (*Word* [*LangageGraphique*] et *Excel*) ne font pas des calculs exacts : ils manipulent des nombres qui n'ont qu'un nombre fixe de décimales. Nous allons jeter un coup d'œil sur les conséquences de cet état des choses, en utilisant la calculatrice comme support de nos explications. En effet, il s'avère que la plupart des logiciels sur ordinateurs travaillent en *base deux*[1], alors que la plupart des calculatrices utilisent la *base dix*, ce qui facilitera nos explications.

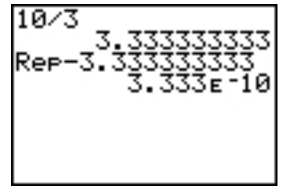


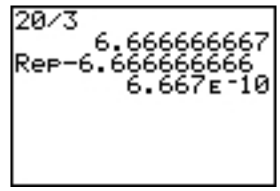


Commençons par faire quelques expériences avec notre calculatrice, dont certaines vous surprendront peut-être. Par exemple, divisons le nombre 10 par 3, et multiplions ensuite le résultat par 3. Le résultat final affiché par la calculatrice est 10 (voir ci-contre), et ceci devrait peut-être nous étonner. En effet si, au cours de la première division, la suite infinie de décimales a été tronquée à un nombre fini, la multiplication subséquente par 3 donnera 9.999...999 et non 10. Pour expliquer de semblables phénomènes, certains échafaudent parfois de mystérieuses théories selon lesquelles la calculatrice garderait trace de ses calculs passés, et pourrait ainsi produire des résultats exacts. Mais cette théorie est battue en brèche par l'expérience suivante (voir ci-contre) : si on ajoute puis on soustrait 10000 entre la division et la multiplication par 3, le résultat final n'est plus 10 mais bien

[1] On peut trouver une façon de déterminer la *base* utilisée par nos outils de calcul dans le livre de J.-M. Muller, *Arithmétique des ordinateurs*, Masson, 1989.

9.999999999 (un nombre de dix chiffres) : comment se fait-il que le résultat ne soit plus exact maintenant ?





L'explication de ces phénomènes repose sur le fait que la calculatrice n'affiche pas tous les chiffres qu'elle utilise. On peut mettre ceci en évidence par le calcul ci-contre : après avoir divisé 10 par 3, on soustrait la réponse **affichée** (qui comporte dix chiffres décimaux) : le résultat obtenu (3.333E-10) comporte 4 chiffres. On peut donc supposer que le nombre **calculé** initialement devait comporter 14 décimales, mais que seules les dix les plus significatives étaient affichées. Qui plus est, l'expérience ci-contre confirme que lors des calculs et des affichages, les nombres sont arrondis pour garder le plus de précision possible : le résultat de l'opération 20/3 effectué par la calculatrice est donc de 6.6666666666667 (nombre arrondi à 14 chiffres), et l'affichage 6.666666667 est arrondi à 10 chiffres. Notons en passant que l'arrondi lors du calcul à 14 chiffres suppose que le calcul se fait en utilisant *plus de 14 chiffres* (car il faut minimalement connaître le 15^{e} chiffre pour savoir arrondir le 14^{e}), mais que le résultat est par la suite placé dans une mémoire n'ayant de la place que pour 14 chiffres.

Diverses expériences comme celles ci-dessus nous amènent à formuler le modèle suivant : les nombres à virgule sont représentés dans la mémoire de notre calculatrice sous la forme de nombres dits « à virgule flottante » comportant 14 décimales et deux chiffres pour l'exposant

$$\pm c_1.c_2c_3c_4c_5c_6c_7c_8c_9c_{10}c_{11}c_{12}c_{13}c_{14}\times 10^{\pm d_1d_2}$$

Les calculs sur ces nombres de 14 chiffres sont effectués avec plus de précision, mais le résultat est ensuite arrondi à 14 chiffres. Pour l'affichage, on arrondit à 10 chiffres, sans pour autant modifier le nombre en mémoire.

Voyons quelle prédiction notre modèle fait si on voulait additionnner et soustraire 1000 au lieu de 10000 dans le second exemple de cette section.

Commande	Valeur calculée	Valeur affichée[1]
10 ÷ 3 **ENTER**	$+3.3333333333333\times 10^{+0}$	3.333333333
+ 1000 **ENTER**	$+1003.3333333333\times 10^{+0}$	1003.333333
- 1000 **ENTER**	$+3.3333333333000\times 10^{+0}$	3.333333333
× 3 **ENTER**	$+9.9999999999000\times 10^{+0}$	10
- 9.999999999 **ENTER**	$+9.0000000000000\times 10^{-10}$	9E-10

Détaillons le calcul de la dernière ligne :

$$\begin{array}{l} +9.9999999999000\times 10^{+0} \\ \underline{-9.9999999990000\times 10^{+0}} \\ +0.0000000009000\times 10^{+0} = 9\times 10^{-10} \end{array}$$

On peut vérifier que les affichages produits par notre calculatrice sont conformes aux prédictions. Nous aurons l'occasion d'appliquer à nouveau ce modèle dans les exercices.

[1] On suppose ici que le mode d'affichage de la calculatrice est normal (**MODE ➤ NORMAL**) et en virgule flottante (**MODE ➤ Flott**).

Prenant un peu de recul, on peut se demander pourquoi notre calculatrice n'affiche pas tous les chiffres qu'elle calcule. Nous croyons qu'il s'agit là d'une sage décision car, comme nous l'avons vu, plusieurs calculs s'accompagnent d'une légère perte de précision (qui affecte donc les dernières décimales), et il semble souhaitable de n'afficher que des décimales exactes. De plus, une précision de dix chiffres décimaux nous semble amplement suffisante dans la plupart des situations où nous aurons recours à une calculatrice. Mentionnons cependant que, lorsque le nombre et la complexité des calculs augmentent, il peut arriver que les erreurs de calculs s'accumulent au point que le résultat final devienne complètement faux. Nous en donnerons un exemple « spectaculaire » dans le projet 5.

Les calculs numériques dans les autres outils technologiques

On peut se demander si les remarques que nous venons de faire restent valables pour d'autres outils technologiques que les calculatrices *TI-84 Plus*. La réponse est à la fois oui et non...

Dans tous les outils technologiques (calculatrices ou ordinateurs), les calculs sont effectués par la section arithmétique d'un microprocesseur, qui manipule les nombres représentés soit comme des entiers, soit en virgule flottante. Dans tous les cas, le nombre maximal de chiffres est déterminé une fois pour toutes, et l'analyse ci-dessus s'applique en tous points. Seule la base utilisée pour représenter les nombres peut varier : elle est habituellement **deux**, dans le cas des ordinateurs, ou **dix**, dans le cas des calculatrices. (Dans ce dernier cas, si on veut être plus exact, les calculs à la base dix sont effectués en codant chaque chiffre décimal à l'aide de 4 chiffres binaires, selon un mode dit « décimal codé binaire ». Mais ceci revient conceptuellement au même.)

Les divers logiciels (tant des calculatrices que des ordinateurs) ont donc accès à cette arithmétique prédéfinie au niveau du matériel électronique, et presque tous effectuent tous leurs calculs de cette façon. Il existe cependant des exceptions.

- Les calculatrices ayant un mode *fractions* ne représentent pas celles-ci par des nombres en virgule flottante, mais plutôt comme un couple d'entiers. Les opérations arithmétiques sont donc effectuées sur ces entiers, en suivant les règles habituelles pour les fractions.
- Les logiciels fonctionnant de façon *symbolique* ne représentent pas les expressions arithmétiques par des nombres en virgule flottante, mais bien comme des expressions mathématiques à simplifier **sans perte d'exactitude**. Par exemple, si nous demandons au logiciel *Maple* ou à la calculatrice *Voyage 200* d'évaluer l'expression $\sqrt{8}+\sqrt{18}$, nous obtiendrons $5\sqrt{2}$ comme réponse. Si nous le désirons, nous pouvons ensuite demander une **valeur approchée** de cette réponse (en spécifiant même, dans le cas de *Maple*, le nombre de décimales requises). Voici une valeur approchée comportant 100 chiffres décimaux 7.0710678118654752440084436210484903928483593768847403658833986899536623923105351942519376716382078 65.

Tout ceci est possible car les logiciels de calcul symbolique peuvent choisir d'ignorer les instructions arithmétiques intégrées dans les microprocesseurs et d'utiliser les autres commandes du microprocesseur pour traiter à leur guise les expressions symboliques ou numériques de façon soit à les transformer en expressions équivalentes, soit à les évaluer avec un degré de précision déterminé par l'utilisateur et non par la machine.

5.5 Représentations graphiques dans nos outils technologiques

L'écran de nos calculatrices est de petite taille (une diagonale mesurant environ 7 cm) et comporte un nombre plutôt limité de pixels (63 lignes de 95 pixels[1]). En conséquence, nous obtenons des graphiques moins fidèles que ceux produits sur des écrans d'ordinateurs, et les pixels utilisés deviennent très apparents (voir, par exemple, la figure 5.5.6). Ceci va nous inciter à poursuivre notre étude des représentations technologiques des fonctions réelles, que nous avions amorcée à la section 4.2.

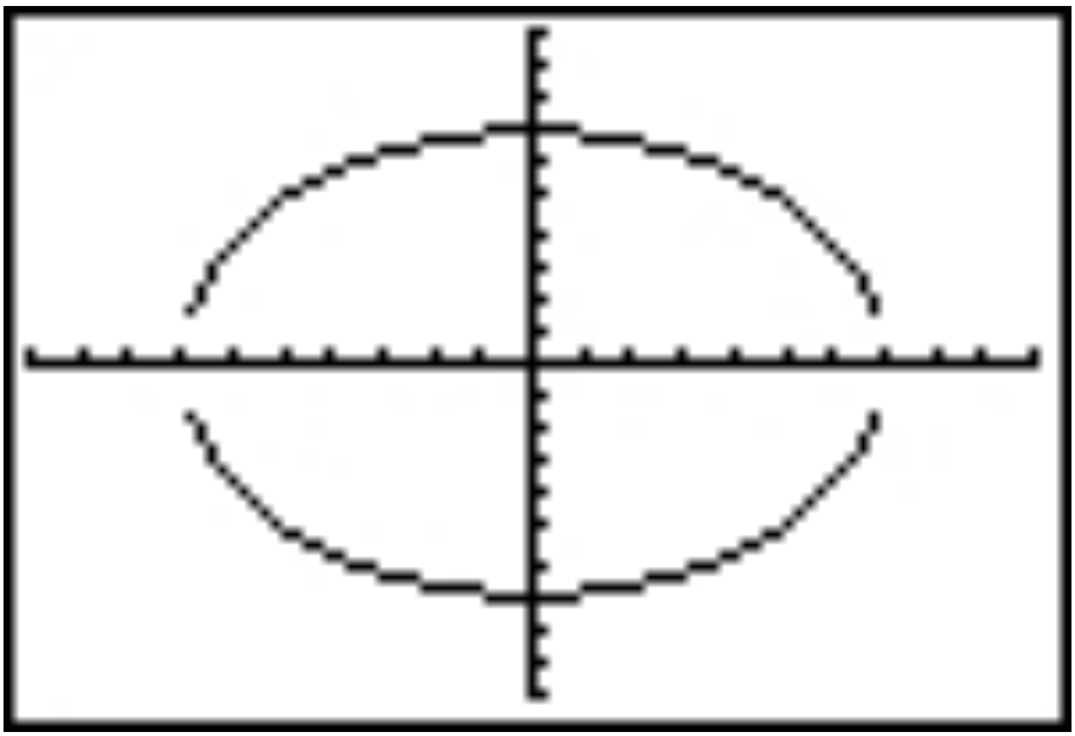


Figure 5.5.6 Le cercle $x^2 + y^2 = 7^2$ représenté à l'aide des deux fonctions $y_1 = \sqrt{7^2 - x^2}$ et $y_2 = -\sqrt{7^2 - x^2}$ dans la région obtenue via la commande *ZStandard*.

Les représentations graphiques dans les autres outils technologiques

Il n'y a pas si longtemps, les ordinateurs produisaient leurs graphiques en pilotant vectoriellement le faisceau d'électrons de tubes à rayons cathodiques de façon à produire diverses courbes qu'il fallait continuellement retracer pour qu'elles restent visibles.

Aujourd'hui, sauf peut-être quelques rarissimes exceptions, tous les écrans utilisés par les calculatrices et les ordinateurs (que ce soit des tubes à écrans cathodiques ou des écrans à cristaux liquides) représentent les graphiques à l'aide d'une grille rectangulaire de pixels. En conséquence, même dans les cas où les représentations sont plus fines que celles des calculatrices, les divers algorithmes évoqués dans cette section restent applicables.

[1] En fait, il s'agit plutôt de 64 lignes de 96 pixels. Mais ces pixels supplémentaires sont utilisés pour indiquer que la calculatrice est occupée, et ne sont donc pas disponibles pour les graphiques.

Posons-nous la question de savoir quels pixels dessiner pour représenter chacune des deux fonctions ci-dessus. Pour répondre à cette question, il nous faut tout d'abord déterminer comment établir une correspondance entre la région mathématique déterminée par **ZStandard** (soit $[-10,10]\times[-10,10]$[1]) et les 63×95 pixels de l'écran de notre calculatrice (voir la figure 5.5.7). Convenons au départ de repérer ces pixels par un système d'axes « écran », assignant les coordonnées (1,1) au pixel en bas à gauche, et (95,63) au pixel en haut à droite. En fait, on peut imaginer chaque pixel comme un carré unité, qu'on identifie par les coordonnées de son point central. Quand on parlera de « la distance entre deux pixels », on voudra dire « la distance entre les points centraux de ces pixels » : même s'ils se touchent, la distance entre deux pixels consécutifs (sur une même horizontale ou verticale) sera donc 1.

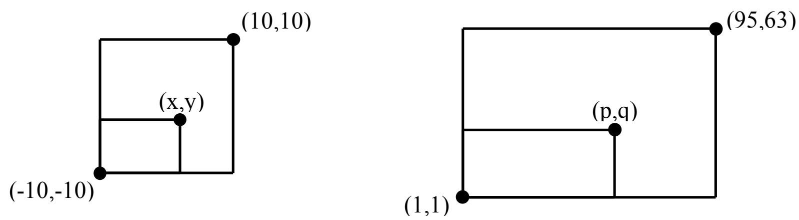


Figure 5.5.7 Représentations de la région mathématique (à gauche) et l'écran de la calculatrice (à droite).

Généralisons notre étude à des régions autres que celle obtenue via la commande ZStandard et des écrans autres que celui de notre calculatrice (voir la figure 5.5.8) : elle sera donc applicable à d'autres calculatrices ou ordinateurs.

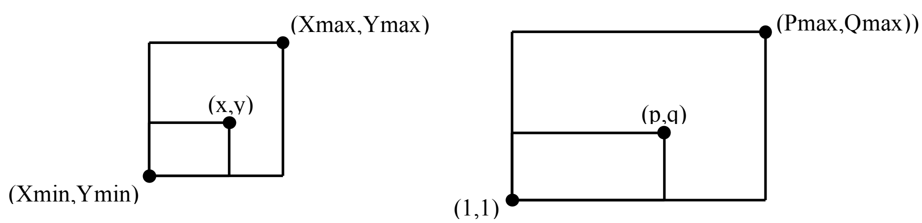


Figure 5.5.8 Correspondance entre les points (x,y) de la région mathématique et les pixels (p,q) de l'écran.

En général, les deux graphiques de la figure 5.5.8 ne seront pas semblables : ils ne le sont pas même dans le cas particulier de la figure 5.5.7. Mais il semble naturel d'exiger une « proportionnalité horizontale » : le rapport

$$\frac{\text{distance du point au côté gauche du rectangle}}{\text{longueur de la base du rectangle}}$$

doit être préservé d'un graphique à l'autre. Ceci se traduit par l'égalité suivante :

$$\frac{x-\text{Xmin}}{\text{Xmax}-\text{Xmin}}=\frac{p-1}{\text{Pmax}-1}.$$

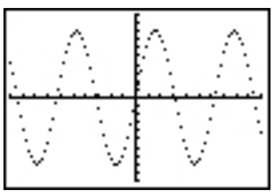
[1] Rappelons que $[-10,10]\times[-10,10]=\{(x,y):-10\le x\le 10 \text{ et } -10\le y\le 10\}$.

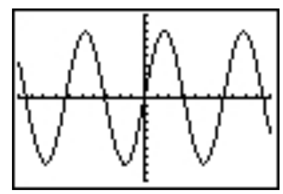
De la même façon, pour conserver une « proportionnalité verticale[1] », nous devrons avoir :

$$\frac{y-\text{Ymin}}{\text{Ymax}-\text{Ymin}}=\frac{q-1}{\text{Qmax}-1}.$$

Voyons maintenant comment notre calculatrice trace le graphe cartésien d'une fonction *f* donnée.

- Pour chaque colonne[2] (de numéro p, variant de 1 à *Pmax*), on calcule la valeur correspondante de x : $p \mapsto x = \text{Xmin} + (\text{Xmax}-\text{Xmin})\dfrac{p-1}{\text{Pmax}-1}$.
- On utilise alors la fonction *f* pour calculer la valeur de y : $x \mapsto y = f(x)$.
- On peut alors déterminer le numéro de ligne q où le pixel devra être tracé :
  $$y \mapsto q = 1 + (\text{Qmax}-1)\frac{y-\text{Ymin}}{\text{Ymax}-\text{Ymin}}.$$
  Notez que la valeur de q ne sera pas toujours entière et qu'il faudra l'arrondir (puisque q doit dénoter la ligne d'un pixel).





Mais ce n'est pas tout : via le menu **MODE**, l'utilisateur a pu choisir l'option *Relié* [voir écran de droite] ou *NonRelié* [voir écran de gauche]. Dans le premier cas, la calculatrice devra ajouter des pixels supplémentaires pour relier entre eux (par des « segments ») les pixels dont nous venons de décrire le mode de positionnement. Le lecteur intéressé à en savoir plus sur les diverses façons de procéder pourra consulter la page web **http://www.math.uqam.ca/_boileau/Segments.html,** où il trouvera un texte explicatif destiné à des élèves de niveau secondaire, ainsi que des programmes de démonstration du phénomène (voir la figure 5.5.9).

