

Lycée Bellepierre – Stage LaTeX

Jean-Luc PONCIN

Juin 2014

Résumé

Ce stage de 3 heures est destiné à des professeurs de mathématiques débutant en \LaTeX . L'objectif est qu'un participant au stage apprenne quelques outils simples pour obtenir un sujet de devoir et son corrigé.

Le logiciel utilisé est *Texmaker*.

Table des matières

1	Première approche, structure du document, préambule	5
1.1	Structure du document et préambule	5
1.2	Compiler et visionner le document	5
1.3	Une première expression mathématique, aller à la ligne	5
1.4	Compilation rapide et basculement éditeur afficheur pdf	6
1.5	Une fraction plus grande, et un problème de module (package)	6
1.6	Assistant nouveau document	6
2	Écritures mathématiques	7
2.1	Un premier exemple, raccourcis au clavier, boutons, menus	7
2.1.1	Utiliser des boutons pour faciliter la saisie.	7
2.1.2	Utiliser des raccourcis au clavier pour faciliter la saisie.	7
2.1.3	Utiliser des menus pour faciliter la saisie.	8
2.2	Lettres, symboles, pointillés	8
2.2.1	Tableau de symboles	8
2.2.2	Le symbole <i>parallèle</i>	8
2.2.3	Pointillés	9
2.3	Expressions mathématiques	9
2.3.1	Fractions, racines, exposants indices	9
2.3.2	Fonctions, limites, intégrales	9
2.3.3	Parenthèses, crochets, accolades, valeurs absolues	9
2.3.4	Texte dans les calculs	10
2.4	Exercice 1, sur les écritures mathématiques	10
3	La complétion automatique dans Texmaker	10
4	Le texte	11
4.1	Taille des caractères	11
4.1.1	Un premier exemple	11
4.1.2	Aide à la saisie, les différentes tailles des caractères	11
4.1.3	Ces tailles de caractères sont relatives.	11
4.2	Style des caractères : avec ou sans empattement, machine à écrire	12
4.2.1	Style avec ou sans empattement : Times New Roman / Arial	12
4.2.2	Style machine à écrire (typewriter type)	12
4.2.3	Aide à la saisie des styles de caractères	12
4.3	Gras, italique, souligné	12
4.3.1	Gras	13

4.3.2	Italiques	13
4.3.3	Souligné	13
4.4	Combinaisons de tailles, styles, etc.	13
4.5	Caractères spéciaux : dollar, accolade, pourcent,	13
4.5.1	Le caractère % – Insertion de commentaire non compilé	14
4.5.2	Afficher un caractère spécial pour lui même	14
4.6	Espaces horizontaux	14
4.6.1	Blancs, espaces entre mots	14
4.6.2	Espaces élastiques et <code>hfill</code>	14
4.6.3	Tirets	14
4.6.4	Points de suspension	14
4.6.5	Pointillés	14
4.7	Espaces verticaux	15
4.7.1	Aller à la ligne	15
4.7.2	Espaces verticaux supplémentaires	16
4.7.3	Saut de page	16
4.8	Alignement du texte, centrer du texte	16
4.8.1	Alignement du texte à gauche	16
4.8.2	Texte centré	17
4.8.3	Alignement du texte à droite	17
4.9	Indentation	18
4.10	Les listes	18
4.10.1	Listes numérotées	18
4.10.2	Listes à puces	19
4.11	Exposant en mode texte : <code>fbox</code> et <code>boxed</code>	19
4.12	Encadrer en mode texte ou mathématiques	19
4.13	Exercice 2, sur le texte et les listes	19
5	Environnement	20
6	Tableaux	20
6.1	Un premier exemple.	20
6.2	Utiliser l’assistant de tableau	21
6.3	Exercice 3, sur les tableaux	22
7	Équation, alignements	22
7.1	Environnement <code>eqnarray*</code>	22
7.2	Environnement <code>align*</code> du module <code>amsmath</code>	23
7.3	Environnement <code>array</code> du module <code>array</code>	23
7.4	Exercice 4, équation	24

8	Matrices et systèmes	24
8.1	Matrices	24
8.2	Systèmes	24
8.3	Exercice 5, matrice et système	24
9	Graphisme en PDFLaTeX	24
9.1	Inclure un fichier externe png, jpeg ou pdf	25
9.1.1	La commande <code>includegraphics</code> avec fichier png	25
9.1.2	Créer un fichier image avec GeoGebra	25
9.1.3	Exercice 6, insérer une figure	26
9.2	Tikz	26
9.2.1	Description de Tikz	26
9.2.2	Un exemple de figure avec Tikz	26
9.2.3	Courbe de fonction, codage Tikz avec GeoGebra	27
9.3	Tableau de variation avec Tikz	28
9.4	Arbre pondéré avec Tikz	29
9.5	Améliorer les tableaux et arbres obtenus avec TikZ	30
10	Graphisme en LaTeX+dvips+ps2pdf	30
10.1	Compilation du fichier par LaTeX+dvips+ps2pdf	30
10.2	Inclure un fichier externe au format eps	30
10.3	PSTricks et LaTeX+dvips+ps2pdf	31
10.3.1	Description de PSTricks	31
10.3.2	Un exemple de figure avec PSTricks	31
10.3.3	Courbe de fonction, codage PSTricks avec GeoGebra	32
10.4	Tableau de variation avec PST+	33
10.5	Arbres pondérés et module pstree	34
11	Mise en page, gagner de la place	35
11.1	Disposition sur deux colonnes	35
11.2	Disposition en deux blocs et environnement <i>minipage</i>	36
12	Diverses choses	37
12.1	Structure du document : sections, sous sections, etc.	37
12.2	Format d’encodage des fichiers : UTF-8, ISO-8859-15 – Logiciel Gedit	37
12.3	Utilisation de LaTeX dans d’autres logiciels	38
12.4	Quelques macrocommandes	38
12.5	Un préambule commun au lycée ?	38

13 Après le stage	39
13.1 Livres sur LaTeX	39
13.2 Documentation sur Internet	39
13.2.1 Le site de documentation de L ^A T _E X et Texmaker	39
13.2.2 Le fichier pdf “LatexPourProfMaths” par Arnaud GAZAGNES	39
13.2.3 Documentation sur TikZ	39
13.2.4 Documentation sur PSTricks	39
13.3 Installations sous Windows	40
13.3.1 Installation MiKTeX avec proTeXt	40
13.3.2 Installation Texmaker	41
13.3.3 Installation PST plus	41
13.3.4 Installation Gedit	41

Après l'allumage de l'ordinateur, cliquer sur **Démarrer**, puis sur **Fermer la session**, et lancer une session **prof** au lieu d'une session **eleve**.

1 Première approche, structure du document, préambule

Ouvrir le logiciel *Texmaker* (raccourci sur le bureau).

Ouvrir le fichier `stage-latex-13-06-14.tex` qui se trouve dans **Mes documents**.

1.1 Structure du document et préambule

Le fichier que l'on vient d'ouvrir contient ceci :

```
\documentclass[12pt,a4paper]{article}
\usepackage[utf8]{inputenc}
\begin{document}
```

Une simple phrase pour commencer, mais ce n'est qu'un début et nous en arriverons à apprendre de quoi obtenir un sujet de devoir.

```
\end{document}
```

Quelques explications :

- Il y a deux parties : les 2 premières lignes qui constituent le préambule, et le contenu du texte entre les commandes `\begin{document}` et `\end{document}`

- **Le préambule et les packages**

La commande `\documentclass[12pt,a4paper]{article}` indique que les caractères seront en 12 points, que le format de feuille sera du A4, et que le type de document sera un « article », ce qui conviendra pour un sujet de devoir.

La commande `\usepackage[utf8]{inputenc}` est relative au codage du fichier texte actuel. Nous y reviendrons.

1.2 Compiler et visionner le document

Nous allons maintenant faire ce qu'on appelle « compiler » ce fichier. Pour cela, aller dans le menu *Outils*, et cliquer sur **PDFLaTeX**. On voit alors au bas de l'écran les phrases **Process started** et **Process exited normally** indiquant que tout s'est bien passé.

Pour afficher le fichier pdf, aller dans le menu *Outils*, et cliquer sur **Voir PDF**.

Une première remarque : la mise en page laisse des marges trop importantes au dessus et sur les côtés, nous allons bientôt y remédier.

Pour revenir à l'éditeur de texte, appuyer sur la combinaison de touches **Ctrl**+**Espace**. Ce dernier raccourci au clavier permet en fait de basculer de l'afficheur pdf à l'éditeur de texte et inversement.

1.3 Une première expression mathématique, aller à la ligne

Ajoutons une première expression mathématique : à la fin du texte précédent « Une simple phrase ...de devoir. », aller à la ligne et ajouter ceci : `$3 \times 6^2 - \frac{3}{7}$`

Enregistrer le fichier.

Pour compiler à nouveau et afficher le nouveau fichier pdf, il sera désormais plus simple d'appuyer simplement sur la touche **F1**.

Dans le fichier pdf on voit alors qu'il n'y a pas de retour à la ligne avant l'expression $3 \times 6^2 - \frac{3}{7}$.

Revenir à l'éditeur de texte (combinaison de touches **Ctrl**+**Espace**).

En \LaTeX , pour obtenir ce retour à la ligne, il faut intercaler un saut de ligne supplémentaire. Intercaler ce saut de ligne, enregistrer et compiler (touche **F1**).

1.4 Compilation rapide et basculement éditeur afficheur pdf

Voici ce qui sera utilisé désormais :

- pour compiler et afficher le fichier pdf : appuyer sur la touche **F1**;
- pour basculer entre l'éditeur et l'afficheur pdf : utiliser la combinaison de touches **Ctrl**+**Espace**.

1.5 Une fraction plus grande, et un problème de module (package)

Autre chose maintenant : au lieu de la fraction sous la forme $\frac{4}{7}$, on préfère en général la fraction sous la forme $\frac{4}{7}$.

Sauter une ligne, saisir ce qui suit, puis enregistrer et compiler : `\dfrac{4}{7}`

Il y a alors un problème, et on voit en bas le message ci-dessous en rouge, qui indique que « `\dfrac` » est une commande inconnue

stage-latex-fichier-d-essais.tex Error line 8 ! Undefined control sequence.<recently read> \dfrac `\dfrac`

En fait, la commande `dfrac` fait partie d'un module (ou package) de LaTeX qu'il faut « appeler » en ajoutant ceci au préambule actuel :

```
\usepackage{amsmath}
```

Le préambule est maintenant :

```
\documentclass[12pt,a4paper]{article}
\usepackage[utf8]{inputenc}
\usepackage{amsmath}
```

Enregistrer, puis appuyer sur **F1**.

Il n'y a plus de message d'erreur et, dans le fichier pdf, la fraction $\frac{4}{7}$ apparaît normalement.

À partir d'ici, chaque fois qu'il faudra enregistrer et compiler la consigne sera simplement « compiler ».

1.6 Assistant nouveau document

Nous allons maintenant reprendre la création d'un document avec un préambule plus complet, sans avoir besoin de tout saisir, et ce préambule va remédier au problème de mise en page.

Effacer d'abord tout le contenu du fichier, puis, dans le menu **Assistants** ci-dessus, cliquer sur **Assistant nouveau document**, et compléter la fenêtre comme cela est indiqué ci-dessous.

Classe document : article

Taille police : 12pt

Format papier : a4paper

Encodage : utf8

Cocher babel Package, et choisir francais

Cocher geometry Package

Cocher Paquets AMS

Cocher graphicx Package

Cliquer sur

On voit alors le préambule ci-dessous, beaucoup plus complet.

```
\documentclass[12pt,a4paper]{article}
\usepackage[utf8]{inputenc}
\usepackage[francais]{babel}
\usepackage[T1]{fontenc}
\usepackage{amsmath}
\usepackage{amsfonts}
\usepackage{amssymb}
\usepackage{graphicx}
\usepackage[left=2cm,right=2cm,top=2cm,bottom=2cm]{geometry}
```

Saisir du texte sur un peu plus d'une ligne entre les commandes `\begin{document}` et `\end{document}`, puis (enregistrer et) compiler. On voit alors une marge de 2 cm en haut et sur les côtés. Ces réglages apparaissent dans la dernière ligne du préambule (ci-dessus), et peuvent être modifiés.

2 Écritures mathématiques

2.1 Un premier exemple, raccourcis au clavier, boutons, menus

2.1.1 Utiliser des boutons pour faciliter la saisie.

Pour obtenir $\sqrt{2}$, il faut saisir `\sqrt{2}`, mais dans Texmaker existent des raccourcis au clavier et des boutons pour faciliter la saisie.

Commençons par les signes \$: cliquer sur le bouton ici à gauche de la zone de saisie, puis cliquer sur le bouton qui lui aussi se trouve ici à gauche de la zone de saisie.

Il ne reste plus qu'à saisir le chiffre 2.

2.1.2 Utiliser des raccourcis au clavier pour faciliter la saisie.

Pour l'exemple ci-dessus, on pouvait aussi utiliser des raccourcis au clavier

- pour obtenir les signes \$: Maj+Ctrl+M
- pour obtenir la racine carrée : Maj+Ctrl+Q

2.1.3 Utiliser des menus pour faciliter la saisie.

Cliquer maintenant sur le menu Math

Pour obtenir les signes \$, cliquer sur la ligne intitulée Math en ligne \$...\$.

Pour obtenir la racine carrée cliquer sur la ligne intitulée `\sqrt{}`






On peut remarquer que comme dans d'autres logiciels, les raccourcis au clavier sont indiqués dans ce menu Math.

Les paragraphes 2.2 et 2.3 qui suivent ne comportent pas de consigne et indiquent comment obtenir différentes écritures mathématiques. Lire ces deux paragraphes, en faisant éventuellement quelques essais, puis faire l'exercice du paragraphe 2.4.

2.2 Lettres, symboles, pointillés

2.2.1 Tableau de symboles

Voici un tableau donnant une série de symboles, le codage en \LaTeX à gauche et l'affichage obtenu à droite.

On retrouve ces symboles et d'autres à l'aide des boutons      qui se trouvent sur le bord gauche de la fenêtre.

<code>\delta</code>	δ	<code>\approx</code>	\approx	<code>\iff</code>	\iff
<code>\Delta</code>	Δ	<code>c \geqslant d</code>	$c \geqslant d$	<code>\Longleftarrow</code>	\Longleftarrow
<code>\mathbb{R}</code>	\mathbb{R}	<code>a \leqslant b</code>	$a \leqslant b$	<code>\longrightarrow</code>	\longrightarrow
<code>\text{\kern-0.21emR}</code>	\mathbb{R}	<code>\in</code>	\in	<code>\overline{A}</code>	\overline{A}
<code>\mathcal{C}</code>	\mathcal{C}	<code>\notin</code>	\notin	<code>\overrightarrow{AB}</code>	\overrightarrow{AB}
<code>\mathscr{C}</code>	\mathscr{C}	<code>\subset</code>	\subset	<code>\widehat{ABC}</code>	\widehat{ABC}
<code>\ell</code>	ℓ	<code>\cup</code>	\cup		
<code>\infty</code>	∞	<code>\cap</code>	\cap		
<code>\times</code>	\times	<code>\equiv</code>	\equiv		
<code>\%</code>	$\%$				

Dans le tableau ci-dessus, la deuxième façon d'afficher le symbole \mathbb{R} est obtenue par le codage `\text{\kern-0.21emR}` proposé par un collègue sur Internet. Ce type de codage un peu long peut être automatisé dans ce qu'on appelle une macro-commande, mais nous n'en sommes pas encore là.