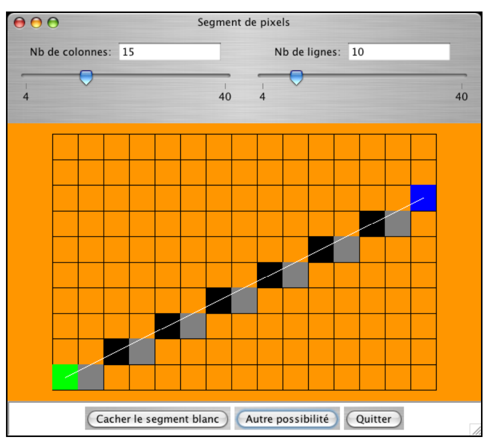


Figure 5.5.9 Écran d'un applet Java permettant de relier interactivement deux pixels par un « segment continu » de pixels.

[1] Soulignons en passant que les facteurs de proportionnalité « horizontale » et « verticale » ne sont pas nécessairement égaux.

[2] Du moins si la variable *Xres* (du menu affiché via la touche **WINDOW**) vaut 1. *Xres* peut prendre toutes les valeurs entières de 1 à 8. Si *Xres* vaut n, on trace les pixels aux colonnes 1, $1+n$, $1+2n$, ... Ceci devient clairement apparent si on choisit l'option *NonRelié* dans le menu **MODE**.

Afin de vous permettre d'expérimenter le mécanisme de construction d'un graphe cartésien de fonction, nous mettons à votre disposition un classeur *Excel* nommé **PixelsGraphiques.xls**, dont nous pouvons voir une fenêtre à la figure 5.5.10.

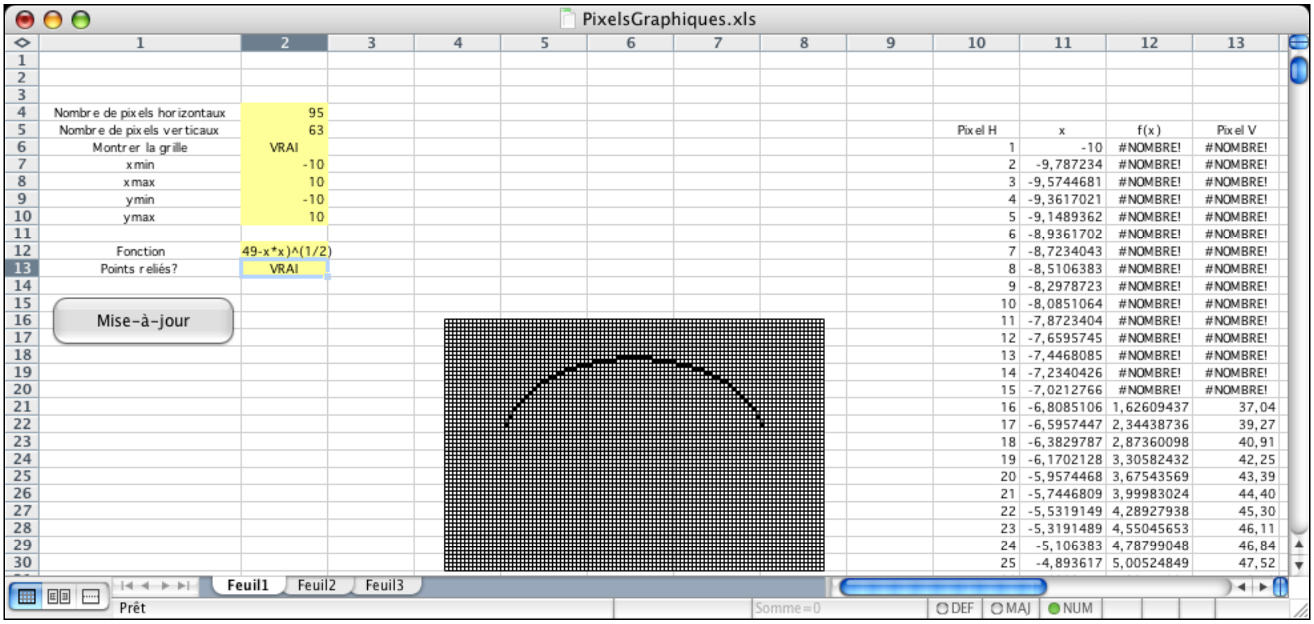
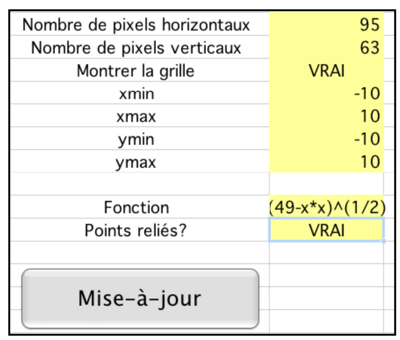
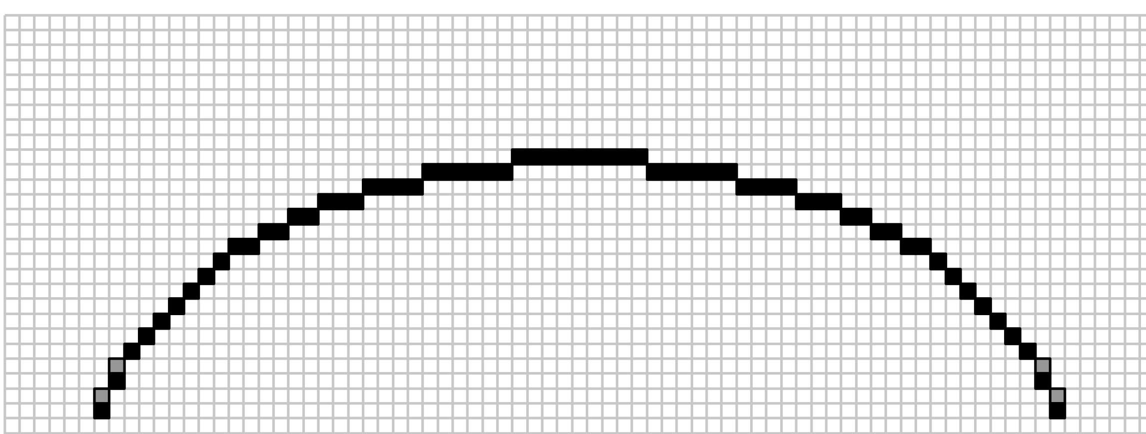


Figure 5.5.10 Tracé du graphe de la fonction $y_1 = \sqrt{7^2 - x^2}$ à l'aide du classeur *PixelsGraphiques.xls*.

Le fonctionnement de **PixelsGraphiques** est très simple : on donne les valeurs désirées aux divers paramètres (ci-contre), puis on clique sur le bouton « Mise-à-jour ». Le graphe cartésien correspondant est alors produit, ainsi qu'une table des valeurs de x, y et q pour toutes les colonnes p du quadrillage. Nous avons reproduit ci-dessous un fragment du graphe cartésien engendré pour la fonction $y_1 = \sqrt{7^2 - x^2}$ dans la région donnée par **ZStandard** : les pixels du graphe sont en noir, tandis que les 4 pixels les reliant avec des « segments informatiques » sont grisés.





Examinons maintenant un fragment de la table de valeurs (voir figure 5.5.11). On peut y lire à la ligne 20 de la feuille qu'encore à la 15^e colonne, la valeur correspondante de x (soit -7,0212766) reste inférieure à -7, et qu'en conséquence la fonction n'est pas définie en ce point x, pas plus que la ligne du pixel à tracer. À la ligne 21 de la feuille, par contre, on passe à la 16^e colonne, qui correspond à une valeur de x (soit -6,8085106) supérieure à -7 : la fonction est définie en ce point et vaut 1,62609437, et on doit tracer le pixel correspondant à la ligne 37 (l'arrondi de 37,04) de cette 16^e colonne de l'écran. Soulignons que la variable x ne prend pas la valeur -7, qui ne correspond pas à une valeur entière pour Pixel H (la colonne du pixel) : c'est ce qui explique que le graphe commence non pas au point (-7,0), mais plutôt au point (-6,8085106 , 1,62609437). Et comme une explication très semblable s'applique aussi pour la fonction $y_2 = -\sqrt{7^2 - x^2}$, on comprend pourquoi il y a un « trou » à la gauche du graphique de la figure 5.5.6. De la même manière, on peut expliquer la présence du « trou » de droite.

PixelsGraphiques.xls

	10	11	12	13
4				
5	Pixel H	x	f(x)	Pixel V
19	14	-7,2340426	#NOMBRE!	#NOMBRE!
20	15	-7,0212766	#NOMBRE!	#NOMBRE!
21	16	-6,8085106	1,62609437	37,04
22	17	-6,5957447	2,34438736	39,27
23	18	-6,3829787	2,87360098	40,91

Feuil1 Feuil2 Feuil3

Prêt

Figure 5.5.11 Fragment de la table de valeur créée par *PixelsGraphiques.xls* pour la fonction $y_1 = \sqrt{7^2 - x^2}$ dans la région $[-10,10]\times[-10,10]$.

Il peut être instructif de comparer les graphes obtenus avec la calculatrice (en utilisant la commande **TRACE**) et avec *PixelsGraphiques* (en jetant un coup d'œil sur la table des valeurs) : on constate alors que les pixels calculés dans les deux cas correspondent avec précision, mais que les pixels de « remplissage » (destinés à relier par des « segments » les pixels « séparés ») ne correspondent pas toujours. Dans le projet 2, vous pourrez comparer les algorithmes utilisés dans les deux cas.

Les lecteurs intéressés pourrons examiner comment *PixelsGraphiques.xls* a été programmé. Ils constaterons alors qu'il a été construit en se basant sur une adaptation de *LangageGraphique* pour *Excel* : il aura suffi d'ajouter des instructions décrivant où et comment tracer les « pixels ».

On peut se demander rétrospectivement s'il était nécessaire d'étudier en détail l'algorithme de tracé de fonctions utilisé par notre calculatrice (et par nos outils technologiques, en général). Il nous a semblé intéressant de le faire pour deux raisons. Premièrement, c'est une clé pour comprendre et pouvoir expliquer (à certains de nos élèves plus curieux) certaines caractéristiques a priori bizarres des graphes cartésiens produits par nos outils technologiques. Et, deuxièmement, ça nous semble un exemple à la fois simple et intéressant d'application des mathématiques à l'informatique. On pourrait rétorquer que c'est ultimement une application des mathématiques aux mathématiques elles-mêmes, puisqu'il est question de tracer des graphes cartésiens. Mais ce serait oublier que les techniques décrites ici sont à la base de l'infographie, qui dessert tous les utilisateurs (ingénieurs, architectes, artistes, etc.) d'applications graphiques.

Une calculatrice éducative ?

Il y a déjà plusieurs années de cela, lors d'une rencontre du *NCTM* (*National Council of Teachers of Mathematics*), un professeur de mathématiques raconta avoit été consulté à propos des caractéristiques que devrait posséder la calculatrice *TI-81*, et nous confia avec fierté avoir réussi à convaincre les ingénieurs de *Texas Instruments* de ne pas inclure de touche « SOLVE », qui aurait permis aux élèves de résoudre sans comprendre. Aujourd'hui la *TI-84 Plus*, descendante en ligne directe de la *TI-81*, comporte une telle « touche » (MATH ➤ Solveur...). Est-ce à dire que les pédagogues ont finalement perdu la bataille ?

En matière pédagogique, rien n'est aussi simple. D'une part, on peut aussi imaginer des activités pédagogiquement intéressantes utilisant un tel solveur automatique, notamment en mathématiques appliquées où la modélisation conduit souvent à des équations qu'on ne sait pas résoudre symboliquement. Et d'autre part, on ne peut soutenir que nos élèves doivent toujours tout comprendre : à ce compte là, nos calculatrices ne comporteraient pas de touches permettant de calculer les fonctions transcendantes (trigonométriques, logarithmiques, exponentielles, etc.).

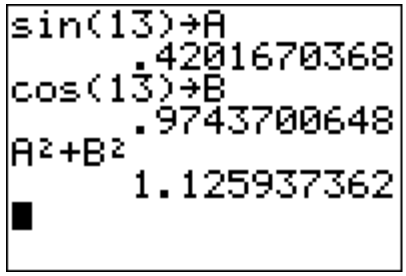
En fait, d'un point de vue pédagogique, le contexte d'utilisation de l'outil est au moins aussi important que l'outil lui-même. Sur le site de *Texas Instruments* (**education.ti.com**), on peut trouver de nombreuses ressources utiles pour employer ces calculatrices dans un contexte pédagogique : matériel pour projeter l'écran d'une calculatrice pour toute une classe, réseau sans fil pour faciliter la communication, fiches d'activités, cahiers pédagogiques, forums de discussion (tant pour les élèves que pour les enseignants), etc. En ce sens, la calculatrice *TI-84 Plus* est incontestablement un outil éducatif.

5.6 Exercices

1- *Utiliser des variables et différents modes de calculs*

Placez la calculatrice en mode de calcul en radians, puis stockez la valeur $\frac{\pi}{2}$ dans une variable nommée A. Ensuite, stockez $\sin(A)$ dans une variable nommée B.
Placez maintenant la calculatrice en mode de calcul en degrés, puis stockez $\sin(A)$ dans une variable nommée C.
Comparez les valeurs des variables B et C. Pourquoi sont-elles différentes?

2- *Utiliser des variables et différents modes de calculs*
Un élève, perplexe, vous montre sa calculatrice qui affiche l'écran ci-dessous :



Donnez une explication rationnelle au problème apparent.

3- *Tracer des graphes et trouver les coordonnées de points caractéristiques*
Placez la calculatrice en mode de calcul en radians.

a) Faites tracer le graphe de la fonction $f(x) = \sin(x) + \cos(x)$ pour x dans l'intervalle $[-\pi, \pi]$.
b) Sans faire de zoom, utilisez le curseur pour évaluer le maximum de *f(x)* dans l'intervalle donné, ainsi que l'abscisse x pour laquelle il est atteint.
c) Utilisez **CALC ➤ maximum** pour comparer les deux valeurs obtenues en b) avec celles données par la calculatrice.
d) Toujours sur le même intervalle, ajoutez les graphes des fonctions cos*(x)* et sin*(x)*. Le maximum de *f(x)* vous semble-t-il correspondre à une abscisse ayant une signification particulière pour les fonctions cos(x) et sin(x)? Faites une conjecture et utilisez la calculatrice pour la vérifier.

4- *Approximer les coordonnées de l'intersection de deux graphes*
Placez votre calculatrice en mode de calcul en radians. En utilisant **ZTrig** pour déterminer la fenêtre, faites tracer les graphes des fonctions $\cos(x)$ et $\sin(x)$.

a) Quel est le domaine utilisé par la calculatrice?
b) Utilisez le zoom pour déterminer les coordonnées, à 0,001 près, du point d'intersection dont l'abscisse est dans $[0, \pi]$. Justifiez votre approximation!
c) Utilisez maintenant **CALC ➤ intersect** pour trouver les coordonnées du point d'intersection. Le résultat confirme-t-il votre approximation obtenue en b) ?

5- *Déterminer le nombre de solutions d'une équation*

Utilisez votre calculatrice pour déterminer le nombre de solutions de l'équation $\sin(x) = \dfrac{x}{10}$.

Justifiez votre méthode!
Plus généralement, analysez le nombre de solution possibles de l'équation $\sin(x) = ax$, $a > 0$, en fonction de la valeur de a. Calculez numériquement valeurs de a pour lesquelles il y a 3, 5 ou 7 solutions.

6- *Déterminer le nombre de solutions d'une équation*
Existe-t-il des valeurs positives de a pour lesquelles l'équation $a\,x^2 = \sin(x)$ possède exactement 3 solutions sur la demi-droite $\left]0,\infty\right[$. Utilisez la calculatrice et le raisonnement pour justifier votre réponse.

7- *Charger des programmes à partir d'un ordinateur*
Pour cet exercice nous supposons que vous avez accès à un ordinateur relié à l'Internet et sur lequel le logiciel *TI Connect* est installé.
À partir de l'ordinateur, allez sur le site du livre, dans le dossier des exemples du chapitre, téléchargez les programmes « **BASE10** » » et « **BASEB** », puis transférez-les sur votre calculatrice
Pour vérifier que tout a bien fonctionné, débranchez votre calculatrice de l'ordinateur, appuyez sur la touche **PRGM** et vous devriez voir « **BASE10** » » et « **BASEB** » dans la liste des programmes.
Utilisez « **BASEB** » pour convertir le nombre décimal 43981 en base 16, puis utilisez « **BASE10** » pour faire la conversion inverse.

8- *Mettre en relation l'apparence d'un graphe et la région où il est tracé*
Utilisez la calculatrice pour tracer le graphe de la fonction $y = x + 20$ dans la région déterminée par « **ZStandard** ». Expliquez pourquoi presque rien n'apparaît à l'écran.

9- *Comprendre comment la calculatrice conserve les nombres*
Nous savons tous, depuis longtemps, que pour trois nombres quelconques a, b, c on a $a + (b + c) = (a + b) + c$. Pourtant, dans certaines situations, cette propriété semble être mise en défaut par la calculatrice. Par exemple :

- Stockez les nombres 100000, $\sqrt{2}$ et -100000 dans trois variables A, B et C respectivement.
- Stockez A+(B+C) dans une variable D et (A+B)+C dans une variable F. On devrait avoir $D = A+(B+C) = (A+B)+C = F$, ou $D - F = 0$.
- Évaluez $D - F$.

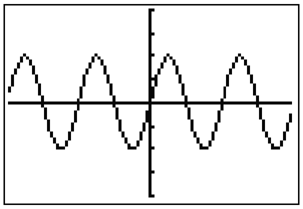
Utilisez le modèle numérique développé dans la section 5.4 pour expliquer le résultat.

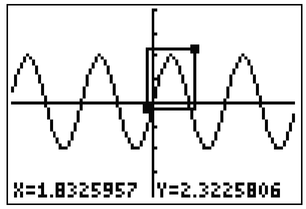
10- *Comprendre comment la calculatrice trace un graphe*
Utilisez la calculatrice pour tracer le graphe de la fonction $y = \dfrac{x^2 - 2x - 8}{x - 4}$ dans la région déterminée par « **ZStandard** ».

a) En $x = 4$ y n'est pas définie, pourtant le graphe n'y a pas de « trou », pourquoi?
b) Si x varie sur $[-4, 4]$ ou sur $[-47, 47]$ le trou apparaît, pourquoi? (Lorsque ce n'est pas évident à l'œil, vous pouvez utiliser « **TRACE** » pour vous convaincre.)