2.2.2 Le symbole *parallèle*

Voici ce qui est prévu en \LaTeX : on saisit `\parallel` et on obtient : \parallel

On pourrait saisir ceci : `//` et obtenir ceci : `//`

Les signes obliques sont trop espacés, cela n'est pas satisfaisant.

Voici une possibilité pour obtenir ce symbole.

Insérer dans le préambule la ligne de commandes : `\usepackage{stmaryrd}`

Quand on saisit la commande `\sslash` on obtient : \parallel

2.2.3 Pointillés

En voici deux types ci-dessous. On peut observer les différences de hauteurs des pointillés par rapport aux autres caractères.





<code>\$a + \cdots + b\$</code>	$a + \dots + b$	<code>\$x + \ldots = 5\$</code>	$x + \dots = 5$
---------------------------------	-----------------	---------------------------------	-----------------

2.3 Expressions mathématiques

2.3.1 Fractions, racines, exposants indices

<code>\$\$\dfrac{3}{5}\$\$</code>	$\frac{3}{5}$	<code>\$u_n\$</code>	u_n	<code>\$\$a^n \times a^p = a^{n+p}\$\$</code>	$a^n \times a^p = a^{n+p}$
<code>\$\$\sqrt{8}\$\$</code>	$\sqrt{8}$	<code>\$u_{n+1}\$</code>	u_{n+1}		
<code>\$\$\dbinom{8}{9}\$\$</code>	$\binom{8}{9}$				

Là encore, plusieurs boutons et raccourcis au clavier existent pour faciliter la saisie :

- les boutons  (indice),  (exposant),  (fraction),  (racine);
- les raccourcis au clavier sont indiqués dans le menu **Math**.

2.3.2 Fonctions, limites, intégrales

<code>\$f:x \longmapsto f(x)\$</code>	$f : x \mapsto f(x)$	<code>\$\$\ln(x)\$</code>	$\ln(x)$	<code>\$\$e^x\$</code>	e^x
		<code>\$\$\cos(x)\$</code>	$\cos(x)$	<code>\$\$\text{e}^x\$</code>	e^x

<code>\$\$\lim\limits_{x \to +\infty} f(x)\$</code>	$\lim_{x \rightarrow +\infty} f(x)$
<code>\$\$\lim\limits_{\begin{subarray}{l} x>0 \\ x \to 0 \end{subarray}} f(x)\$</code>	$\lim_{\substack{x>0 \\ x \rightarrow 0}} f(x)$
<code>\$\$\displaystyle \int_2^5 f(x) dx\$</code>	$\int_2^5 f(x) dx$
<code>\$\$[F(x)]_3^6\$</code>	$[f(x)]_3^6$

2.3.3 Parenthèses, crochets, accolades, valeurs absolues

Crochets : trois exemples pour un intervalle ci-dessous, on peut remarquer qu'on peut l'écrire sans les signes \$, mais alors le signe moins est étroit, d'autre part, si l'on veut des espaces autour du point-virgule, il faut utiliser le blanc insécable (~).


<code>\$\$[-3~;~4]\$</code>	$[-3 ; 4]$	<code>\$\$[-3;4]\$</code>	$[-3;4]$	<code>\$\$[-3;4]\$\$</code>	$[-3;4]$	<code>\$\$[-3;4]\$\$</code>	$[-3;4]$
-----------------------------	------------	---------------------------	----------	-----------------------------	----------	-----------------------------	----------

Accolades : avec le codage : `\{1, 2, 3\}` on obtient : $\{1, 2, 3\}$

Grandes parenthèses et grandes barres de valeur absolue

Voici deux exemples :

<code>\left(1 - \dfrac{2}{3} \right) \times 5</code>	$\left(1 - \frac{2}{3} \right) \times 5$
<code>\left \dfrac{2}{3} \right </code>	$\left \frac{2}{3} \right $

La saisie pour ces grandes parenthèses et grandes barres de valeur absolue est facilitée en utilisant le bouton  sur le bord gauche de la fenêtre.

2.3.4 Texte dans les calculs

Sil'on saisit ceci : `$Valeur finale - valeur initiale$` on obtient : *Valeur finale–valeur initiale*

Le texte est donc en italiques et avec un espacement anormal entre les lettres et pas d'espace entre deux mots.

Pour résoudre ce problème, on peut utiliser la commande `\text{}` du module Amsmath :

codage : `$$\text{Valeur finale} - \text{valeur initiale}$$`

affichage : Valeur finale – valeur initiale

2.4 Exercice 1, sur les écritures mathématiques

Il faut obtenir l'affichage des expressions mathématiques du tableau ci-dessous.

$p_A(B) = \frac{p(A \cap B)}{p(A)}$	$\ln\left(\frac{x}{x+1}\right)$	$\left \frac{1}{z}\right $
$p(X = k) = \binom{n}{k} p^k (1-p)^{n-k}$	cosinus = $\frac{\text{côté adjacent}}{\text{hypoténuse}}$	\mathcal{C}_f
$(\overrightarrow{OA}, \overrightarrow{OB}) = \arg\left(\frac{a}{b}\right) = \frac{\pi}{2}$	$u_n = e^{\ln n - n \ln 2}$	$I_n = \int_0^1 x^n e^{x^2} dx$
$\binom{n+3}{2} = \frac{(n+3)(n+2)}{2}$	$] - \infty ; 0[\cup] 0 ; \infty[$	$1 + q^2 + q^3 + \dots + q^n = \frac{1 - q^n}{1 - q}$
$p(X = 7)$ (avec le X droit)	$\Omega M = 5 \text{ cm}$	

3 La complétion automatique dans Texmaker

Comme beaucoup de logiciels ou d'éditeurs de textos, Texmaker a un système de complétion automatique, c'est à dire que dès qu'on commence à saisir un code connu, le logiciel fait apparaître une petite liste, dans laquelle il suffit de sélectionner le code que l'on voulait saisir.

Le code apparaît parfois avec un ou plusieurs gros points noirs, qui correspondent à des zones de texte à compléter. On passe d'un point noir à l'autre avec la touche tabulation.

4 Le texte

4.1 Taille des caractères

4.1.1 Un premier exemple

Si on saisit : `{\Large Blabla}` on obtient : **Blabla**


Si l'on veut agrandir une portion de texte plus longue, on saisit :

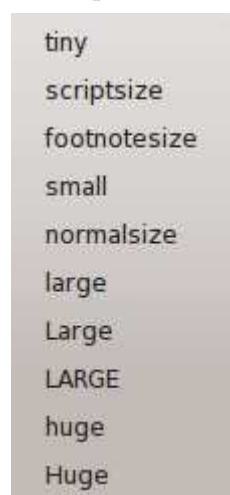
```
\begin{Large}
Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
blabla blabla blabla blabla blabla
\end{Large}
```

et on obtient :

Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla

4.1.2 Aide à la saisie, les différentes tailles des caractères

La saisie est facilitée dans *Texmaker* en cliquant sur le bouton  qui se trouve en haut de cette fenêtre. On voit alors la liste ci-dessous, indiquant la liste des tailles de caractères disponibles, de la plus petite **tiny** à la plus grande **Huge** en passant par la taille normale **normalsize**. Il suffit alors de cliquer sur la taille recherchée.