11- *Comprendre comment la calculatrice trace un graphe*
Placez la calculatrice en mode de calcul en radians, puis tracez le graphe de la fonction $y = 2\sin(50x)$ dans la région déterminée par « **ZTrig** ». Vous obtenez l'écran de gauche.

Faites, maintenant, un Zoom semblable à celui indiqué sur l'image du centre et tracez le graphe.





Pourquoi l'écran obtenu ressemble-t-il à la figure de droite?

12- *Comprendre comment la calculatrice trace un graphe*

Utilisez la calculatrice pour tracer le graphe de la fonction $y = \dfrac{x^2 + x - 1}{x - 1}$ dans la région déterminée par « **ZStandard** ». La fonction y n'est pas définie en $x = 1$ et en fait y possède une asymptote verticale. Expliquez pourquoi le graphe obtenu ne reflète pas correctement ce phénomène.

13- *Comprendre comment la calculatrice trace un graphe*

Utilisez la calculatrice pour tracer les graphes des fonctions $y_1 = \sqrt{9 - x^2}$ et $y_2 = -y_1$ dans la région $[-15, 15] \times [-4, 4]$. On devrait obtenir le graphe du cercle $x^2 + y^2 = 9$ (Pourquoi?).

- Expliquez les différences entre le résultat obtenu et le résultat espéré.
- Comment remédier à ce problème?

5.7 Projets

1- *Résoudre numériquement des équations*

Dans la section 5.2, vous avez vu quatre méthodes pour résoudre numériquement des équations à l'aide de la calculatrice :

- en utilisant un graphique pour visualiser les solutions, un curseur pour afficher les coordonnées et des zooms pour améliorer la précision
- en utilisant une recherche graphique automatique (commande **CALC ➤ zero**)
- en utilisant une résolution « automatique » (commandes **MATH ➤ MATH ➤ Solver**)
- en utilisant des tables de valeurs (commandes **TABLE** et **TBLSET**)
  a) Utilisez ces quatre méthodes pour résoudre l'équation $x^3 - 17x^2 + 7x + 13 = 0$.
  b) Décrivez ensuite, dans un document *Word*, les étapes que vous avez suivies dans votre cheminement, les solutions auxquelles vous êtes arrivés et les apports éventuels de chacune de ces méthodes à votre compréhension de la situation. Insérez des images de l'écran de la calculatrice pour soutenir votre argumentation.

2- *Comprendre comment la calculatrice trace un graphe*

Pour réaliser ce projet, vous devez commencer par télécharger le classeur *Excel* « *PixelsGraphiques.xls* » (disponible parmi les exemples de ce chapitre) et aller sur le site http://www.math.uqam.ca/_boileau/Segments.html pour lire le texte *Segments.pdf*.

Comme à la section 5.5, faites tracer le graphe de la fonction $y = \sqrt{7^2 - x^2}$:

- avec la calculatrice, dans la région donnée par « **ZStandard** »
- avec « *PixelsGraphiques.xls* »,
  1. en choisissant des nombres de pixels horizontaux et verticaux égaux à ceux de l'écran de la calculatrice, soit 95 horizontalement et 63 verticalement
  2. en traçant dans une région identique à celle choisie pour la calculatrice
  3. avec la grille visible.

En alternant, si nécessaire, entre les modes de traçés avec points reliés ou non :

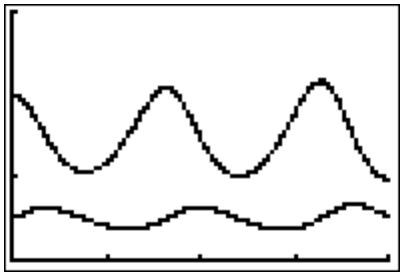
a) Identifiez par leurs lignes et colonnes, les « pixels ajoutés » par la calculatrice.
b) Vérifiez que les « pixels ajoutés », ne sont pas toujours placés de la même façon par la calculatrice et par « *PixelsGraphiques.xls* ».
c) En utilisant l'algorithme donné dans le texte *Segments.pdf*, calculez la position que devraient avoir les « pixels ajoutés ». Le résultat est-il en accord avec la calculatrice, avec « *PixelsGraphiques.xls* » ou aucun des deux?

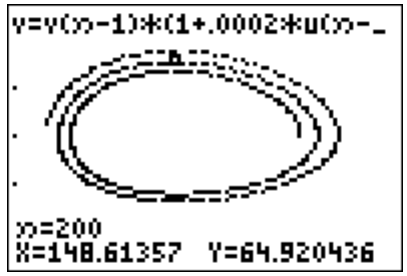
> **Remarque** : Pour réaliser les projets qui suivent, vous devrez faire appel au manuel de la calculatrice. Bien que ce manuel ait été traduit en français, il utilise, malheureusement, les termes anglais lorsqu'il réfère à des commandes. En général la traduction est évidente : par exemple **Solver** et **Solveur**. Signalons, cependant, un cas où elle ne l'est pas : le format **Time** est traduit par ***f(n)*** en français.
> Vous devez être conscient de ce problème et soit placer votre calculatrice en mode « langue anglaise » ou accepter de faire l'aller retour « français-anglais-français » pour trouver la traduction d'une commande qui n'est pas évidente.

3- *Les arbres d'une forêt*
Réalisez l'activité « *Les arbres d'une forêt* », décrite au chapitre 6 (« *Représentation graphique d'une suite* ») du manuel de votre calculatrice *TI-84 Plus*.

a) Répondez aux questions posées dans le manuel.
b) Si on arrondissait les nombres à l'entier le plus proche plutôt que de les tronquer (via iPart), obtiendrait-on les mêmes résultats?
(On arrondit en utilisant la fonction « arrondi », (round en anglais). Notez qu'il faut indiquer le nombre de décimales : *arrondi(nombre à arrondir, nombre de décimales désirées)*)
c) Qu'obtiendrait-on si on ne prenait pas la peine de transformer les nombres décimaux en entiers? Peut-on trouver un sens à cette façon de procéder?

4- *Le modèle prédateur- proie*
Réalisez l'activité « *Le modèle prédateur-proie* », décrite dans le chapitre 6 (« *Représentation graphique d'une suite* ») du manuel de votre calculatrice TI-84 Plus.
Écrivez un court texte qui explique ce que les différents graphiques (voir ci-dessous) nous montrent au sujet des populations de lapins et de loups.





Terminez en envisageant les possibilités de développement à long terme.

5- *Une suite définie par récurrence*
Considérons la suite u_n, définie comme suit :

$$u_0 = \frac{11}{2},\ \ u_1 = \frac{61}{11},\ \ u_n = 111 - \frac{1130}{u_{n-1}} + \frac{3000}{u_{n-1}\,u_{n-2}} \text{ (quand } n \geq 2\text{)}.$$

- Définissez cette suite dans votre calculatrice, et étudiez ce qui se passe quand la variable n augmente. Pour ce faire, faites appel à la fois aux tables de valeurs et aux graphes de suites (en mode **TRACE**).
- Refaites le même cheminement avec *Excel*. Trouvez-vous des différences?
- Voyez-vous matière à remettre en question l'exactitude des résultats obtenus?

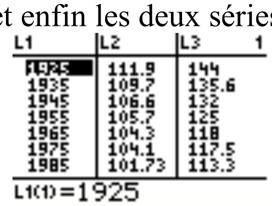
6- *Interprétation d'un modèle mathématique*
Pour réaliser ce projet, vous trouverez les informations utiles dans les chapitres 11 et 12, « Listes » et « Statistiques », du manuel de la calculatrice.

Selon certaines recherches, les femmes améliorent leurs performances en course à pied (ici au 800 m) plus rapidement que les hommes. À la lumière des données ci-contre, on est amené à se poser la question :
« Quand les femmes courront-elles plus rapidement que les hommes? »

Année	Record masculin (en s)	Record féminin (en s)
1925	111.9	144.0
1935	109.7	135.6
1945	106.6	132.0
1955	105.7	125.0
1965	104.3	118.0
1975	104.1	117.5
1985	101.73	113.3

Entrez tout d'abord les données dans les listes L_1, L_2 et L_3 (*Figure 1*), puis visualisez

- tout d'abord les temps des hommes en fonction de l'année (*Figure 2*), en les représentant par des marques en forme de petits carrés
- puis les temps des femmes en fonction de l'année (*Figure 3*), en les représentant par des marques en forme de petites croix
- et enfin les deux séries de points combinées (*Figure 4*).



L1	L2	L3
1925	111.9	144
1935	109.7	135.6
1945	106.6	132
1955	105.7	125
1965	104.3	118
1975	104.1	117.5
1985	101.73	113.3

L1(1)=1925

Figure 1

Figure 2

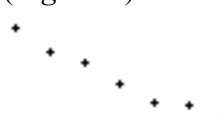


Figure 3

Figure 4

Faites ensuite calculer

- la droite de régression des données masculines, qui servira à définir Y_1
- la droite de régression des données féminines, qui servira à définir Y_2 et affichez l'écran obtenu avec la touche « Y= ».

Faites alors afficher les deux droites de régression, d'abord avec les données (*Figure 5*), puis sans (*Figure 6*). Si nécessaire, modifiez la fenêtre de visualisation de façon à voir le point d'intersection (*Figure 7*).

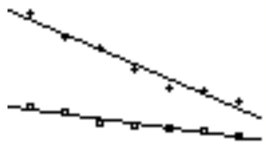


Figure 5

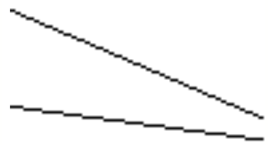


Figure 6

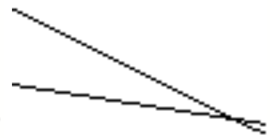


Figure 7

Faites ensuite calculer le point d'intersection de ces deux courbes en gardant une copie de l'écran utilisé pour commander la détermination du point d'intersection et de l'écran où la réponse de la calculatrice est affichée.

Utilisez le travail précédent pour répondre aux questions :

- Quand les femmes courront-elles aussi vite que les hommes?
- Lorsque cela se produira, combien de temps hommes et femmes prendront-ils pour courir le 800 m?
- Cette démarche est-elle fiable? Pourquoi?

7- *Nombre de solutions d'une équation*

On se propose d'étudier, en fonction des valeurs du paramètre A, le nombre de solutions, strictement positives, de l'équation :

$$Ax = \sin(x) \quad \text{où} \quad 0 < A.$$

a) Sachant qu'en un point $\left(x, \sin(x)\right)$ la tangente au graphe de la fonction sinus a pour pente $\cos(x)$, déterminez, avec l'aide de la calculatrice à affichage graphique, pour quelles valeurs de A il y a 4, 5 ou 6 solutions.

b) De façon générale, comment se comporte le nombre de solutions en fonction de A?

Chapitre 6

La géométrie dynamique avec Cabri

Tout comme la calculatrice TI-84Plus, le logiciel *Cabri Géomètre*[1] (que nous désignerons simplement par *Cabri*) 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 *Cabri*, en les situant parfois dans une perspective plus globale.

6.1 Vue d'ensemble : figure, construction et dessin

À l'ouverture de *Cabri*, nous constatons la présence d'une barre d'outils : un clic soutenu sur une icône fait apparaître un menu permettant de choisir un outil de sa catégorie (voir la figure 6.1.1). Nous vous recommandons d'activer immédiatement l'item « Aide » du menu du même nom, qui fait apparaître une fenêtre flottante donnant de l'information sur l'outil sélectionné.

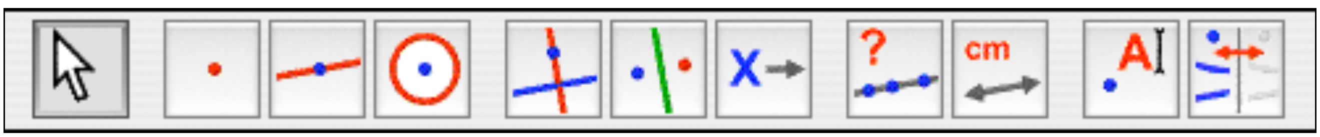


Figure 6.1.1 La barre d'outils de Cabri II Plus.
Les menus d'outils disponibles sont (de gauche à droite) « Manipulation », « Points », « Lignes », « Courbes », « Constructions », « Transformations », « Macros », « Propriétés », « Mesure », « Texte et symboles » et « Attributs ».

Pour commencer à nous familiariser avec *Cabri*, 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, *Cabri* 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 « Lignes » ➤ « Segment », nous désignons par un clic la première extrémité, repositionnons le curseur-souris, puis désignons par un second clic la deuxième extrémité. À chaque clic, un point est créé, et nous pouvons le nommer au vol en tapant le nom désiré au clavier : dans notre cas, nous nommerons *A* et *B* les deux extrémités. Notons au

---

[1] Dans ce chapitre, nous utiliserons la version la plus récente (*Cabri II Plus*), mais la plupart de nos remarques s'appliqueraient aussi aux versions antérieures de *Cabri*.

[2] Documentation et exemples sont disponibles en ligne au http://www.cabri.com/ ➤ « Télécharger » ➤ « Cabri II Plus ».

passage le fonctionnement de *Cabri* : un segment est créé suite à deux clics (désignant deux points) et non via un mouvement de « glisser » de la souris.

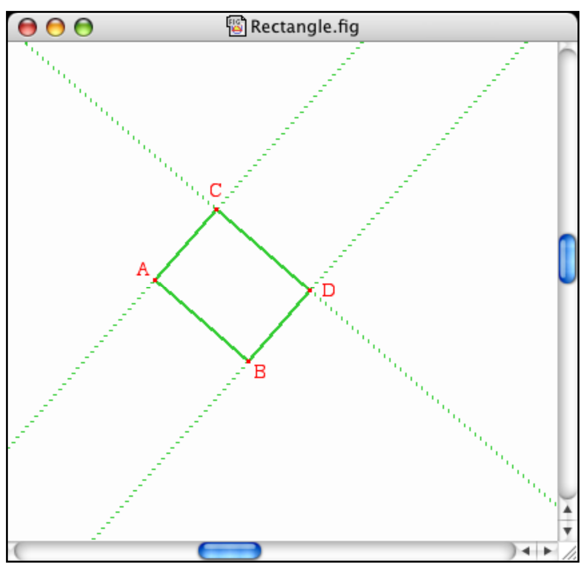
- Si désiré, l'outil « Manipulation » ➤ « Pointer » nous permet de sélectionner et de déplacer les divers éléments de notre figure (points, segment, noms). On revient tellement souvent à cet outil qu'il existe un raccourci clavier pour y accéder : la touche « escape ».

Pour corriger une erreur

On peut toujours annuler la dernière action en invoquant le menu « Edition » ➤ « Annuler ». Mais cette action est limitée à la seule dernière action effectuée.

On peut aussi détruire un objet quelconque en le sélectionnant (par un clic avec l'outil « Manipulation » ➤ « Pointer ») pour ensuite invoquer le menu « Edition » ➤ « 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 « Constructions » ➤ « Droite perpendiculaire ». Nous désignons d'abord par un clic l'extrémité *A* (notez que, lorsque nous sommes près du point *A*, le curseur *Cabri* change d'aspect et indique « Par ce point », confirmant ainsi que *Cabri* 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 *Cabri* change d'aspect et indique « Perpendiculaire à ce segment », confirmant ainsi que *Cabri* l'a bien identifié). La droite perpendiculaire désirée apparaît alors.
- Après avoir sélectionné l'outil « Points » ➤ « Point sur un objet », nous nous approchons de la droite perpendiculaire que nous venons de créer (le curseur *Cabri* nous confirme « Sur cette droite ») 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 « Points » ➤ « Point(s) d'intersection », nous désignons successivement les deux droites perpendiculaires (confirmation « Cette droite ») que nous venons de créer : nous nommons *D* le point d'intersection ainsi construit.
- 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 « Attributs » ➤ « Cacher/Montrer », puis nous désignons successivement par un clic les trois droites en



question. Tant que l'outil « Cacher/Montrer » reste sélectionné, les objets cachés apparaissent en pointillés (ou en « treillis » dans le cas des points), 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.

- Pour terminer notre rectangle, il nous suffit alors de créer les segments *AC*, *BD* et *CD*. Nous obtenons alors le résultat ci-contre (vu lorsque l'outil « Caché/Montrer » est choisi).

Une fois notre rectangle terminé, on peut le transformer de diverses manières avec l'outil « Manipulation » ➤ « Pointer ». 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 « Points » ➤ « Point »).

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

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

Dans le contexte d'utilisation de *Cabri*, 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 *Cabri*. Et le **dessin** est la représentation graphique sur l'écran produite par *Cabri*. 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*.

Notons que *Cabri* nous permet aussi de tracer le **dessin** d'un rectangle, en utilisant 4 segments horizontaux et verticaux (obtenus en utilisant la *touche majuscule*) 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 6.1.2).

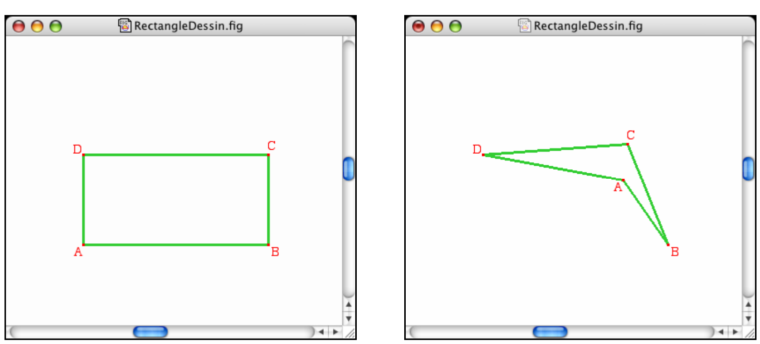


Figure 6.1.2 Le « dessin d'un rectangle » dans *Cabri* peut être aisément déformé.

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

6.2 Mathématiques expérimentales avec Cabri

Nous allons maintenant utiliser *Cabri* 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 6.2.3). Ce faisant, nous pourrons constater que les outils de *Cabri* ne se limitent pas à la seule géométrie *synthétique* : on peut aussi mesurer des longueurs et des angles, puis utiliser la *calculatrice* intégrée pour faire des calculs à partir des nombres ainsi obtenus.

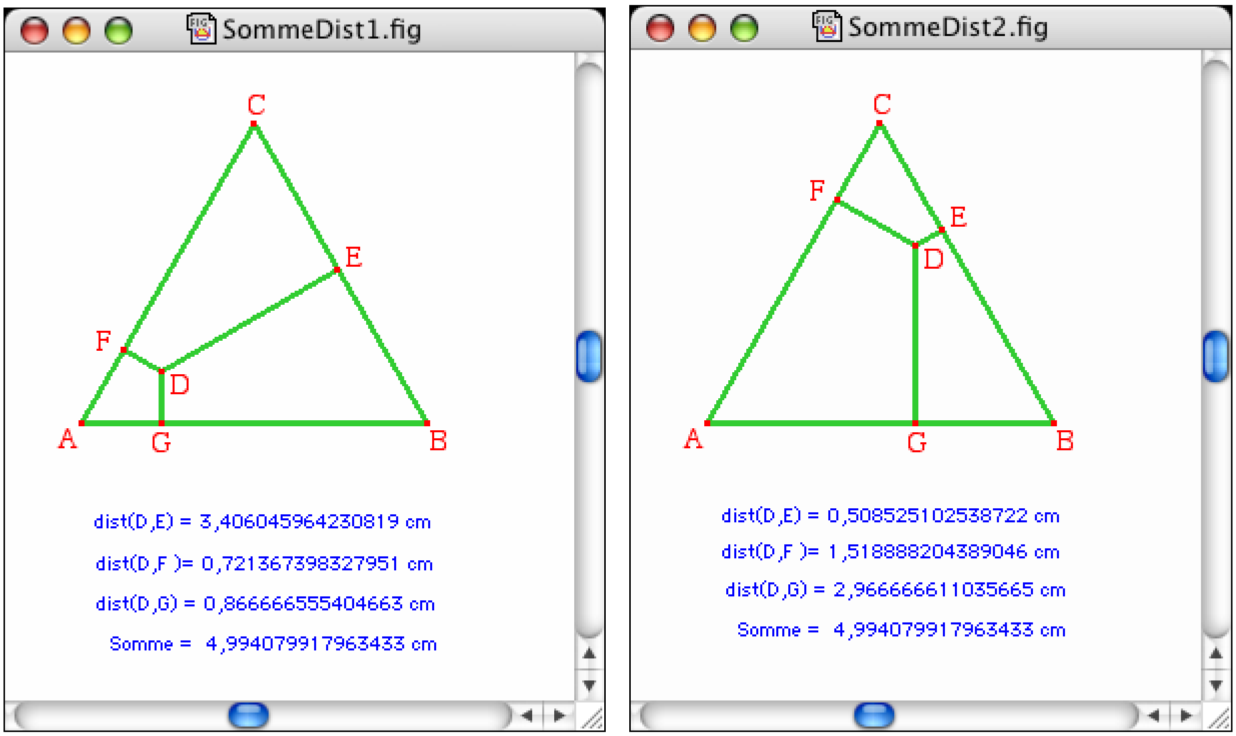


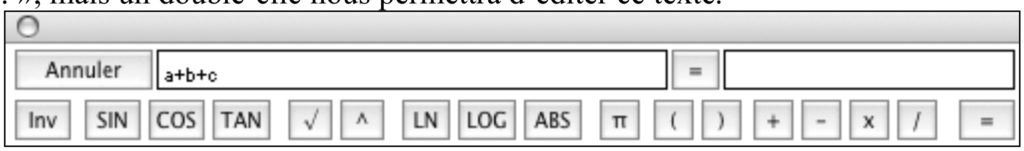
Figure 6.2.3 Deux positions du point *D* à l'intérieur du triangle équilatéral *ABC* : la somme des distances de *D* aux trois côtés du triangle semble constante.

Commençons par décrire une construction pour réaliser dans *Cabri* 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 « Courbes » ➤ « Cercle ») : 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[1]). 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*.

- À l'aide de l'outil « Mesure » ➤ « Distance ou longueur », on peut mesurer la longueur des segments *DE*, *DF* et *DG* en les désignant par un clic[2]. On obtient ainsi trois mesures, que l'on peut ensuite déplacer avec l'outil « Pointer ». Un double-clic sur un nombre permet d'éditer celui-ci : en particulier, un clic dans la zone d'édition immédiatement à gauche du nombre (ceci demande beaucoup de précision) permet d'ajouter un texte, « d(D,E) = » par exemple.
- Il ne nous reste plus qu'à faire la somme de ces trois longueurs. Pour ce faire, nous utiliserons l'outil « Mesure » ➤ « Calculatrice ». À l'aide d'une suite appropriée de clics sur les nombres[3] et sur les boutons de la calculatrice, nous formons une expression qui sera calculée lorsqu'on cliquera sur le bouton « = » : il suffira alors de cliquer une fois sur le résultat affiché par la calculatrice, puis une autre fois à l'endroit où nous voulons afficher ce résultat sur la feuille de dessin. Par défaut, ce nombre calculé sera précédé de la mention « Résultat : », mais un double-clic nous permettra d'éditer ce texte.



Modification des préférences de *Cabri*

Vous avez peut-être remarqué que tous les nombres affichés dans notre exemple comportent 16 chiffres décimaux et que les mesures sont en centimètres. En fait, nous pouvons dicter nos préférences en la matière à *Cabri* : il suffit de choisir le menu « Options » (ou « Cabri II Plus » dans la version Macintosh) ➤ « Préférences... » ➤ « Précision d'affichage et unités », puis de choisir une précision d'affichage *maximale* et « Centimètre (cm) » comme unité de longueur. 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.

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 « Lignes » ➤ « Polygone régulier »[4].

---

[1] Si notre triangle avait été obtenu à l'aide de l'outil « Polygone régulier », l'outil « Point(s) d'intersection » aurait créé les deux points d'intersection de la perpendiculaire et du triangle (puisqu'on n'aurait pu spécifier un côté sans le créer au préalable). Nous aurions eu par la suite à spécifier visuellement le « bon » point d'intersection.

[2] On aurait aussi pu mesurer la distance entre les deux points à leurs extrémités en désignant successivement ces deux points.

[3] Lors d'un clic sur un nombre, *Cabri* lui donne automatiquement un nom temporaire (en ce sens que ce même nombre peut recevoir un nom différent lors d'un calcul subséquent). Dans la version actuelle de *Cabri*, nous ne pouvons changer ce nom pour le rendre plus significatif.

[4] Soulignons que ces choix ont un impact sur la façon dont on peut manipuler, par la suite, le triangle ainsi obtenu. Dans notre cas, nous pourrons déplacer indépendamment les extrémités du côté *AB*; tandis que la construction en tant que polygone régulier nous aurait permis de modifier le centre du cercle circonscrit et un point du cercle correspondant à un sommet du triangle.

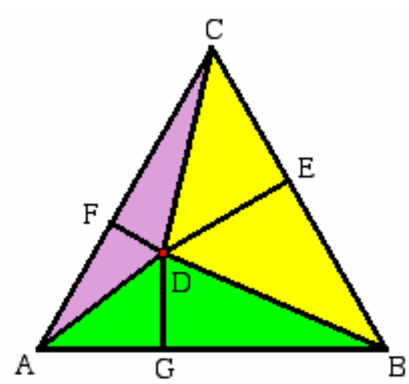
Maintenant que la construction est terminée, nous pouvons procéder à des expérimentations. En faisant varier la position du point *D*, nous pouvons faire les observations suivantes :

- 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 *Cabri*. 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



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

  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 ?)

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

« Cabri 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 *Cabri* 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

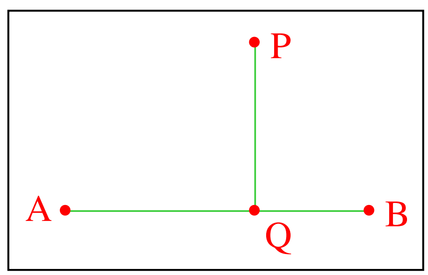
Les menus et la barre d'outils de *Cabri* mettent à notre disposition un grand nombre de possibilités. À l'aide du menu « Options » ➤ « Configuration des outils... », on peut aussi disposer autrement ou même faire disparaître certains des outils disponibles. (Pour plus de détails sur la façon de procéder, nous vous référons au manuel de référence de *Cabri*.) 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**.

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 *Cabri* 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 *Cabri* : 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 *Cabri* 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.
- En utilisant l'outil « Constructions » ➤ « 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 tout d'abord l'outil « Macros » ➤ « Objet(s) initial(aux) », puis on désigne par un clic tous les objets servant de base à notre construction : dans notre cas, il s'agit du point P et du segment AB. Ces objets initiaux devront être suffisants pour permettre la construction des objets finals que nous rencontrerons à la prochaine étape. (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).

(2) On choisit ensuite l'outil « Macros » ➤ « Objet(s) final(s) », puis on désigne tous les objets que l'on désire voir construits par la macro-construction : dans notre cas, il s'agit du seul segment *PQ*. Notons en passant que *Cabri* 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 qu'un clic sur un objet sélectionné permet de le désélectionner, et donc de corriger d'éventuelles erreurs : ceci est valable tant pour les objets initiaux de l'étape précédente que pour les objets finals de la présente étape.

(3) On choisit enfin l'outil « Macros » ➤ « Définir une Macro » (Mac) ou « Valider une macro... » (Win). *Cabri* vérifie alors si les objets initiaux sont suffisants pour construire les objets finals, et nous avertit en cas de problème. Par contre, si tout se passe bien, nous sommes invités (voir figure 6.3.4) à donner quelques informations pour compléter la définition de la macro : nom (qui apparaîtra dans la liste des outils), nom à donner au premier objet construit (visible en cas de survol par le pointeur de la souris), icône (visible dans la barre d'outils quand la macro est sélectionnée), et texte d'aide (visible dans la fenêtre d'aide quand la macro est sélectionnée). N'oublions pas de cocher la case « Enregistrer... » si nous voulons que notre macro soit sauvegardée dans un fichier, ce qui nous permettra de la réutiliser lors de sessions ultérieures avec *Cabri*.

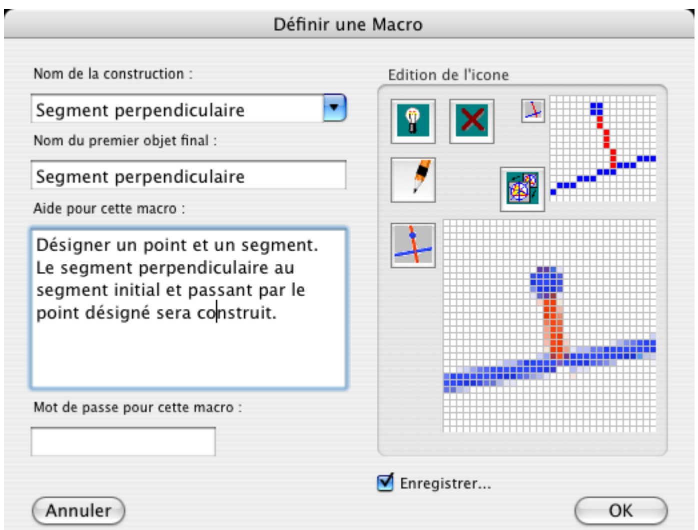


Figure 6.3.4 Fenêtre de dialogue pour compléter la définition de notre macro.

Au terme de ces trois étapes, un nouvel outil « Segment perpendiculaire » sera ajouté au menu d'outils « Macros » et pourra être utilisé tant que l'application *Cabri* restera ouverte. Cet outil pourra aussi être ramené (en utilisant le menu « Fichier » ➤ « Ouvrir menus, macros ... ») lors des sessions ultérieures de *Cabri*.

Nous allons maintenant rendre notre macro plus versatile en permettant aussi de désigner le segment par le biais de ses extrémités. Pour ce faire, on procède comme si on voulait définir une nouvelle macro en désignant **dans l'ordre** les points *P*, *A* et *B* comme objets initiaux, et le segment *PQ* comme objet final. Mais au moment de compléter la définition de la nouvelle macro, on lui donne exactement le même nom que la macro précédente, tout en modifiant le texte de l'aide pour couvrir les deux possibilités (voir la figure 6.3.5).

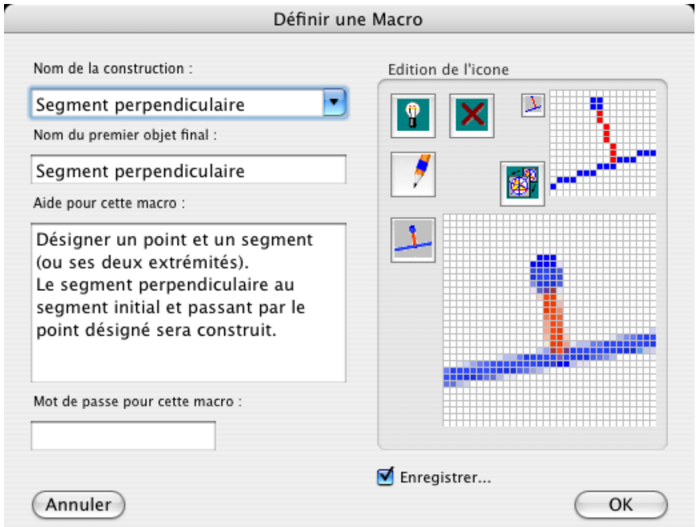


Figure 6.3.5 On « surcharge » une macro en lui donnant le même nom qu'une macro existante.

Cabri nous indique alors qu'une macro de même nom existe déjà et nous donne les choix indiqués à la figure 6.3.6. Si nous voulons pouvoir désigner à notre guise le segment par un clic sur celui-ci ou deux clics identifiant ses extrémités, il nous faudra choisir l'option « Ajouter » : *Cabri* fusionnera alors nos deux définitions de macros en une seule.

Figure 6.3.6 Dialogue affiché par *Cabri* quand on tente de créer une macro de même nom qu'une macro existante.

Précisons que, dans le cas où nous choisirions de désigner le segment par des clics sur ses extrémités, il nous faudrait **respecter l'ordre** utilisé lors de la définition de la seconde macro : un premier clic pour désigner le point extérieur au segment, et deux autres clics pour désigner les extrémités du segment.

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 $\sqrt{n+1}$.

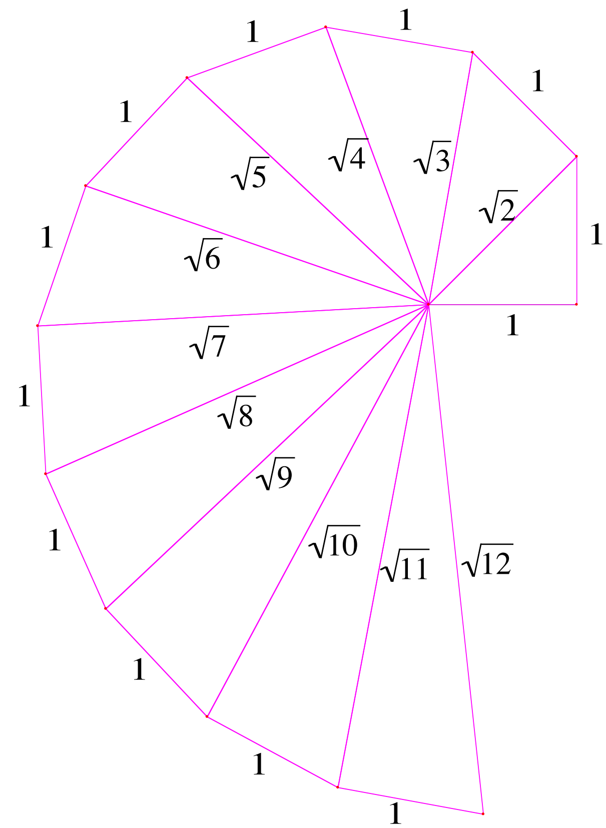


Figure 6.3.7 Portion de la spirale de Pythagore.
La construction a été faite dans Cabri, et les annotations ont été ajoutées dans *Word* en utilisant l'outil « Ajouter une étiquette » (de *LangageGraphique*) et l'éditeur d'équations.

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.

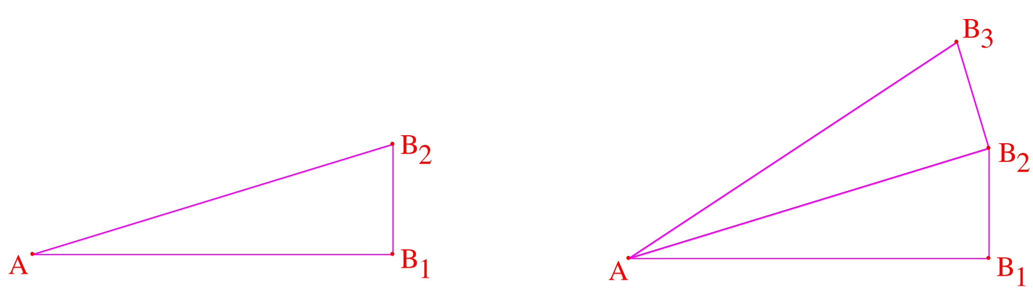


Figure 6.3.8 À gauche, le triangle initial. À droite, le nouveau triangle a été ajouté.

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 « Lignes » ➤ « Triangle »).

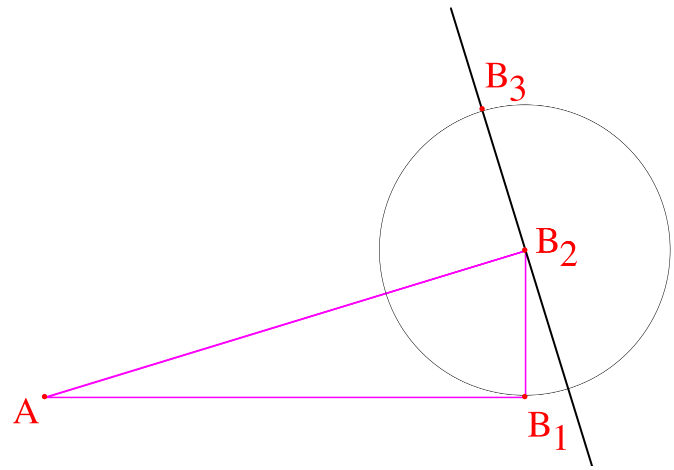


Figure 6.3.9 Construction du point B_3 à partir du triangle AB_1B_2.