4.1.3 Ces tailles de caractères sont relatives.

Une précision : toutes ces tailles de caractères sont relatives, c'est à dire que **normalsize** est la taille de caractère déclarée en tout début de fichier (12 points par exemple), la taille en dessous **small** est un peu plus petite et **footnotesize** encore un peu plus petite et ainsi de suite . . . , par conséquent, si dans une autre version du texte on décidait de modifier la taille de caractère déclarée en tout début de fichier (10 points au lieu de 12 points par exemple), les tailles de caractères seraient modifiées proportionnellement.

4.2 Style des caractères : avec ou sans empattement, machine à écrire

4.2.1 Style avec ou sans empattement : Times New Roman / Arial

Les styles de caractères couramment utilisés dans les traitements de texte comme *Word* ou *LibreOffice* sont obtenus à partir des polices *Times New Roman* et *Arial*.

En Times New Roman, la lettre M ressemble à ceci : **M**, et en Arial, elle ressemble à ceci : **M**

On dit que la première est une police *avec empattement* (*roman* ou *serif*) et la deuxième sans empattement (*sans serif*).

En \LaTeX il y a une seule police, avec différents styles.

Le style courant est avec empattement, et pour obtenir le style sans empattement, voici un exemple :

si on saisit : `\textsf{T'as vu mon style~!}`, on obtient : T'as vu mon style!

et si on saisit :

```
\begin{sffamily}
Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
blabla blabla blabla blabla blabla blabla blabla blabla.
\end{sffamily}
```

on obtient :

Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla.

4.2.2 Style machine à écrire (typewriter type)

Un autre style est parfois utilisé, le style machine à écrire. Dans les autres traitements de texte, on l'obtient par exemple avec la police *Courier*.

Ce style est utilisé pour donner des extraits de programme informatique, du code \LaTeX , ou des adresses Internet, comme dans l'exemple ci-dessous.

Code : `\texttt{http://www.maths.lyceebellepierre.fr}`

Affichage : `http://www.maths.lyceebellepierre.fr`

4.2.3 Aide à la saisie des styles de caractères

Voir menu \LaTeX , rubrique Styles de caractères. Des raccourcis au clavier sont indiqués

4.3 Gras, italique, souligné

Pour faciliter la saisie on pourra utiliser les boutons indiqués ci-dessous, que l'on voit sur le bord gauche de la zone de texte.



4.3.1 Gras

Si on saisit : `\textbf{Blabla}`, on obtient : **Blabla**

et si on saisit :

```
\begin{bfseries}
Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
blabla blabla blabla blabla blabla blabla blabla blabla.
\end{bfseries}
```

on obtient :

Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla.

4.3.2 Italiques

Si on saisit : `\textit{Blabla}`, on obtient : *Blabla*

et si on saisit :

```
\begin{itshape}
Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
blabla blabla blabla blabla blabla blabla blabla blabla.
\end{itshape}
```

on obtient :

Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla.

4.3.3 Souligné

Si on saisit : `\underline{Blabla}`, on obtient : Blabla

4.4 Combinaisons de tailles, styles, etc.

Le principe est d'emboîter les commandes des paragraphes précédents. Voici deux exemples ci-dessous.

Gras italique : avec `\textbf{\textit{Blabla}}`, on obtient ***Blabla***

Gras et grand : avec `\textbf{\large Blabla}` on obtient **Blabla**

4.5 Caractères spéciaux : dollar, accolade, pourcent, ...

Comme on l'a déjà constaté, certains caractères tels que le dollar, ou l'accolade jouent un rôle particulier en \LaTeX .

Deux autres caractères jouent un rôle particulier : `&` et `%`

Le symbole `&` intervient dans les tableaux, nous y viendrons plus loin.

4.5.1 Le caractère % – Insertion de commentaire non compilé

Si on saisit ce qui suit : `Blabla % blabla` on obtient : Blabla

On constate donc que le deuxième mot n'apparaît pas. En \LaTeX , ce qui est écrit entre un signe % et le bout de la ligne est considéré comme un commentaire, et ne sera pas compilé, \LaTeX n'en tiendra aucun compte. C'est pourquoi dans Texmaker, dans la zone de texte le caractère % et le deuxième mot « blabla » apparaissent en gris.

4.5.2 Afficher un caractère spécial pour lui même

Il arrive que l'on veuille afficher un tel caractère pour lui même. Dans ce cas, il faut le faire précéder d'une contre-oblique.

<code>\\$</code>	<code>\$</code>	<code>\{</code>	<code>{</code>	<code>\%</code>	<code>%</code>
<code>\&</code>	<code>&</code>	<code>\}</code>	<code>}</code>		

4.6 Espaces horizontaux

4.6.1 Blancs, espaces entre mots

L'exemple ci-dessous illustre quatre façons d'augmenter l'espace entre deux mots.

Codage : `Blabla blabla \ blabla \quad blabla \quad\quad blabla \hspace{2cm} blabla`

Affichage : Blabla blabla blabla blabla blabla blabla

4.6.2 Espaces élastiques et `hfill`

La commande `\hfill` insère un espace élastique qui repousse le texte à gauche et à droite de telle façon que le texte de la ligne occupe toute la largeur de la page, comme dans l'exemple ci-dessous.

Codage : `Blabla blabla blabla blabla \hfill blabla blabla blabla blabla`

Affichage :

Blabla blabla blabla blabla

blabla blabla blabla blabla

4.6.3 Tirets

Comme le montre l'exemple ci-dessous, il existe trois types de tirets :

Codage : `blabla - blabla -- blabla --- blabla`

Affichage : blabla - blabla – blabla — blabla

4.6.4 Points de suspension

Codage : `blabla \dots blabla` **Affichage :** blabla ...blabla

4.6.5 Pointillés

Codage : `Blabla~: \dotfill`

Affichage : Blabla :

Codage : `Blabla \dotfill blabla \dotfill \dotfill blabla \dotfill`

Blablablablablabla

4.7 Espaces verticaux

4.7.1 Aller à la ligne

Comme cela a déjà été évoqué, pour aller à la ligne dans le texte affiché, il faut saisir un saut de ligne en \LaTeX . On peut aussi utiliser la commande \par

Exemple :

Le codage `Blabla \par blabla` donne ceci :

Blabla

blabla

4.7.2 Espaces verticaux supplémentaires

Quatre type d’espaces entre lignes existent, ce qui est illustré dans l’exemple ci-dessous.

Codage :

```
blabla blabla blabla blabla \par
blabla blabla blabla blabla \par
\smallskip
blabla blabla blabla blabla \par
\medskip
blabla blabla blabla blabla \par
\bigskip
blabla blabla blabla blabla \par
\vspace{1cm}
blabla blabla blabla blabla
```

Affichage :

```
blabla blabla blabla blabla
blabla blabla blabla blabla
blabla blabla blabla blabla
blabla blabla blabla blabla
blabla blabla blabla blabla
blabla blabla blabla blabla
blabla blabla blabla blabla
```

4.7.3 Saut de page

Pour forcer un saut de page, il suffit de saisir la commande `\newpage` .

4.8 Alignement du texte, centrer du texte

Pour faciliter la saisie, on peut utiliser les trois boutons qui se trouvent ici à gauche de la zone de texte, et qui sont représentés ci-dessous.



Contrairement à *Word* ou *LibreOffice*, le réglage par défaut de l’alignement du texte est l’alignement à gauche et à droite, comme dans l’exemple ci-dessous. Au besoin, \LaTeX applique une césure (coupure de mot en fin de ligne), comme pour le mot « anticonstitutionnellement » ci-dessous.

Exemple :

```
Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla bla anticon-
stitutionnellement blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
blabla blabla blabla blabla blabla blabla bla blabla blabla blabla blabla blabla blabla blabla blabla
blabla blabla blabla blabla
```

Les exemples ci-dessous illustrent les trois type d’alignement : à gauche, centré, à droite.