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). **Pour que ces choix de sommets puissent rester implicites**, 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 *Cabri* 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 tout d'abord l'outil « Macros » ➤ « Objet(s) initial(aux) », puis on désigne le triangle AB_1B_2 comme seul objet initial. (Nous aurions pu désigner [dans l'ordre] les sommets A, B_1 et B_2 comme objets initiaux. Cependant, pour simplifier l'utilisation de la macro, nous avons choisi d'utiliser le triangle plutôt que ses sommets.)

(2) On choisit ensuite l'outil « Macros » ➤ « Objet(s) final(s) », puis on désigne le triangle AB_2B_3 comme seul objet final. Notons en passant que *Cabri* 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 finals si nous voulons qu'ils restent invisibles.

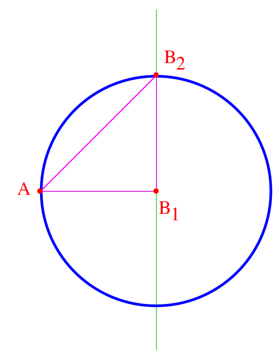
(3) On choisit enfin l'outil « Macros » ➤ « Définir une Macro » (Mac) ou « Valider une macro... » (Win). Nous donnons alors le nom « Étape de la spirale » à notre macro.

La levée des ambiguïtés dans *Cabri*

Normalement, dans *Cabri*, quand le pointeur de la souris survole un objet, il change de forme en affichant une indication comme « Ce point » ou « Ce triangle ». Mais quand il y a ambiguïté, c'est-à-dire quand plusieurs objets sont superposés à la position du pointeur, *Cabri* affiche plutôt « Quel objet ? ». En appuyant sur le bouton gauche de la souris, on peut alors faire apparaître (dans un menu local) une liste des objets présents, affichés en ordre chronologique inverse et dans la couleur correspondante : on peut alors sélectionner l'objet désiré.

Par exemple, si on survole le segment AB_2 de la portion droite de la figure 6.3.8 et que l'on appuie sur le bouton gauche, nous aurons la possibilité de sélectionner l'un ou l'autre des deux triangles. Cependant, quand viendra le moment d'appliquer la macro « Étape de la spirale » au triangle AB_2B_3, il sera plus aisé d'éviter d'avoir à lever une ambiguïté en désignant directement ce triangle par un clic sur un côté non commun (AB_3 ou B_2B_3).

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 « Macros » ➤ « Étape de la spirale » et l'on désigne par un clic ce premier triangle : un nouveau triangle est ajouté.
- On continue de la sorte autant de fois que désiré : un clic sur le dernier triangle construit produira un nouveau triangle de la spirale.
- On peut alors choisir l'outil « Manipulation » ➤ « Pointer » 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, comme nous le montre 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. Mais pourquoi en est-il ainsi ?

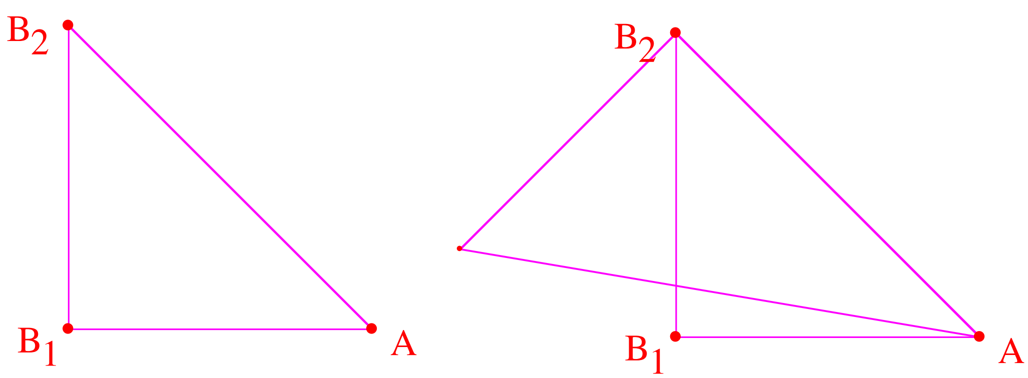


Figure 6.3.10 Quand on tente d'appliquer la macro « Étape de la spirale » à la figure de gauche, on obtient le résultat à droite.

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.

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 :

- Soit S l'image du point B_1 par la symétrie centrale de centre B_2. On a que S et B_1 sont dans des demi-plans distincts.
- Soit P la projection (perpendiculaire) de S sur la perpendiculaire à AB_2 élevée à partir de B_2. Comme SP est parallèle à AB_2, S et P sont dans le même demi-plan.
- La demi-droite issue de B_2 et passant par P coupera donc le cercle en un seul point B_3, qui est dans le demi-plan voulu.

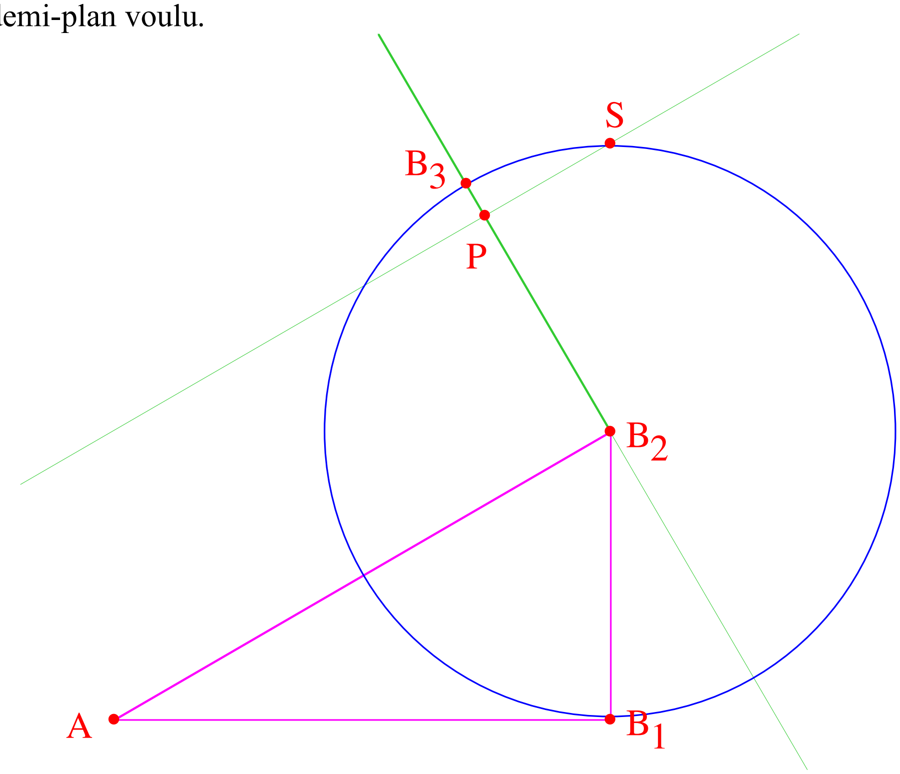


Figure 6.3.11 Construction du point d'intersection approprié.

Si nous utilisons cette construction pour définir une nouvelle macro (que nous nommerons « Étape (générale) »), 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 (en désignant ses trois sommets) lors de sa définition[1] : la figure 6.3.12 en donne un exemple.

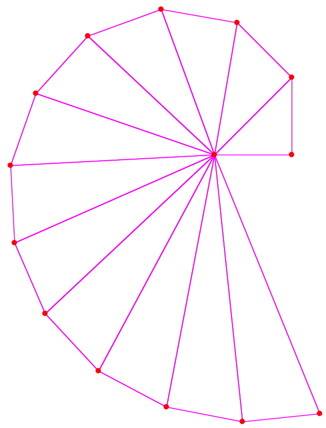


Figure 6.3.12 Deux portions symétriques de spirales de Pythagore.

6.4 Initiation aux lieux géométriques

Cabri nous permet aussi de représenter des lieux géométriques. Illustrons ceci en considérant la **parabole** définie non pas comme une section de cône mais plutôt comme le lieu 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).

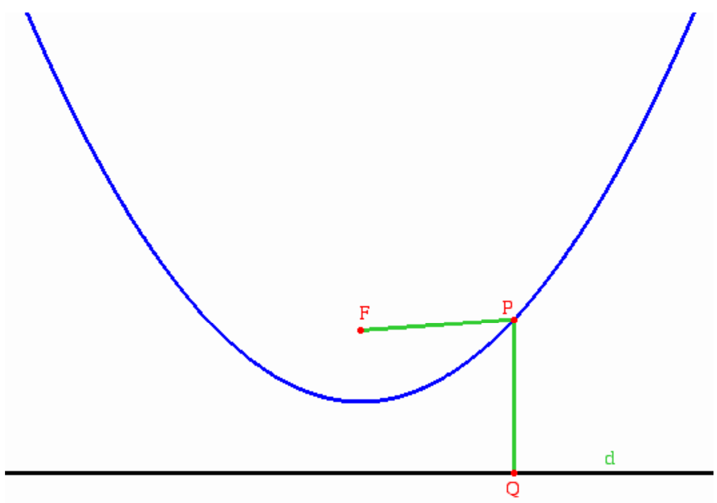


Figure 6.4.13 Définition d'une parabole en tant que lieu des points *P* à égale distance du foyer *F* et de la directrice *d*.

Bien que *Cabri* dispose d'un outil pour tracer une conique quelconque passant par 5 points (voir l'outil « Courbes » ➤ « Conique »), aucun outil disponible ne nous permet de créer directement une parabole étant donné un foyer et une directrice.

[1] 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.

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 *Cabri* dispose de deux outils pour rendre tout ceci plus simple.

Le premier outil de simplification est « Texte et symboles » ➤ « Trace ». Quand on clique sur un ou plusieurs objets avec cet outil, *Cabri* retient qu'il faudra laisser une trace de ces objets quand on effectuera des déplacements avec l'outil « Manipulation » ➤ « Pointer ». 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 une 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 sélectionnés par l'outil « Trace » continueront de laisser une trace tant et aussi longtemps qu'ils ne seront pas désélectionnés (via un autre clic avec l'outil « Trace »). Pour effacer toutes les traces obtenues, il suffira alors d'utiliser le menu « Édition » ➤ « Tout redessiner ».

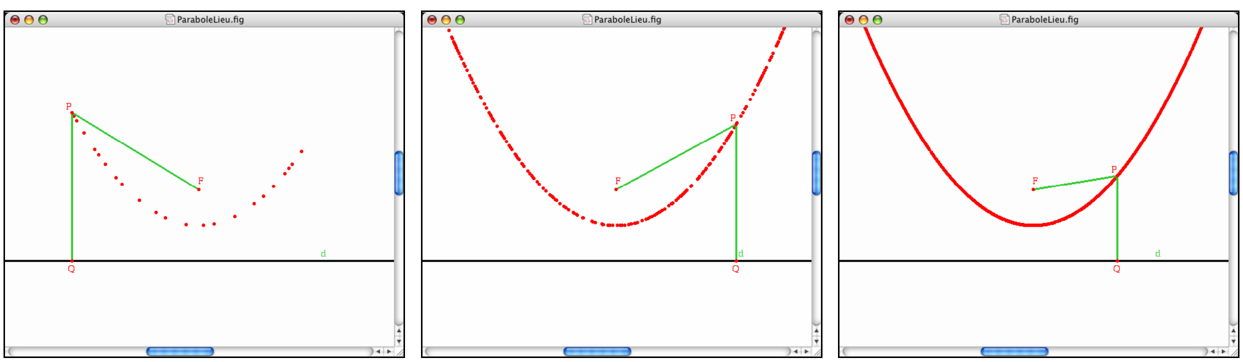


Figure 6.4.14 La trace du point *P* de la parabole quand on déplace le point *Q* sur la directrice. En repassant tout en ralentissant, on peut multiplier le nombre de points tracés.

Le second outil de simplification est « Constructions » ➤ « Lieu ». On devra tout d'abord désigner un objet dont *Cabri* laissera une sorte de trace automatique (un *lieu*), puis un point sur un objet que *Cabri* déplacera (automatiquement, là aussi) pour obtenir le lieu en question. À l'aide du menu « Options » (ou « Cabri II Plus » dans la version Macintosh) ➤ « Préférences... » ➤ « Lieux », on peut aider *Cabri* à tracer un lieu plus fidèle en spécifiant le nombre de points à utiliser pour sa « trace automatique », et aussi s'il faut relier les points obtenus par des segments.

Le grand avantage de l'outil « Lieu » sur l'outil « 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), on peut construire des points sur des lieux ou des 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 *Cabri*). Dans les exercices, nous verrons comment construire les autres coniques (ellipses et hyperboles) en tant que lieux géométriques.

6.5 Géométrie analytique avec Cabri

Jusqu'à présent, nous avons utilisé *Cabri* dans un contexte de **géométrie synthétique** : les seuls nombres qui interviennent apparaissent dans des contextes de mesure (de longueurs, d'angles, etc.). Mais *Cabri* nous permet aussi de travailler dans un contexte de **géométrie analytique**, utilisant coordonnées et équations. En fait, *Cabri* fait implicitement référence à un système d'axes permettant d'afficher les coordonnées et les équations des divers objets (à l'aide de l'outil « Mesure » ➤ « Coord. ou équation ») : on peut d'ailleurs les faire apparaître et disparaître à l'aide de l'outil « Attributs » ➤ « Montrer les axes » / « Cacher les axes »)[1]. On peut aussi changer l'échelle des deux axes en même temps (en saisissant et en déplaçant l'une des graduations de l'axe des abscisses) ou du seul axe des ordonnées (en saisissant et en déplaçant l'une des graduations de l'axe des ordonnées). On peut incliner le système d'axes (en « saisissant » l'axe des x), ou même changer l'angle entre les deux axes (en « saisissant » l'axe des y),. Aussi bizarre que cela puisse paraître, ceci n'affecte en rien les diverses mesures (longueurs, aires, angles) : seules les coordonnées et les équations seront modifiées.

Voyons comment utiliser *Cabri* pour construire la fonction sinus en enroulant 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 *Cabri*. (Le résultat final apparaît à la figure 6.5.15.)

- En faisant appel à l'outil « Attributs » ➤ « Montrer les axes », on fait apparaître ceux-ci.
- On veut ensuite construire le cercle unité centré à l'origine. On peut désigner l'origine des axes comme centre du cercle, mais on ne peut désigner une graduation comme point du cercle. En choisissant l'outil « Attributs » ➤ « Grille » puis en désignant par un clic le système d'axes, on fait apparaître une grille qui nous sera utile.
- On trace alors un cercle dont le centre est l'origine et qui passe par le point (1,0) de la grille. (Pour lever l'ambiguïté, on doit choisir « Point sur... » : « Cette grille » et non « Cet axe ».) Si on le désire, on peut maintenant masquer la grille (outil « Attributs » ➤ « Cacher/Montrer »).
- On place un point x sur l'axe des abscisses (outil « Points » ➤ « Point sur un objet »). Attention, ne pas le placer sur la grille mais bien sur l'axe : ainsi, on pourra le déplacer continûment, et non par sauts. On affiche ensuite les coordonnées du point x (outil « Mesure » ➤ « Coord. ou équation »).
- On veut maintenant enrouler l'axe des abscisses autour du cercle, et trouver ainsi le point P où aboutira le point x. Pour ce faire, on choisit l'outil « Constructions » ➤ « Report de mesure », puis on désigne successivement l'abscisse du point x (un nombre), le cercle unité et le point (1,0). Un point (que nous nommerons P) apparaît alors.

[1] On ne peut détruire ce système d'axes original. On peut cependant créer d'autres systèmes d'axes, que l'on peut cacher ou détruire à notre guise.

Une remarque en passant : bien que l'outil se nomme « Report de mesure », il tient compte du 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 Q comme l'intersection de la perpendiculaire à l'axe des abscisses passant par x et de la perpendiculaire à l'axe des ordonnées passant par P. Le lieu du point Q quand le point x 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.

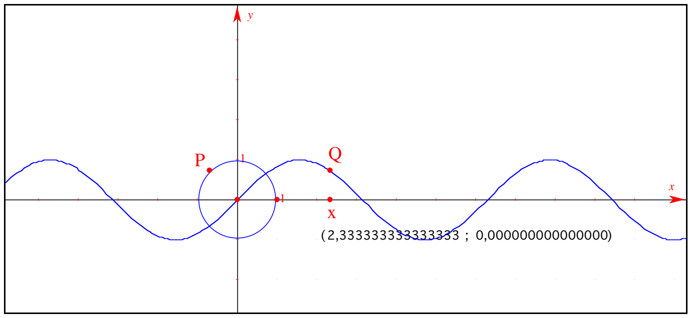


Figure 6.5.15 Le graphe de la fonction sinus, obtenu via une fonction d'enroulement.

Comme nous l'avions annoncé, nous avons donc obtenu le graphe d'une fonction (le sinus) au terme d'une construction culminant en un lieu géométrique. Mais *Cabri* nous permet aussi d'obtenir directement le graphe cartésien associé à une expression. Par exemple, si nous voulons tracer la fonction $y = \tan(x)$, on procède comme suit :

- À l'aide de l'outil « Texte et symboles » ➤ « Nombre », on détermine avec la souris (par un clic ou un glisser) un rectangle de texte où l'on écrira un nombre (par exemple 2).
- À l'aide de l'outil « Texte et symboles » ➤ « Expression », on détermine avec la souris (par un clic ou un glisser) un rectangle de texte où l'on écrira « tan(a*x) ».
- Si nécessaire, on fait apparaître les axes (via « Attributs » ➤ « Montrer les axes »).
- Puis, en utilisant l'outil « Mesure » ➤ « Appliquer une expression », on clique d'abord sur l'expression (ici « tan(a*x) ») puis sur le nombre, et enfin sur le système d'axes qui sera utilisé pour la représenter : le graphe cartésien voulu sera tracé (sous la forme d'un lieu géométrique). On pourra ensuite modifier le paramètre a au moyen d'un double-clic sur le nombre.