4.8.1 Alignement du texte à gauche

Codage :

```
\begin{flushleft}
Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
```

```
blabla bla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
blabla blabla blabla blabla blabla blabla blabla blabla blabla bla blabla blabla
blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
\end{flushleft}
```

Affichage :

Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla bla blabla
blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
blabla blabla blabla blabla bla blabla blabla blabla blabla blabla blabla blabla blabla
blabla blabla

4.8.2 Texte centré

Codage :

```
\begin{center}
Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
\end{center}
```

Affichage :

Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
blabla blabla blabla blabla

4.8.3 Alignement du texte à droite

Codage :

```
\begin{flushright}
Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
blabla bla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
blabla blabla blabla blabla blabla blabla blabla blabla blabla bla blabla blabla
blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
\end{flushright}
```

Affichage :

Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla bla blabla
blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
blabla blabla blabla blabla bla blabla blabla blabla blabla blabla blabla blabla blabla
blabla blabla

4.10.2 Listes à puces

Une liste à puces avec tirets

Codage :

```
\begin{itemize}
\item ligne 1
\item ligne 2
\item ligne 3
\end{itemize}
```

Affichage :

```
– ligne 1
– ligne 2
– ligne 3
```

Les tirets qui apparaissent ci-dessus dans la liste à puces sont imposés par le module `francais`, parce que c'est la règle en typographie française.

Une 2^e liste à puces avec des gros points noirs

Codage :

```
\begin{itemize}
\item[{$\bullet$}] ligne 1
\item[{$\bullet$}] ligne 2
\item[{$\bullet$}] ligne 3
\end{itemize}
```

Affichage :

```
• ligne 1
• ligne 2
• ligne 3
```

Cette dernière solution pour la liste à puces avec points noirs est fastidieuse à saisir. Pour obtenir que, malgré le module `francais`, l'option par défaut soit la liste à puces à points noirs, on peut insérer la commande ci-dessous dans le préambule.

```
\AtBeginDocument{\renewcommand{\labelitemi}{\textbullet}}
```

4.11 Exposant en mode texte : `fbox` et `boxed`

Deux exemples :

Codage : `n\up{o}` affichage : n^o Codage : `1\up{er}` affichage : 1^{er}

4.12 Encadrer en mode texte ou mathématiques

En mode texte :

codage : `\fbox{blabla}` affichage : blabla

En mode mathématiques :

codage : `\dfrac{2}{3} \approx \boxed{0,67}` affichage : $\frac{2}{3} \approx \boxed{0,67}$

La commande `\boxed` fait partie du module `amsmath`.

4.13 Exercice 2, sur le texte et les listes

Il faut obtenir la présentation ci-dessous.

Devoir de mathématiques n° 1

Nom, prénom, classe :

Exercice 1

La fonction f est définie par $f(x) = (x + 1)e^{0,5x}$

1. Calculer 50 % de 48.
On rappelle que
 - 50 est la moitié de 100 ;
 - 48 est le double de 24.
2. Compléter ici et ici
 - (a) Question 2a
 - (b) Question 2b
3. Quel est l'ensemble $\{1 ; 2 ; 3\} \cup \{3 ; 4 ; 5\}$?

5 Environnement

Il est temps de préciser ce que l'on appelle un « environnement » en LaTeX.

Il s'agit d'une structure du type ci-dessous.

```
\begin{blabla}
  ligne 1
  ligne 2
  etc.
\end{blabla}
```

Plusieurs environnements ont été utilisés précédemment, tels que *Large* pour agrandir un plusieurs lignes de texte, *sffamily* pour afficher plusieurs lignes de texte dans un style sans empattement, *itshape* pour afficher plusieurs lignes de texte en italiques, *center* pour centrer plusieurs lignes de texte, *enumerate* pour obtenir une liste numérotée, *itemize* pour obtenir une liste à puces, etc.

6 Tableaux

6.1 Un premier exemple.

Codage :

```
\begin{tabular}{|l|c|r|p{5cm}|}
  \hline
  Blabla & blabla blabla blabla & blabla blabla &
  bla bla bla bla bla bla bla bla bla bla \\
  \hline
  1&2&3&4\\
  \hline
\end{tabular}
```

Affichage :

Blabla	blabla blabla blabla	blabla blabla	bla bla bla bla bla bla bla bla bla bla bla bla
1	2	3	4

Explications sur le codage

On utilise l'environnement *tabular*.

À propos du codage `|l||c|r|p{5cm}|` sur la première rangée :

- chaque barre verticale indique un trait vertical ;
- les lettres l, c, r, signifient respectivement texte aligné à gauche, centré, aligné à droite ; dans ces trois cas la largeur de colonne s'adapte au contenu ;
- `p{5cm}` signifie un paragraphe de largeur fixée à 5 cm, le texte est aligné à gauche.

Pour le reste du codage

- la commande `\hline` indique un trait horizontal entre deux rangées ;
- le signe `&` sépare deux éléments d'une même rangée ;
- le signe `\\` indique la fin d'une rangée

6.2 Utiliser l'assistant de tableau

Nous allons chercher à obtenir ceci :

x	1	2
$f(x)$		

Veiller à placer le curseur dans une ligne vide, puis dans le menu Assistants, cliquer sur Assistant tableau

Nombre de lignes et de colonnes : 3 colonnes, 2 lignes, voir ci-dessous.

Nb colonnes Nb lignes

Format des colonnes :

- colonne 1 : texte aligné à gauche, voir ci-dessous à gauche ;
- colonne 2 : texte centré, largeur imposée, voir ci-dessous à droite ;
- colonne 3 comme la colonne 2 ;
- bordure gauche et bordure droite : ne rien changer.

Colonnes
Colonne :
Alignement :

Colonnes
Colonne :
Alignement :

En bas à droite, cocher la rubrique Ajouter marge verticale pour chaque ligne

Compléter le tableau en haut à gauche comme ceci :

		c p{}	c p{}
1	$\$x\$$	1	2
2	$\$f(x)\$$		

Cliquer sur

On obtient ceci :

```
% \usepackage{array} is required
\begin{tabular}{|l|>{\centering\arraybackslash}p{•}|>{\centering\arraybackslash}p{•}|}
\hline
\rule[-1ex]{0pt}{2.5ex} $x$ & 1 & 2 \\
\hline
\rule[-1ex]{0pt}{2.5ex} $f(x)$ & • & • \\
\hline
\end{tabular}
```

La première ligne du code ci-dessus signifie qu'il faut appeler le module `array` dans le préambule, il faut donc ajouter au préambule la commande `\usepackage{array}`.

Complétons maintenant le code, en remplaçant les points noirs comme ceci :

```
\begin{tabular}{|l|>{\centering\arraybackslash}p{2cm}|>{\centering\arraybackslash}p{2cm}|}
\hline
\rule[-1ex]{0pt}{2.5ex} $x$ & 1 & 2 \\
\hline
\rule[-1ex]{0pt}{2.5ex} $f(x)$ & & \\
\hline
\end{tabular}
```

On compile, et on obtient l'affichage ci-dessous

x	1	2
$f(x)$		

Dans le codage précédent, remplacer `\rule[-1ex]{0pt}{2.5ex}` par `\rule[-2ex]{0pt}{5ex}`

Puis compiler, et on obtient le tableau annoncé.

6.3 Exercice 3, sur les tableaux

Il faut obtenir le tableau ci-dessous.

k	1	2	3
a			
b			

7 Équation, alignements

Pour la présentation d'un calcul ou de la résolution d'une équation on souhaite aligner les signes $=$ ou \leq ou \iff .

Voici trois exemples. Chacun déterminera ce qui lui convient.