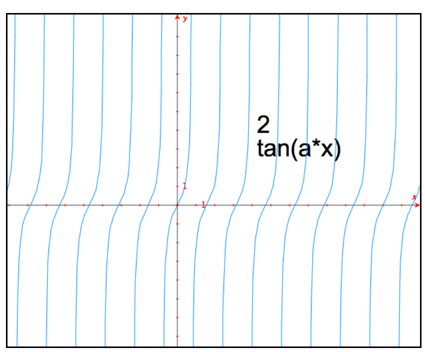


Figure 6.5.16 Tracé direct d'un graphe cartésien
à l'aide des outils « Expression » et « Appliquer une expression »

Cette façon d'obtenir directement le graphe cartésien d'une fonction décrite par une expression algébrique n'est pas la seule applicable. Par exemple, pour représenter une famille de fonctions $af\left(b(x-h)\right)+k$, nous pouvons entrer dans la calculatrice les diverses coordonnées des points-paramètres afin de calculer la valeur désirée, que nous devrons ensuite reporter sur l'axe des y avant de pouvoir compléter la construction (voir, par exemple, la figure 6.5.17). Vous aurez l'occasion de mettre ceci en pratique dans les exercices.

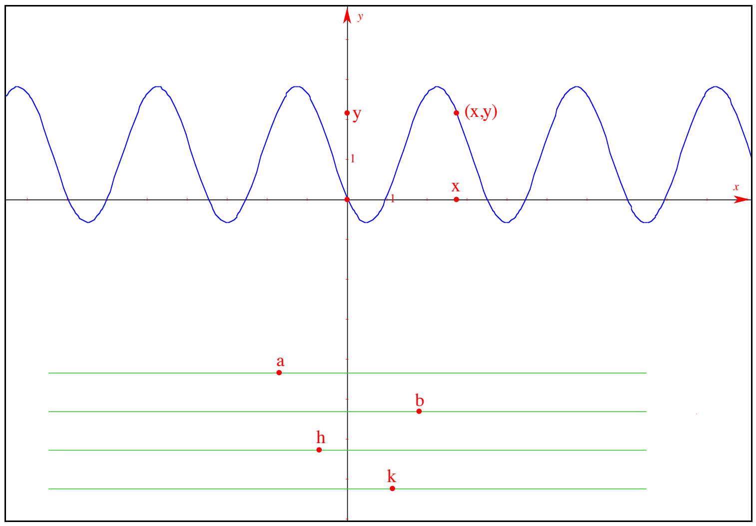


Figure 6.5.17 Représentation de la famille de courbes $a\sin\left(b(x-h)\right)+k$ dans *Cabri*. La valeur des paramètres a, b, h et k est déterminée en déplaçant les points correspondants sur des segments.

6.6 Aller plus loin avec Cabri

Il arrive parfois que *Cabri* produise un résultat qui nous surprend, et il nous faut alors comprendre un peu de son fonctionnement interne pour expliquer le phénomène observé. C'est ce qui nous occupera dans cette section.

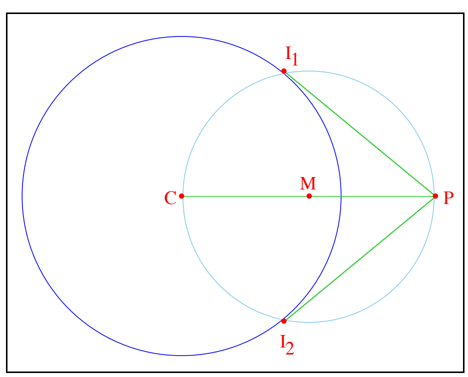


Figure 6.6.18 Tangentes par un point *P* à un cercle de centre *C*. Les points de tangence I_1 et I_2 seront les intersections du cercle initial avec le cercle de diamètre CP.

Supposons que nous voulions réaliser la construction des tangentes communes à deux cercles. Nous commencerons tout d'abord par un problème plus facile : la construction des tangentes à un cercle par un point hors de ce cercle (voir figure 6.6.18 et l'exercice 3).

Une façon de résoudre notre problème initial est soustraire le rayon r du petit cercle au rayon R du grand cercle pour se ramener au problème de tracer les tangentes à un cercle passant par un point donné : les tangentes communes recherchées s'obtiendront ensuite aisément comme droites parallèles (voir figure 6.6.19).

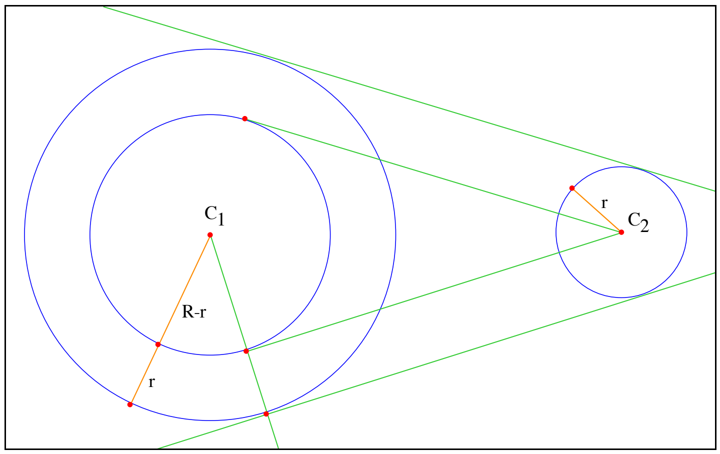


Figure 6.6.19 Construction des tangentes communes à deux cercles.

Cette construction pose cependant problème, précisément en raison du dynamisme permis par *Cabri* : si, en modifiant le rayon du petit cercle, celui-ci en vient à devenir plus grand que l'autre, alors la construction précédente cesse d'être valable et les deux tangentes disparaissent (puisque la construction utilisée cesse de les définir dans ce nouveau cas de figure).

En pratique, pour résoudre notre problème, on pourra construire les tangentes communes aux deux cercles deux fois[1] : dans le cas où le premier cercle est plus grand que le second, puis dans le cas où c'est le second qui est plus grand que le premier. Cependant, il ne faudra pas oublier que, contrairement à ce qui peut sembler se passer à l'écran de l'ordinateur, il existe en fait 4 droites tangentes, dont 2 au maximum peuvent apparaître à un moment donné. Si nous désirons poursuivre la construction (par exemple, en plaçant un point sur l'une des tangentes), il nous faudra le faire deux fois, une fois pour chaque cas de figure[2]. Sans compter que si notre construction prolongée nous conduisait à devoir considérer un nouveau cas de figure, il faudrait alors gérer 4 cas au total : imaginez si les cas de figures venaient à se multiplier !

[1] En fait, au lieu de répéter deux fois la construction, on peut la réaliser dans un cas de figure, l'enregistrer sous forme de **macro**, puis l'appliquer dans l'autre cas de figure.

[2] Poursuivant la note précédente, disons que le meilleur choix serait alors d'enregistrer la **macro** quand la totalité de la construction serait complétée.

Il existe une solution plus satisfaisante, mais cependant plus complexe, à notre problème. Elle est basée sur une étude fine des algorithmes internes de *Cabri*, faite par Éric Hakenholz[1] et qui a donné naissance à ce qu'on appelle la « géométrie booléenne ». Le point de départ est la découverte du phénomène d'orientation des droites et de leurs points d'intersection avec les cercles, tel que décrit à la figure 6.6.20. À partir de cette constatation, on peut développer des « macros booléennes » qui nous permettront de construire, à partir de nos deux cercles initiaux, deux autres cercles (correspondant au plus grand et au plus petit des deux cercles initiaux), sur lesquels nous pourrons réaliser la construction des tangentes communes : ceci fera l'objet du projet 5.

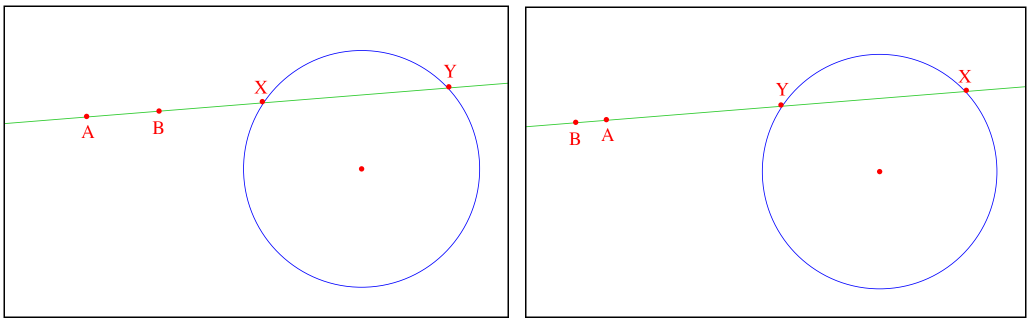


Figure 6.6.20 À gauche, on a fait passer une droite par deux points A et B, puis on a nommé X et Y les points d'intersection de la doite avec un cercle. Quand on change l'ordre des points A et B, on constate que l'ordre des points X et Y change aussi. La « géométrie booléenne » est l'art d'exploiter cette orientation implicite dans *Cabri*.

Nous devons cependant constater que le type d'analyse précédent est basé sur des considérations algorithmiques subtiles, qui apportent une dose de complexité qu'on aimerait pouvoir éviter. En fait, ce qui fait défaut ici, c'est de pouvoir « dire » au logiciel que l'on veut travailler non pas sur tel cercle, désigné par un geste, mais sur tel cercle, désigné par une propriété (comme celle d'avoir un rayon supérieur à tel autre cercle). Nous touchons ici la délicate recherche d'un équilibre entre deux manières de communiquer avec nos outils technologiques : **textuellement** (en tapant des commandes au clavier) et **gestuellement** (par des mouvements de souris et des clics de boutons). *Cabri* a largement choisi de privilégier une communication gestuelle, mais on constate ici les problèmes apportés par l'exclusivité de ce choix, notamment dans la gestion des cas de figure.

6.7 Coup d'oeil sur d'autres logiciels de géométrie dynamique

Tel que nous l'avons déjà mentionné au chapitre sur les calculatrices, diverses versions de *Cabri* existent aussi sur certaines calculatrices *Texas Instruments* (TI-83 Plus, TI-84 Plus, TI-89, TI-92, Voyage 200). Par comparaison avec leurs contreparties sur ordinateur, ces versions souffrent de la résolution limitée, du manque de couleur et de l'absence de véritables claviers et de souris des calculatrices (voir la figure 6.7.21). Mais c'est parfois la seule façon d'utiliser *Cabri* avec nos élèves...

---

[1] Le lecteur désireux d'en savoir plus pourra consulter la page Web suivante :
http://icosaweb.ac-reunion.fr/GeomJava/abraCAda/Docs/LesAbras/alice.htm

Et, si le professeur a accès à *Cabri* sur ordinateur, il peut toujours préparer les fichiers plus complexes dans cet environnement, pour ensuite les enregistrer sous une forme lisible par la calculatrice (via le menu « Fichier » ➤ « Exporter pour la calculatrice... ») et utiliser *TI-Connect* pour effectuer le transfert. Dans *Cabri*, un clic-droit permet de choisir (dans un menu contextuel) de faire apparaître un rectangle correspondant à l'écran de la calculatrice visée, ce qui nous permet de disposer la figure de façon à ce qu'elle soit utilement représentée sur la calculatrice, qui ne dispose d'aucun mécanisme pour faire défiler l'affichage.

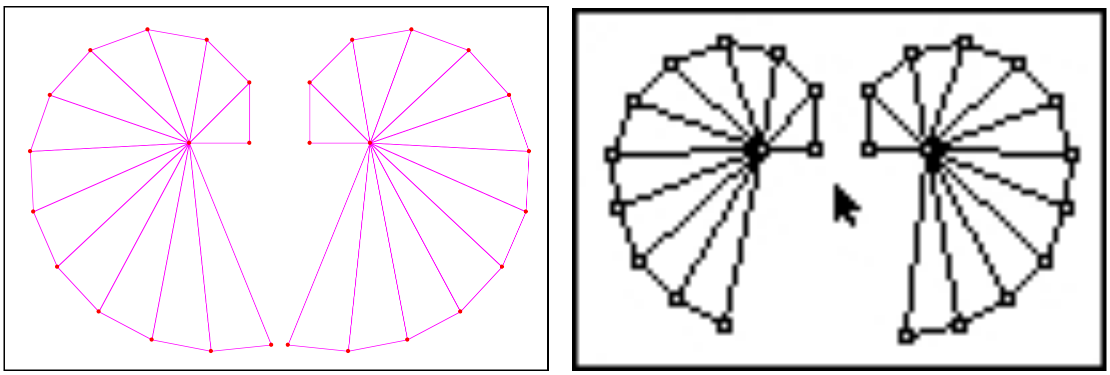


Figure 6.7.21 À gauche, la figure obtenue à la fin de la section 6.3.
À droite, la figure correspondante transférée sur une calculatrice *TI-84 Plus*.
Notez que la figure comportait trop d'objets pour *Cabri Jr*, et a été tronquée.

Signalons aussi *Cabri 3D*, une version tridimensionnelle de *Cabri*. Même si la version actuelle (2.0) ne comporte pas encore de macros, de lieux géométriques, ni de mécanisme de gestion des ambiguïtés, et même si les objets graphiques produits sont peu paramétrables (dimensions des représentations des « plans », « cônes » et autres objets, degré de transparence, etc.), ce produit s'est déjà considérablement amélioré depuis sa première apparition, et semble promis à un bel avenir. La figure 6.7.22 permet de constater la qualité graphique des représentations produites par *Cabri 3D*, mais ne donne cependant pas une bonne idée du dynamisme permis à la fois par la manipulation directe des objets libres de la figure (comme dans *Cabri 2D*), et par un changement interactif du point de vue (essentiel en 3D car l'observateur fait alors partie de l'espace occupé par les objets avec lesquels il expérimente).

On peut trouver une mine de renseignements à propos de la famille *Cabri* sur le Web. Mentionnons tout d'abord le site universitaire de *Cabri*

http://www.cabri.net/

qui permet au visiteur de s'abonner à un forum de discussion pour les utilisateurs de *Cabri* et contient plusieurs liens vers d'autres sites intéressants. De son côté, le site commercial de *Cabri*

http://www.cabri.com/

regorge de ressources intéressantes : versions de démonstration de *Cabri II Plus*, de *Cabri 3D* et des plug-in correspondants, versions électroniques des manuels en plusieurs langues, tutoriels en ligne, liste de publications, etc.

Mais *Cabri* n'est pas le seul logiciel de géométrie dynamique. Dans un article (en anglais) de l'encyclopédie en ligne Wikipedia

http://en.wikipedia.org/wiki/Interactive_geometry_software on dresse une liste de plusieurs de ces logiciels, avec des références aux sites où l’on peut se les procurer. Certains de ceux-ci sont libres, donc gratuits, ce qui peut s’avérer utile dans des écoles où l’achat de logiciels est difficile.

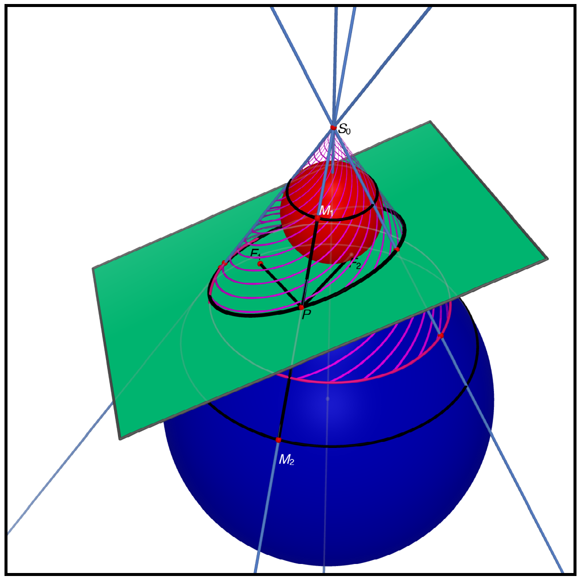


Figure 6.7.22 Représentation des sphères de Dandelin dans *Cabri 3D*. Chacune de ces deux sphères est tangente à la fois au cône (en un cercle) et au plan de coupe (en un point-foyer).

Parmi ces logiciels de géométrie dynamique, mentionnons tout d’abord le *Geometer’s Sketchpad*, ne serait-ce que parce que son usage est conseillé par le *National Council of Teachers of Mathematics* et que sa position est dominante aux États-Unis. Il partage avec *Cabri* certaines caractéristiques fondamentales (objets et constructions de base, lieux géométriques, macros, etc.) tandis que certains outils sont manquants (polygones réguliers, coniques par 5 points, intersection de lieux) ou ajoutés (en particulier des possibilités d’itérations - voir figure 6.7.23). Pour plus de renseignements, vous pouvez consulter le site

http://www.dynamicgeometry.com/.

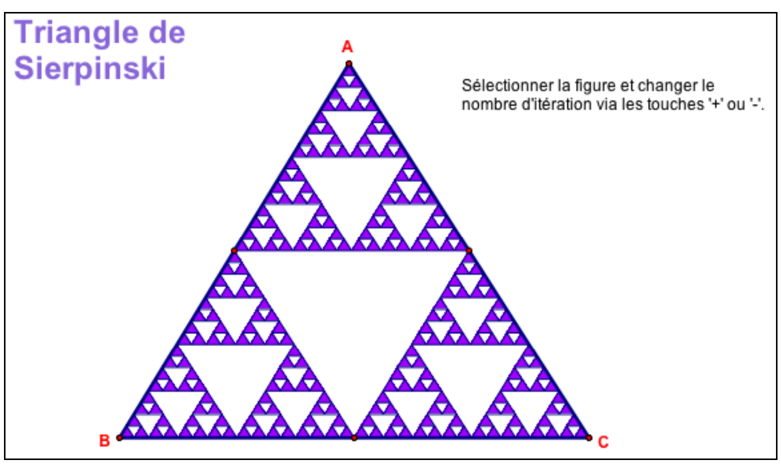


Figure 6.7.23 Dans *Geometric Sketchpad*, on dispose d’un outil d’itération qui permet de réaliser et de manipuler des figures tel ce triangle de Sierpinski.

Un autre logiciel de géométrie dynamique digne de mention est *GeoGebra*. Son fonctionnement général est très semblable à celui de *Cabri* et, bien que ses possibilités soient moins grandes dans certains domaines (notamment pour les lieux géométriques), il a ses avantages propres. En particulier, il offre la possibilité d'une interaction textuelle (voir figure 6.7.24), avec des commandes spécifiques pour les constructions conditionnelles et itérées. De plus, il s'agit d'un logiciel libre, et donc gratuit : il est donc disponible à quiconque (professeur, élève, parent) dispose d'un ordinateur. Vous trouverez plus de renseignements à l'adresse Web suivante :
http://www.geogebra.org/.

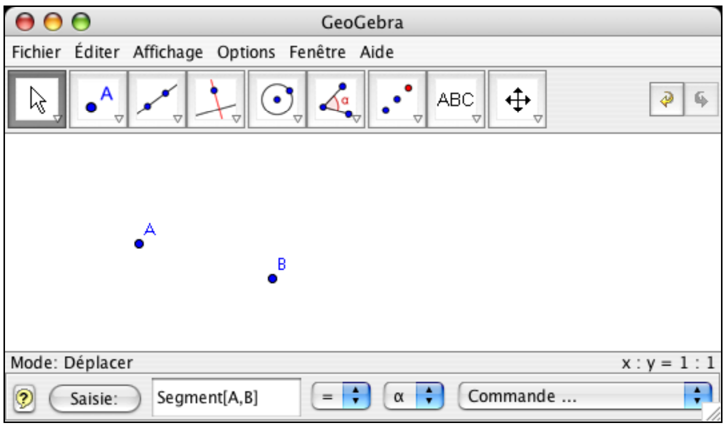


Figure 6.7.24 *GeoGebra* permet de construire des objets gestuellement et textuellement. Ici, pour construire un segment reliant les points *A* et *B*, on peut faire un glisser de *A* vers *B* ou saisir la commande « Segment [A,B] ».

En terminant, mentionnons *Cinderella*, un logiciel de géométrie passablement original, en raison des choix qui ont présidé à son élaboration. Les macros, par exemple, ont été implantées très différemment : on procède par un copier-coller suivi d'une redéfinition des objets ainsi créés. Comme l'illustre la figure 6.7.25, ce logiciel semble destiné à des utilisateurs avancés. Il permet notamment une approche à la fois gestuelle et textuelle, dont un langage de programmation complet (bien que non complètement intégré pour le moment[1]).

Mais *Cinderella* diffère de *Cabri* à un niveau plus fondamental : à l'opposé de *Cabri*, *Cinderella* est **continu** et **non-déterministe**[2]. Expliquons ce que ceci veut dire, en commençant par la continuité. Nous avons vu à la section 6.6 (voir, en particulier, la figure 6.6.20) qu'un déplacement progressif d'un point avec la souris pouvait provoquer des « sauts » dans la position des objets qui en dépendent : en raison de la présence de ce type de phénomène, on dira que *Cabri* est un logiciel de géométrie dynamique qui **n'est pas continu**. Dans *Cinderella*, au contraire, il y a absence de tels phénomènes de « sauts » : *Cinderella* est donc **continu**. À priori,

[1] La version courante (2.0.6) ne permet pas de créer par programmation des objets qui pourront être manipulés directement via la souris.
[2] Pour en savoir plus en la matière, le lecteur pourra consulter la thèse de Bernard Genevès, *Vers des spécifications formelles : Fondements Mathématiques et Informatiques pour la Géométrie Dynamique*, soutenue à l'université Joseph Fourier de Grenoble, et disponible au
http://www-iam.imag.fr/ThesesIAM/these_BG.pdf.

on pourrait souhaiter que nos logiciels de géométrie dynamique soient continus, de façon à éviter de brusques changements suite à de petits déplacements d'un élément de la figure. Rappelons cependant que certains ont su tirer profit de la discontinuité présente dans *Cabri*, en créant la **géométrie booléenne**.

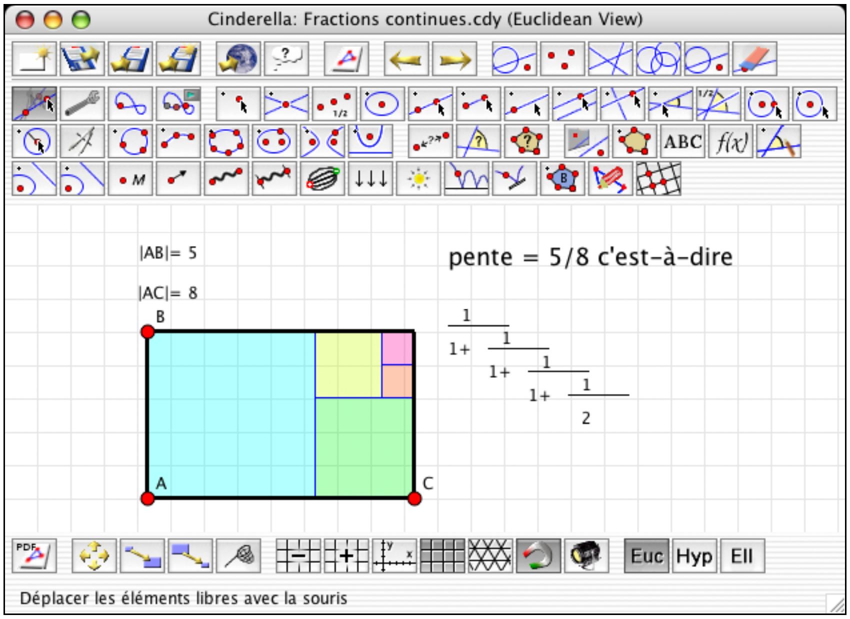


Figure 6.7.25 *Cinderella* dispose de caractéristiques avancées : langage de programmation, trois types de géométries (euclidienne, hyperbolique et elliptique), et laboratoire de simulations physiques.

Parlons maintenant du caractère déterministe ou non d'un logiciel de géométrie dynamique. On dira qu'un tel logiciel est **déterministe** quand la configuration de la figure est entièrement déterminée par la position des éléments libres et semi-libres (les points placés sur des objets). En ce sens, *Cabri* est un logiciel déterministe, mais *Cinderella* n'en est pas un, comme l'illustre la figure 6.7.26 : dans un logiciel non-déterministe, la figure obtenue dépend non seulement de la position des éléments libres et semi-libres, mais aussi des déplacements les ayant conduit à occuper leurs positions.

Le manque de déterminisme peut sembler un défaut encore plus gênant que le manque de continuité, et l'on peut se demander s'il est sage de troubler nos élèves avec de tels comportements. En fait, on souhaiterait utiliser un logiciel de géométrie dynamique à la fois déterministe et continu. Mais on peut démontrer qu'un tel logiciel ne peut exister[1]. Il faut donc nécessairement faire un choix, et *Cabri* et *Cinderella* illustrent les deux options possibles...

---

[1] Pour plus de détails, on pourra consulter la thèse de Bernard Genevès, mentionnée plus haut, ou l'article *Towards a Theory of Visualization by Dynamic Geometry Software Paradigms, Phenomena, Principles* de Thomas Gawlick, disponible au http://www.icme-organisers.dk/tsg16/papers/Gawlick.theoryofvisDGS.pdf

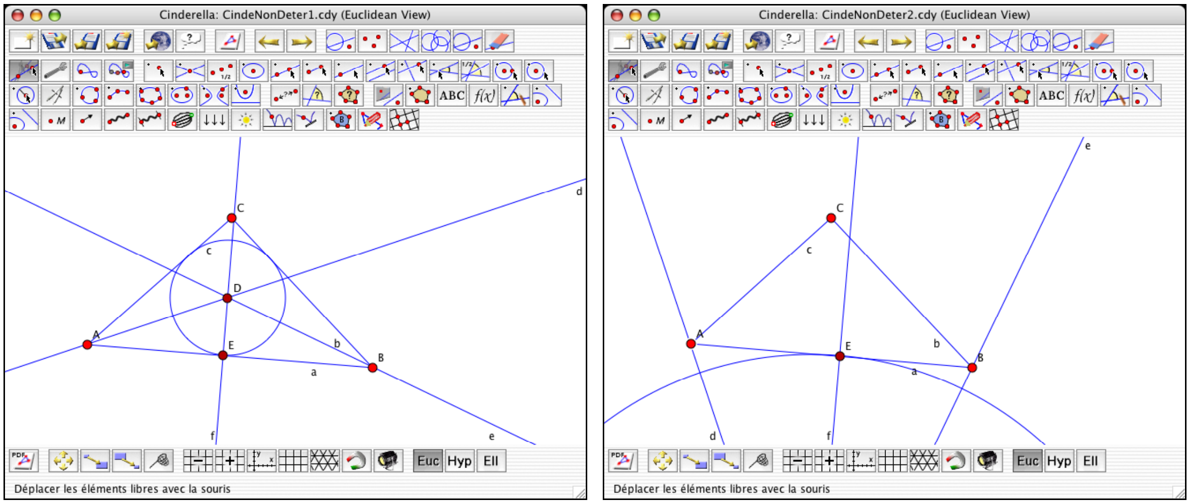


Figure 6.7.26 ***Cinderella*** **n'est pas *déterministe*. La figure de droite a été obtenue à partir de la figure de gauche en déplaçant le point *A* de façon à le faire passer par le point *B*. Même si le point *A* est revenu à sa position initiale, le cercle n'est plus inscrit dans le triangle.**

Nous terminons ici notre trop brève incursion dans le monde des logiciels de géométrie dynamique. Il est à prévoir que ces logiciels continueront d'évoluer, et que des progrès importants surviendront dans le cas de la géométrie dans l'espace. On peut même imaginer que, dans un avenir pas trop éloigné, des interfaces en 3D (telle l'immersion en réalité virtuelle illustrée à la figure 6.7.27) se démocratiseront et permettront de nouveaux modes d'interaction.

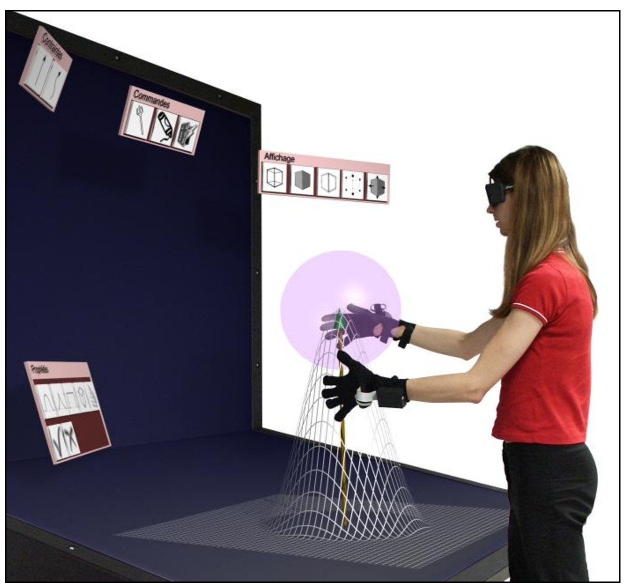


Figure 6.7.27 Une forme possible de réalité virtuelle : le sujet, muni de lunettes stéréoscopiques et de gants spéciaux, manipule gestuellement des objets tridimensionnels[1].

[1] Crédit photo : Université de Strasbourg, LSIIT, UMR CNRS 7005.
(Voir https://lsiitng.u-strasbg.fr/igg-fr/upload/2/27/Dogmerv.jpg)

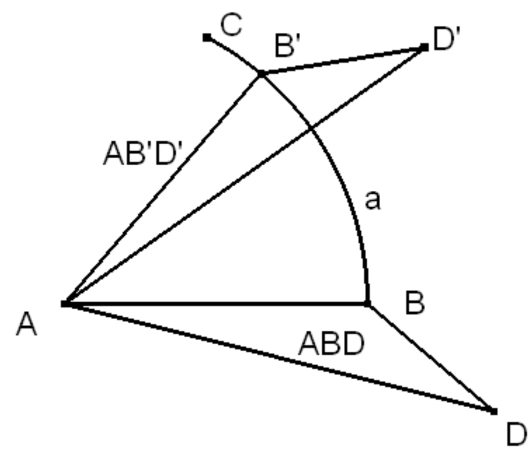
6.8 Exercices

1- *Construire un carré de différentes façons.*
Utilisez Cabri 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 : point libre, segment, cercle, intersection.

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 Cabri :
- Construisez un quadrilatère ABCD.
- Construisez les milieux E, F, G et H des côtés AB, BC, CD et DA respectivement.
- Construisez le quadrilatère EFGH.

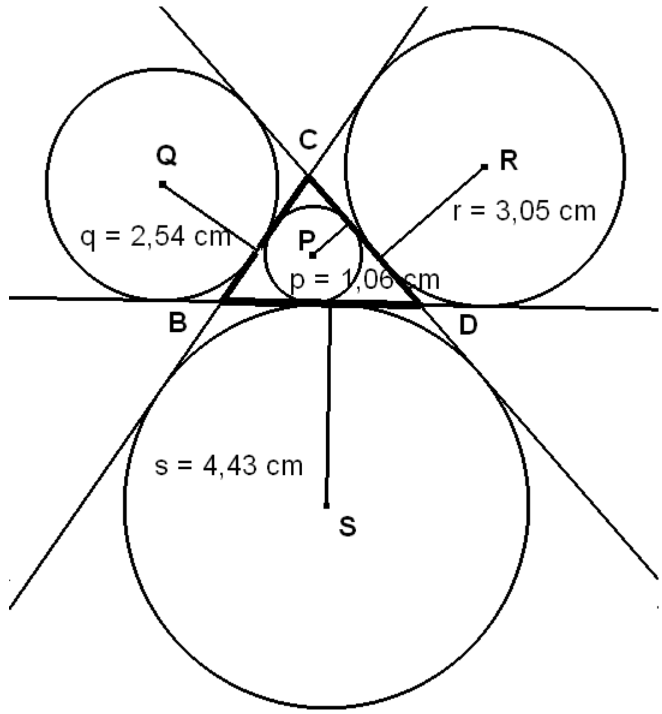
Faites une hypothèse sur la nature du quadrilatère EFGH et utilisez Cabri pour la vérifier.

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 Cabri 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.

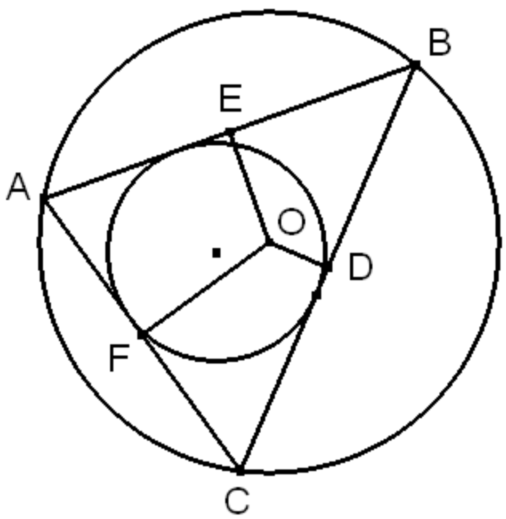
7- *Déterminer expérimentalement une relation.*

- 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.*



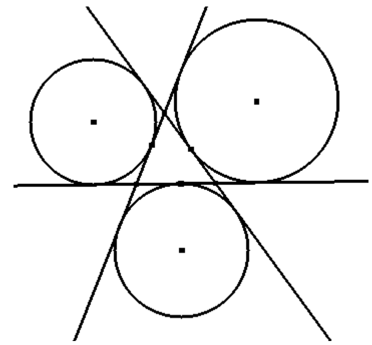
Étant donné un triangle ABC, son cercle circonscrit de centre O et de rayon R et son cercle inscrit de rayon r, on trace les segments OD, OE et OF perpendiculaires aux trois côtés. Un théorème de Lazare Carnot (dit Théorème japonais…) établit une relation du type :

$$\pm OD \pm OE \pm OF = R + r$$

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 Cabri pour déterminer la condition qui permet de d'associer un cas de figure avec la relation correspondante.

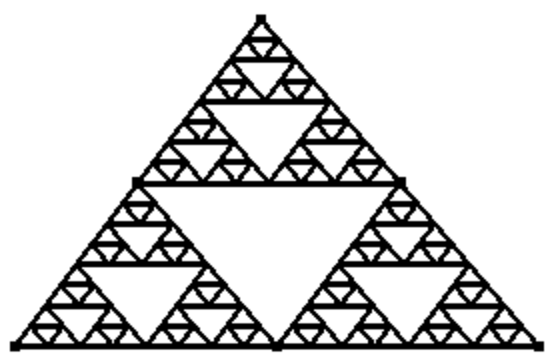
9- *Réaliser une macro-construction.*
Complétez la macro-construction « Perp » réalisée dans la section 6.3 pour qu'elle fonctionne aussi avec les objets rectilignes demi-droite et droite. Rappelons que cette macro construit le segment reliant un point donné au pied de la perpendiculaire à l'objet rectiligne.