7.1 Environnement `eqnarray*`

Codage

```
\begin{eqnarray*}
(3x+4)(x-2) & = & 3x^2 -6x +4x -8 \\
& = & 3x^2 -2x -8
\end{eqnarray*}
```

Affichage : le résultat est affiché ci-dessous, inséré dans du texte.

Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla

$$\begin{aligned}(3x + 4)(x - 2) &= 3x^2 - 6x + 4x - 8 \\ &= 3x^2 - 2x - 8\end{aligned}$$

Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla

Remarques :

- le codage ressemble à celui d'un tableau ;
- dans l'environnement `eqnarray*` le style mathématiques est automatiquement appliqué, on n'a pas besoin des signes `$`.

7.2 Environnement *align** du module `amsmath`

Codage :

```
\begin{align*}
(3x+4)(x-2) & = & 3x^2 -6x +4x -8 \\
& = & 3x^2 -2x -8
\end{align*}
```

Affichage :

Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla

$$\begin{aligned}(3x + 4)(x - 2) &= & 3x^2 - 6x + 4x - 8 \\ &= & 3x^2 - 2x - 8\end{aligned}$$

Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla

7.3 Environnement *array* du module `array`

À la différence des deux exemples précédents, on inclut l'environnement `array` entre deux dollars.

Codage :

```
$
\begin{array}{rcl}
(3x+4)(x-2) & = & 3x^2 -6x +4x -8 \\
& = & 3x^2 -2x -8
\end{array}
$
```

Affichage :

Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla

$$\begin{aligned}(3x + 4)(x - 2) &= 3x^2 - 6x + 4x - 8 \\ &= 3x^2 - 2x - 8\end{aligned}$$

Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla

7.4 Exercice 4, équation

Il faut obtenir l’affichage ci-dessous.

$$\begin{aligned}
 f(x) = 3 &\iff x^2 - 8x + 7 = 0 \\
 &\iff (x - 4)^2 - 9 = 0 \\
 &\iff (x - 4)^2 = 9 \\
 &\iff x - 4 = 3 \text{ ou } x - 4 = -3 \\
 &\iff \boxed{x = 7 \text{ ou } x = 1}
 \end{aligned}$$

8 Matrices et systèmes

8.1 Matrices

Pour les matrices, on peut utiliser l’environnement *pmatrix* du module `amsmath`.

Codage :

```

$
\begin{pmatrix}
1&2\\
3&4
\end{pmatrix}
$

```

Affichage :

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$$

8.2 Systèmes

Codage :

```

$
\left\{ \begin{array}{rcl}
3x+4y&=&5 \\
-2x+y&=&7
\end{array} \right.
\right.
$

```

Affichage : $\left\{ \begin{array}{l} 3x + 4y = 5 \\ -2x + y = 7 \end{array} \right.$

8.3 Exercice 5, matrice et système

Il faut obtenir ceci : $\begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{pmatrix}$ et ceci : $\begin{cases} 3x + 4y - 5z = 5 \\ -2x + y + z = 7 \\ x - 6y + 9z = -11 \end{cases}$


9 Graphisme en PDFLaTeX

PDFLaTeX est la méthode de compilation qui est utilisée jusqu’ici, il y en a une autre qui sera expliquée au paragraphe 10.

9.1 Inclure un fichier externe png, jpeg ou pdf

9.1.1 La commande `includegraphics` avec fichier png

Dans le menu LaTeX, cliquer sur `includegraphics{file}`

Cliquer sur l'icône 

Sélectionner le fichier image, valider, cliquer sur OK.

Le code qui apparaît est :

```
\includegraphics[scale=1]{image-geogebra-fonction.png}
```

Le paramètre `scale=1` peut être modifié.

Après compilation, on voit l'image qui suit.



Le format de fichier image utilisé ici est le format png.

Avec PDFLaTeX, la méthode de compilation utilisée jusqu'ici, on peut aussi inclure un fichier image au format jpeg ou pdf.

9.1.2 Créer un fichier image avec GeoGebra

Dans GeoGebra, on peut en effet exporter une figure en tant que fichiers image, au format png par exemple.

Pour cela, après avoir tracé une figure dans GeoGebra, on peut faire un cadrage rectangulaire avec

le bouton 

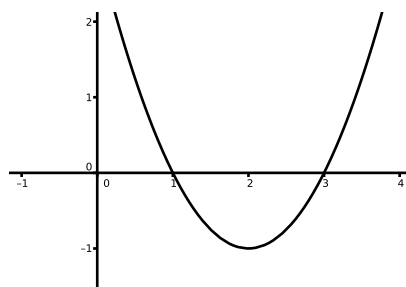
Dans le menu Fichier de GeoGebra, choisir Exporter, puis Graphique en tant qu'image

Cliquer sur Sauvegarder, nommer ce fichier image, cliquer sur Sauvegarder.

9.1.3 Exercice 6, insérer une figure

Ouvrir GeoGebra, tracer la représentation graphique de la fonction définie par $f(x) = (x - 2)^2 - 1$

Il faut ensuite obtenir l’affichage ci-dessous



Ne pas refermer GeoGebra, cette représentation graphique va resservir un peu plus loin.

9.2 Tikz

9.2.1 Description de Tikz

Tikz est un ensemble de macros-commandes permettant de dessiner des figures de géométrie, des quadrillages, des courbes de fonctions, etc.

Ajouter au préambule : `\usepackage{tikz}`

9.2.2 Un exemple de figure avec Tikz

La saisie du codage de Tikz est facilité en utilisant le bouton sur le bord gauche de la fenêtre.

Ci-dessous un exemple de codage et la figure produite plus bas.

Explications

- Pour tracer une figure on utilise l’environnement `tikzpicture` qui permet de définir un rectangle de contour dans lequel la figure est tracée.
- `\draw ... \grid ...` permet de tracer la grille du point de coordonnées $(-1 ; -1)$ au point de coordonnées $(6 ; 3)$;
- `\coordinate ... at ...` permet de définir un point en le nommant, sans le tracer.
- `\draw (A) node[below left] {A} node{\bullet$}`; indique le placement de l’étiquette du point A, puis le marquage du point A par un point noir ;
- `\draw (A) rectangle (C)`; trace le rectangle ABCD
- `\draw (4,1) circle (1)` trace le cercle centré au point de coordonnées $(4 ; 1)$, de rayon 1
- `\draw[...] (0,1) -- (4,1)` trace le segment qui joint les points de coordonnées $(0 ; 1)$ et $(4 ; 1)$, et les options entre crochets indiquent dans l’ordre l’épaisseur, la couleur, et les pointes de flèche.

Codage (commenté) :

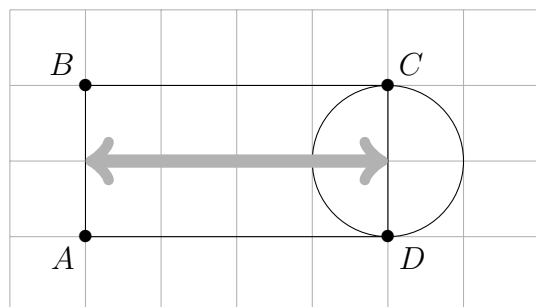
```
\begin{tikzpicture}
\draw[color=gray!60] (-1,-1) grid [step=1] (6,3); % grille en gris
% définitions des quatre points
\coordinate (A) at (0,0) ;
\coordinate (B) at (0,2) ;
\coordinate (C) at (4,2) ;
\coordinate (D) at (4,0) ;
% étiquettes et marques des quatre points
```

```

\draw (A) node[below left] {$A$} node{$\bullet$};
\draw (B) node[above left] {$B$} node{$\bullet$};
\draw (C) node[above right] {$C$} node{$\bullet$};
\draw (D) node[below right] {$D$} node{$\bullet$};
\draw (A) rectangle (C); % tracé du rectangle ABCD
\draw (4,1) circle (1) ; % cercle diamètre (CD)
% une épaisse double flèche grise
\draw[line width=5pt,color=gray!60,<->] (0,1) -- (4,1) ;
\end{tikzpicture}

```

Affichage



9.2.3 Courbe de fonction, codage Tikz avec GeoGebra

Prenons comme exemple la représentation graphique de la fonction définie par $f(x) = (x - 2)^2 - 1$.

Dans GeoGebra, la représentation graphique de cette fonction a été faite au paragraphe 9.1.3.

Dans GeoGebra, tracer un rectangle de sélection à la souris.

Dans le menu Fichier, cliquer sur Exporter, puis sur Graphique vers PGF/TikZ ...

Dans la fenêtre qui s'est ouverte, cliquer sur le bouton Générer le code PGF/TikZ puis sur le bouton

Copier dans Presse-papiers

Revenir dans Texmaker et coller le codage qui ressemble à ce qui est en dessous, les symboles % en moins.

Ces symboles % indiquent toutes les lignes qu'il faut enlever.

Les deuxième et troisième lignes indiquent aussi ce qu'il faut ajouter au préambule :

```

\usepackage{pgf,tikz}
\usetikzlibrary{arrows}

```

Codage PGF/Tikz issu de GeoGebra

```

%\documentclass[10pt]{article}
%\usepackage{pgf,tikz}
%\usetikzlibrary{arrows}
%\pagestyle{empty}
%\begin{document}
\begin{tikzpicture}[line cap=round,line join=round,>=triangle 45,x=1.0cm,y=1.0cm]
\draw[->,color=black] (-1,0.0) -- (4,0.0);
\foreach \x in {,1.0,2.0,3.0}

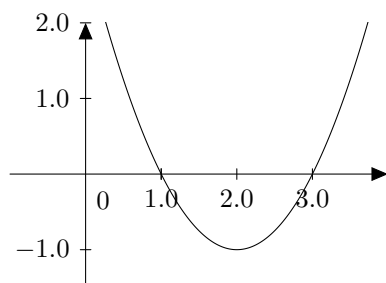
```

```

\draw[shift={(\x,0)},color=black] (0pt,2pt) -- (0pt,-2pt) node[below]
  {\footnotesize $\x$};
\draw[->,color=black] (0.0,-1.5) -- (0.0,2.);
\foreach \y in {-1.0,1.0,2.0}
\draw[shift={(0,\y)},color=black] (2pt,0pt) -- (-2pt,0pt) node[left]
  {\footnotesize $\y$};
\draw[color=black] (0pt,-10pt) node[right] {\footnotesize $0$};
\clip(-1,-1.5) rectangle (4,2);
\draw[smooth,samples=100,domain=-1:4] plot(\x,{((\x)-2.0)^(2.0)-1.0});
\begin{scriptsize}
\draw[color=black] (-0.3,5) node {$f$};
\end{scriptsize}
\end{tikzpicture}
%\end{document}

```

Affichage



9.3 Tableau de variation avec Tikz

Suivre le lien ci-dessous :

<http://math.et.info.free.fr/TikZ/> ou saisir tikz impatient dans Google et cliquer sur la 2e proposition.

On arrive sur le site intitulé *TikZ pour l'impatient* qui contient une documentation, des exemples, des liens, et des utilitaires, comme celui que nous allons utiliser tout de suite.

Cliquer sur l'onglet Tableau de variations

Cliquer à droite de l'indication Lancer l'application ici →

Compléter le tableau comme c'est indiqué ci-dessous.

Tableau de Variations

x	3	-	5	+	9
f'(x)					
f(x)	11		2		8

Aide à l'édition : π ∞ \sqrt{a} $\frac{a}{b}$ \updownarrow \leftrightarrow \overleftrightarrow{a}

<
TikZ
>

Hauteur maximum du tableau :
 2 cm
 2,5 cm
 3 cm
 3,5 cm
 4 cm
 Ne pas afficher la dérivée
 Ne pas afficher la fonction
 Pointillés de rappel

puis cliquer sur le bouton



On voit alors tout le codage du tableau de variations, déjà sélectionné.

Faire un copier-coller dans le fichier, puis compiler.

On voit alors le tableau de variations ci-dessous.

Si on veut pas qu'il soit centré, on enlève les commandes `\begin{center}` et `\end{center}`.

x	3	5	9
$f(x)$	11	2	8

9.4 Arbre pondéré avec Tikz

Nous continuons à utiliser le site évoqué au paragraphe précédent.

<http://math.et.info.free.fr/TikZ/>

Cliquer sur l'onglet Arbres

Cliquer à droite de l'indication Lancer l'application ici →

Considérons que l'arbre à tracer est un arbre pondéré en probabilité.

On voit +

Remplacer Racine par `\Omega`

Cliquer sur +

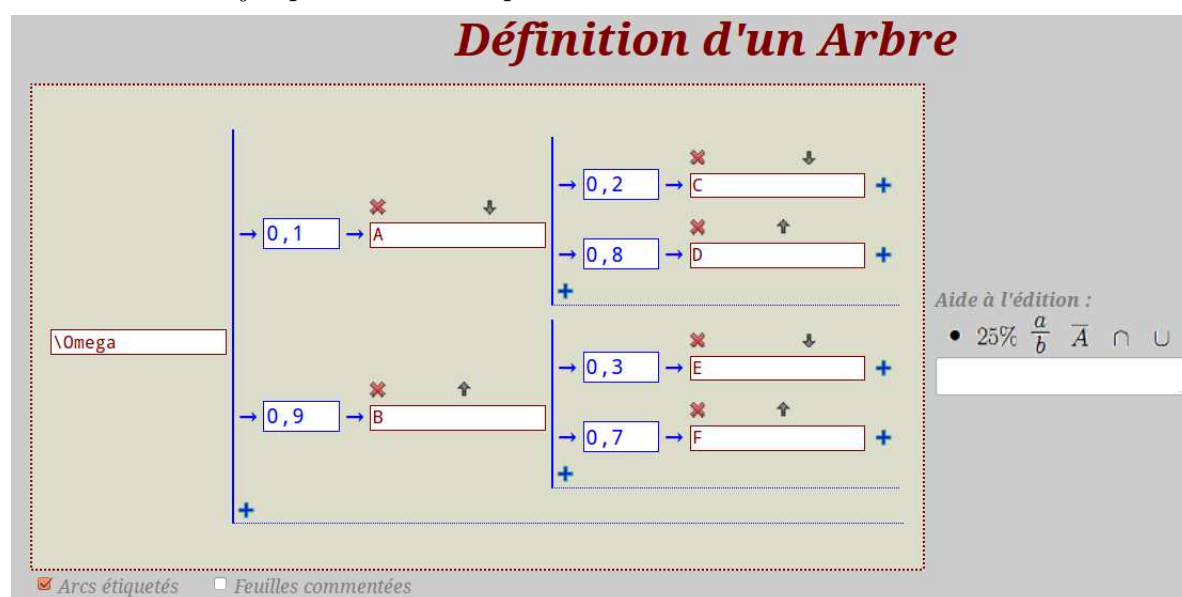
Arc1 est l'étiquette d'une branche, c'est à dire une probabilité et Nœud1 est un événement, les remplacer respectivement par 0,1 et A.


Cliquer sur + en dessous de 0,1

Remplacer Arc2 par 0,9 et Nœud2 par B

Cliquer sur + à droite de A

Poursuivre ainsi jusqu'à obtenir ce qui est en dessous.