10- *Réaliser une macro-construction.*
Réalisez une macro-construction qui, à partir de 3 points non alignés ou d'un triangle, 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



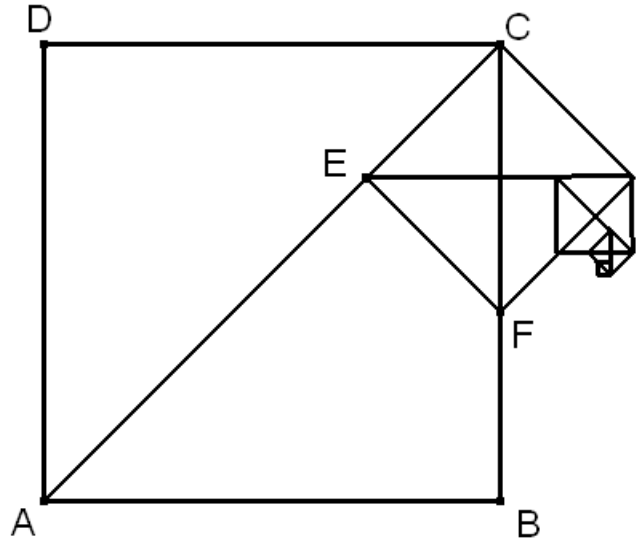
11- *Utiliser une boîte noire.*
Dans le contexte de *Cabri* on appelle « boîte noire » une figure dont une partie de la construction est cachée. Une telle figure est souvent utilisée en éducation, dans le but d'amener l'utilisateur à expérimenter avec la figure pour arriver à en déduire quelle est la construction cachée.
Une façon de cacher efficacement une construction c'est de la réaliser à l'aide d'une macro-construction affectée d'un mot de passe. La construction réalisée par la macro ne peut alors être visualisée sans la connaissance du mot de passe.
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 Cabri 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 accepter, indifféremment le triangle ABC ou ses 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 d'un carré ou de ses quatre sommets, construit le carré suivant. Utilisez ensuite cette macro pour obtenir la figure ci-dessus

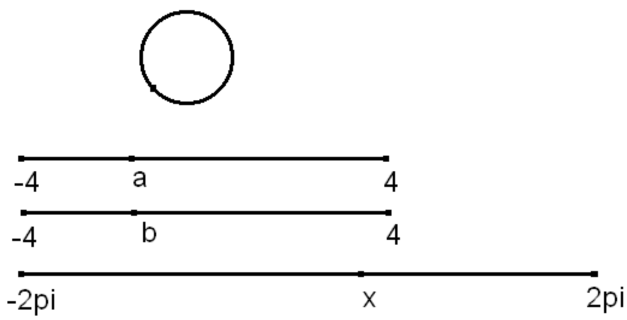
14- *Construire des coniques comme lieux de points.*

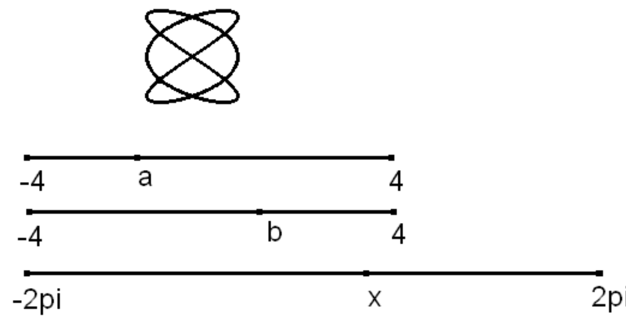
a) On peut définir une ellipse comme le lieu des points P tels que :
$$PF_1 + PF_2 = 2a$$
où F_1 et F_2 sont deux points distincts (les foyers) et a une constante positive ($F_1F_2 < 2a$).
F_1, F_2 et un segment (dont la longueur représente la constante 2a) étant donnés, utilisez la définition ci-dessus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



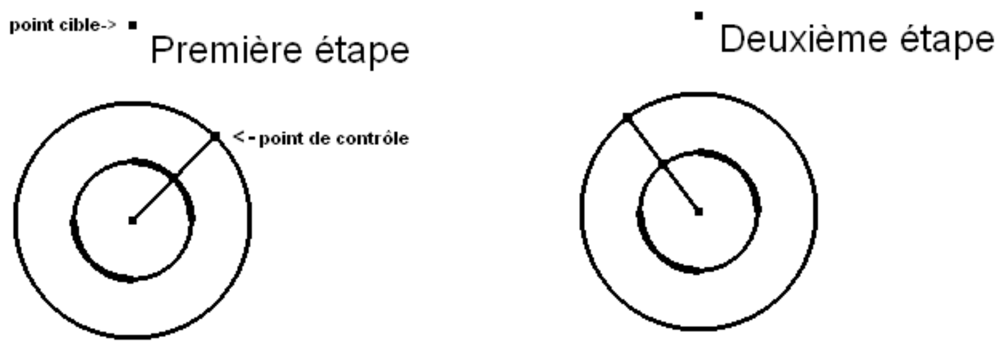


17- *Contruire un lieu géométrique qui dépend de paramètres.*
En vous basant sur les indications données dans la section 6.5 :

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.

6.9 Projets

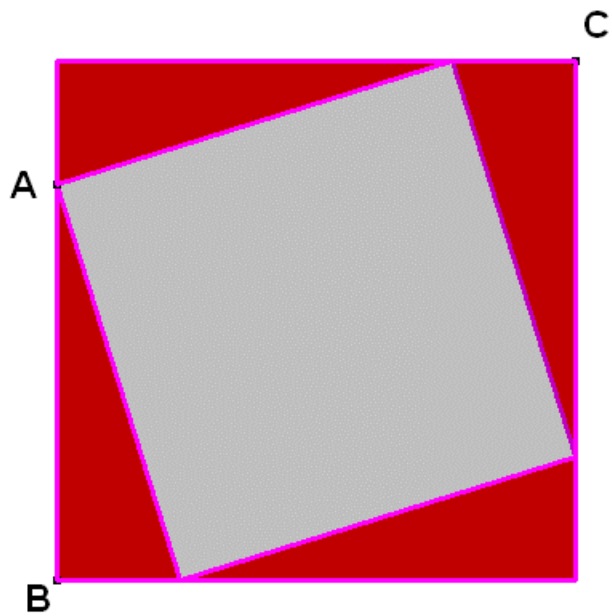
1- *Contruire une roulette de choix multiples.*
Quelques fois, lorsqu'on fournit une figure *Cabri* , on aimerait y inclure un mode d'emploi sous forme d'étapes successives. Dans ce projet vous allez réaliser une « roulette » qui permet d'afficher une seule étape à la fois. Vous pouvez en voir un exemple à la figure « Démo.fig », dans le dossier « Démo roulette » qui se trouve dans le dossier « Exemples chap.6» sur la page Web du livre. Les figures ci-dessous montrent les deux premiers états d'une « roulette à quatre états :



Le principe :
On part avec un cercle (qui supportera le point de contrôle) et un point cible (qui indiquera la position des textes). À partir du cercle initial on construit un second cercle, de même centre, qu'on divise en un certain nombre d'arcs (autant que de textes différents à afficher), ici quatre.
À chacun des arcs on associe un point, coincidant avec le point cible, qui n'existe que lorsque le segment joignant le centre du cercle initial au point de contrôle rencontre l'arc. Ici nous avons donc quatre points, qui coincident avec le point cible et qui n'existent pas en même temps.
Les textes à afficher sont les noms de ces points.

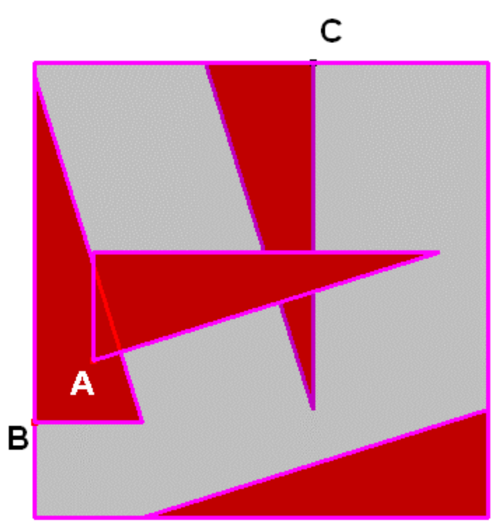
a) Réalisez une roulette à huit états.
b) À partir de votre roulette, réalisez une macro qui retourne une roulette fonctionnelle à laquelle il ne restera qu'à ajouter les textes appropriés.

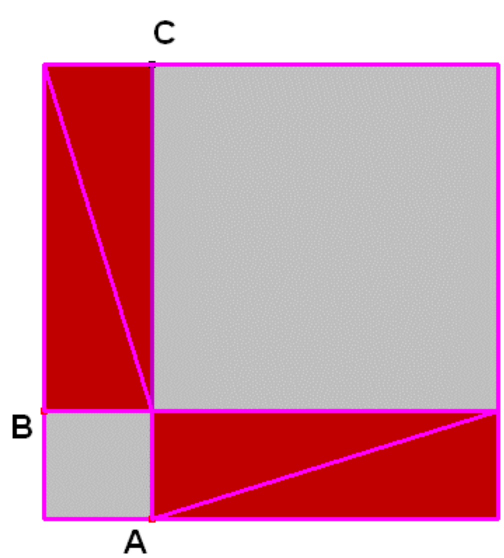
2- *Construire une démonstration « visuelle » du théorème de Pythagore.*
L'objectif de ce projet est de réaliser une figure *Cabri* illustrant une démonstration du théorème de Pythagore.



- 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 :

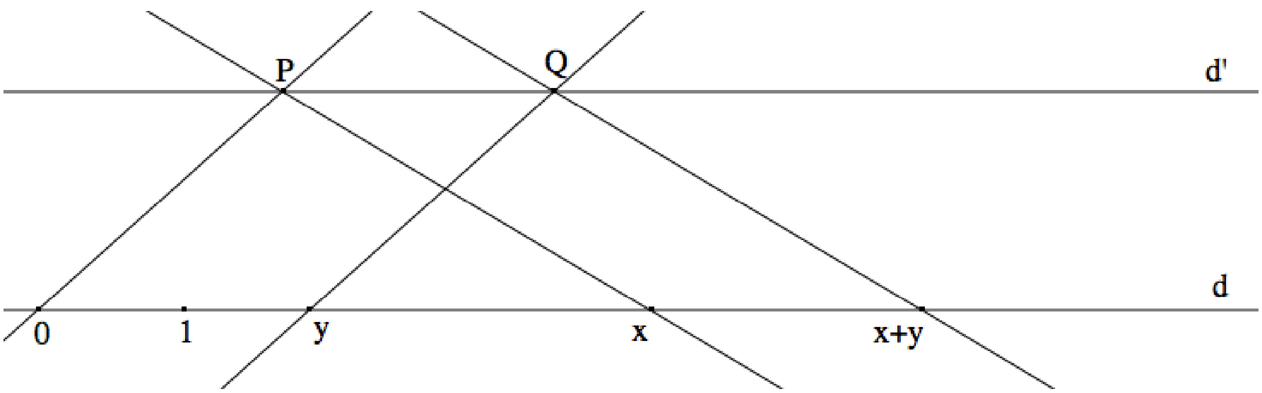




Dans la figure initiale le carré central est construit sur l'hypothénuse du triangle et, lorsqu'on compare avec la figure finale, on voit que son aire doit être égale à la somme des aires des deux carrés construit sur les côtés de l'angle droit.

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.

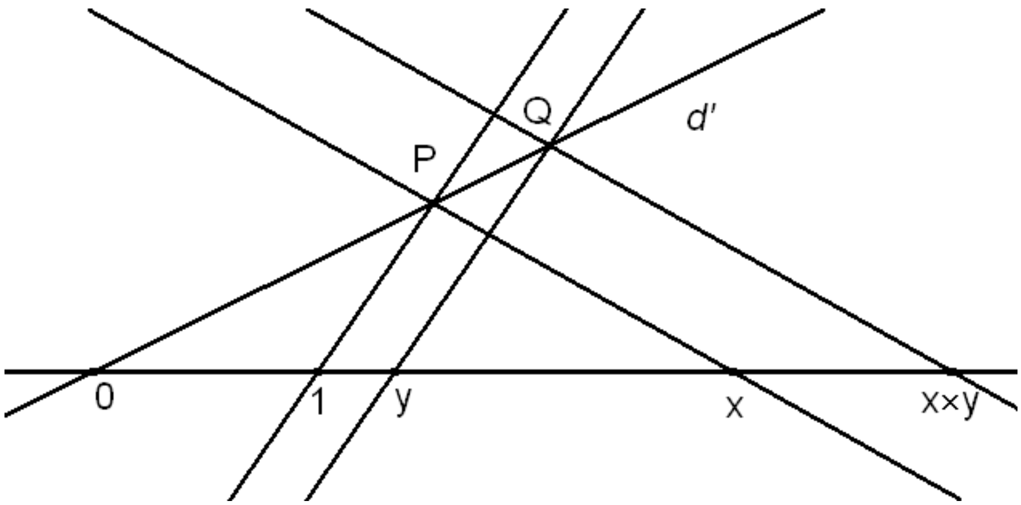
La soustraction
Au lieu de définir une soustraction, il est plus aisé de définir l'opposé -x de x (comme l'image de x par une symétrie de centre 0) et de poser

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

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

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

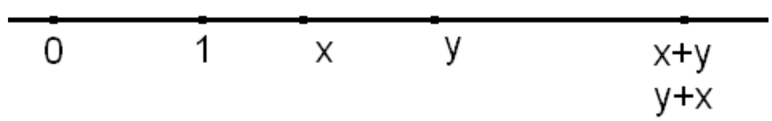
Des macros

Définissez ensuite les macros suivantes :

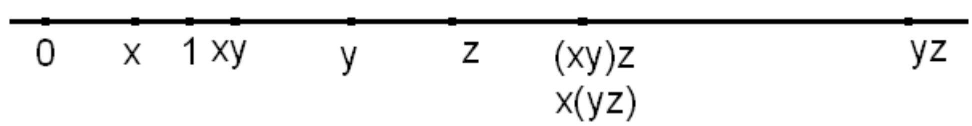
- Opposé : étant donnés l‘origine 0 et un point x, construit -x
- Inverse : étant donnés l’origine 0, l’unité 1 et un point x, construit x^{-1}
- Somme : étant donnés l’origine 0 ainsi que deux points x et y, construit x+y
- Produit : étant donnés l’origine 0, l’unité 1 ainsi que deux points x et y, construit x × y

et utilisez-les pour vérifiez les propriétés suivantes :

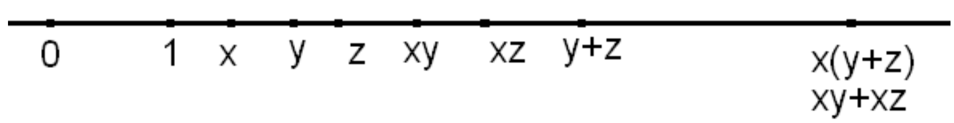
- la commutativité de l’addition : x + y = y + x



- l’associativité de la multiplication : x × (y × z) = (x × y) × z



- la distributivité de la multiplication sur l’addition : x × (y+z) = (x × y) + (x × 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- *Construire les tangentes communes à deux cercles et s'initier à la géométrie booléenne.*
Le but de ce projet est d'approfondir les constructions, problèmes et solutions mentionnés à la section 6.6.

- Utilisez la méthode suggérée à la figure 6.6.18 pour construire les tangentes à un cercle issues d'un point extérieur au cercle.
- Utilisez la méthode suggérée à la figure 6.6.19 pour construire les tangentes extérieures communes à deux cercles.
  Vérifiez que la construction est mise en défaut, lorsque le plus grand cercle devient le plus petit.

Introduction à la géométrie booléenne d'après E. Hakenholz (*abra*CA*da*BRI n 10 Sept-Oct 1995 ou http://icosaweb.ac-reunion.fr/GeomJava/abraCAda/Docs/LesAbras/alice.htm)
Expérience 1 :
Tracez une droite *d*, sur cette droite placez deux points A et B. Cachez la droite *d* et construisez la droite $\overleftrightarrow{AB}$. Placez maintenant un point P sur la droite $\overleftrightarrow{AB}$.
En déplaçant le point B, vous pouvez remarquer que le point P se déplace et en particulier que lorsque le vecteur $\overrightarrow{AB}$ change de sens il en est de même pour le vecteur $\overrightarrow{AP}$.
Ceci suggére que les droites de Cabri sont orientées et que la position des « points sur objet » placés sur ces droites dépend de cette oriention.
Expérience 2 :
Tracez une droite *d*, sur cette droite placez deux points A et B. Cachez la droite *d* et construisez la droite $\overleftrightarrow{AB}$. Par un point C extérieur à la droite $\overleftrightarrow{AB}$ construisez une parallèle *d*' à $\overleftrightarrow{AB}$ et placez un point P sur *d*'.
En déplaçant le point B, vous pouvez remarquer que lorsque le vecteur $\overrightarrow{AB}$ change de sens il en est de même pour le vecteur $\overrightarrow{CP}$.
Ceci suggére que la droite *d*' est aussi orientée et que son orientation dépend de l'orientation de $\overleftrightarrow{AB}$.
Expérience 3 :
Tracez une droite *d*, sur cette droite placez deux points A et B. Cachez la droite *d* et construisez la droite $\overleftrightarrow{AB}$.Construisez maintenant un cercle qui rencontre la droite $\overleftrightarrow{AB}$ en deux points, créez les points d'intersections et nommez les C et D.
En déplaçant les points A et B, vous pouvez vérifier que lorsque le vecteur $\overrightarrow{AB}$ change de sens *Cabri* échange les points C et D. La position de ces points d'intersections dépend donc aussi de l'orientation de la droite $\overleftrightarrow{AB}$.

Nous allons mettre à profit les informations tirées des expériences ci-dessus pour construire un point qui saute d'une position V à une position F en fonction de l'orientation comparée de deux droites :

- Construisez une droite *d* et placez deux point A et B sur cette droite. Construisez la droite $\overleftrightarrow{AB}$. Lorsque le vecteur $\overrightarrow{AB}$ change de sens l'orientation de la droite $\overleftrightarrow{AB}$ va changer mais celle de la droite *d* n'a aucune raison de changer. Si nous déclarons, arbitrairement, qu'initialement les deux droites ont même sens, nous pourrons, par un déplacement de B, avoir deux cas de figure : $\overrightarrow{AB}$ et *d* de même sens ou $\overrightarrow{AB}$ et *d* de sens opposés.
- Donnons nous maintenant deux points de base V (comme vrai) et F (comme faux). Le point M, que vous allez construire, coïncidera avec V si les droites ont même orientation et avec F dans le cas contraire.
- Créez le point I milieu du segment VF et le cercle C de centre I passant par V.
- Créez la parallèle à *d* passant par I, elle coupe le cercle C en deux points. Créez ces points et nommez les G et H.

  Créez la parallèle à $\overleftrightarrow{AB}$ passant par I, elle coupe le cercle C en deux points. Créez ces points et nommez les G' et H', en choisissant de faire coïncider G avec G' et H avec H'.
- Construisez le symétrique de G' par rapport à la bissectrice de l'angle VIG et nommez le M.

Initialement M coïncide avec V mais lorsque le vecteur $\overrightarrow{AB}$ change de sens M saute de V à F.

Utilisez maintenant la construction ci-dessus pour construire les tangentes extérieures communes à deux cercles.
Voici quelques indications pour vous aider :

- Soient C_1 et C_2 deux cercles de centres O_1 et O_2. C_1 initialement plus grand que C_2.
- $\overleftrightarrow{O_1O_2}$ la ligne des centres et d_1 et d_2 les perpendiculaires à $\overleftrightarrow{O_1O_2}$ passant par O_1 et O_2 respectivement. Soit A_1 l' intersection entre C_1 et d_1 et A_2 celle entre C_2 et d_2 qui est du même côté que A1 par rapport à $\overleftrightarrow{O_1O_2}$.
- Soit B la projection orthogonale de A2 sur d_1.

Vous pouvez constater que lorsqu'on change la taille des cercles, pour que le plus grand devienne le plus petit, le vecteur $\overrightarrow{A_1B}$ change de sens. En vous basant sur cette propriété, vous pouvez construire deux points dont l'un coïncide toujours avec le centre du plus grand cercle et l'autre avec le centre du plus petit cercle. À partir de ces deux points vous pouvez alors construire deux cercles dont l'un coïncide toujours avec le plus grand cercle et l'autre avec le plus petit cercle. C'est sur ces deux derniers cercles qu'on fait la construction des tangentes communes.

Index

V