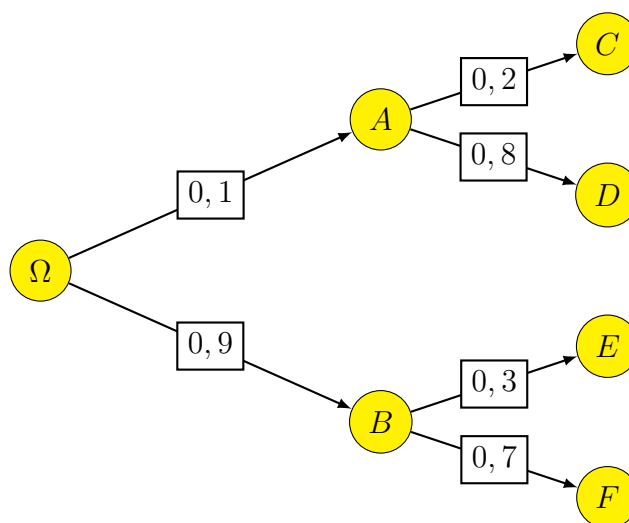
Cliquer sur le bouton 

On voit alors tout le codage de l'arbre, déjà sélectionné.

Faire un copier-coller dans le fichier, puis compiler.

On voit alors l'arbre ci-dessous.

Si on ne veut pas qu'il soit centré, on enlève les commandes `\begin{center}` et `\end{center}`.



9.5 Améliorer les tableaux et arbres obtenus avec TikZ

En s'initiant à TikZ et en utilisant certains modules disponibles sur Internet, on peut modifier ce qui a été obtenu dans les paragraphes précédents.

10 Graphisme en LaTeX+dvips+ps2pdf

10.1 Compilation du fichier par LaTeX+dvips+ps2pdf

La compilation du fichier a été abordée au paragraphe 1.2. Le fichier a été compilé avec PDFLaTeX du menu *Outils*, après quoi on visualise le document pdf avec Voir PDF du menu *Outils*. Tout cela était automatisé avec l'appui sur la touche **F1** que l'on n'a cessé d'utiliser depuis.

Avec PSTricks et tout ce qui est abordé dans la section 10, il faut compiler différemment, il faut enchaîner les commandes LaTeX, Dvi->PS, PS->PDF. Tout cela peut être automatisé, mais il faut modifier un réglage de Texmaker.

Dans le menu *Options*, cliquer sur Configurer Texmaker. Dans la fenêtre qui s'est ouverte, cliquer à gauche sur *Compil rapide*, puis cocher LaTeX + dvips + ps2pdf + View PDF. Cliquer enfin sur OK.

Le problème est que, quand on compile ainsi, on ne peut plus inclure un fichier image au format png, comme nous l'avons fait au paragraphe 9.1.1. Voir le paragraphe ci-dessous.

10.2 Inclure un fichier externe au format eps

Tout ce qui a été indiqué dans le paragraphe 9.1.1 reste valable, mais on ne peut plus faire appel à un fichier png.

Le format de fichier image compatible avec LaTeX + dvips + ps2pdf est le format eps.

La ligne de commande qui avait été saisie était :

```
\includegraphics[scale=1]{image-geogebra-fonction.png}
```

et il faut la remplacer par

```
\includegraphics[scale=1]{image-geogebra-fonction.eps}
```

Attention, la ligne de commande ci-dessus fonctionne parce qu'un fichier nommé `image-geogebra-fonction.eps` existe bel et bien dans le même répertoire que le présent fichier.

10.3 PSTricks et LaTeX+dvips+ps2pdf

10.3.1 Description de PSTricks

PSTricks est un vaste ensemble de macros-commandes permettant de dessiner des figures de géométrie, des quadrillages, des courbes de fonctions, des arbres, des graphes, etc. Cet ensemble est composé de plusieurs modules.

Nous allons pour l'instant ajouter le module `pst-all` qui suffira pour l'instant.

Ajouter `\usepackage{pst-all}` au préambule.

10.3.2 Un exemple de figure avec PSTricks

La saisie du codage de PSTricks est facilitée en utilisant le bouton PS sur le bord gauche de la fenêtre.

Ci-dessous un exemple de codage et la figure produite plus bas.

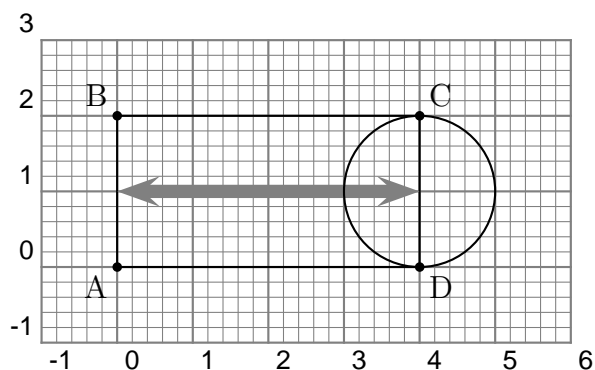
Explications

- `\psset` permet de régler les unités
- Pour tracer une figure on utilise l'environnement `pspicture` qui permet de définir un rectangle de contour dans lequel la figure est tracée. On peut tracer la figure sans cet environnement, et dans ce cas elle recouvre le texte.
- `\psgrid` permet de tracer la grille
- `\psdots` trace les points
- `\uput` permet d'obtenir les noms des points, et le nombre entre crochets est un angle en degrés

Codage :

```
\psset{xunit=1cm,yunit=1cm}
\begin{pspicture}(-1,-1)(6,3)
\psgrid[gridcolor=gray](-1,-1)(6,3)
\psframe(0,0)(4,2)
\psdots(0,0)(0,2)(4,2)(4,0)
\uput[-135](0,0){A} \uput[135](0,2){B}
\uput[45](4,2){C} \uput[-45](4,0){D}
\psline[linewidth=5pt,linecolor=gray]{<->}(0,1)(4,1)
\pscircle(4,1){1}
\end{pspicture}
```

Affichage :



10.3.3 Courbe de fonction, codage PSTricks avec GeoGebra

Prenons comme exemple la représentation graphique de la fonction définie par $f(x) = (x - 2)^2 - 1$. Dans GeoGebra, on crée la courbe de cette fonction, puis on trace un rectangle de sélection à la souris.

Dans le menu Fichier, cliquer sur Exporter, puis sur Graphique vers PSTricks ...

Dans la fenêtre qui s'est ouverte, cliquer sur le bouton Générer le code PSTricks puis sur le bouton Copier dans Presse-papiers

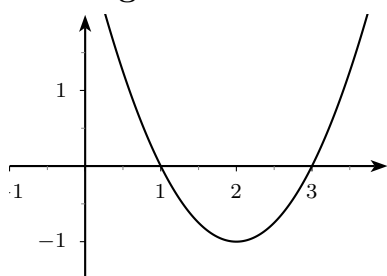
Revenir dans Texmaker et coller le codage qui ressemble à ce qui est en dessous, les symboles % en moins.

Ces symboles % indiquent toutes les lignes qu'il faut enlever.

Codage issu de GeoGebra

```
%\documentclass[10pt]{article}
%\usepackage{pstricks-add}
%\pagestyle{empty}
%\begin{document}
\psset{xunit=1.0cm,yunit=1.0cm,algebraic=true,dimen=middle,%
dotstyle=o,dotsize=3pt 0,linewidth=0.8pt,arrowsize=3pt 2,%
arrowinset=0.25}
\begin{pspicture*}(-1,-1.5)(4,2)
\psaxes[labelFontSize=\scriptstyle,xAxis=true,yAxis=true,
Dx=1,Dy=1,ticks=-2pt 0,subticks=2]{->}(0,0)(-1,-2)(4,2)
\psplot[plotpoints=200]{-1}{4}{(x-2)^(2)-1}
\begin{scriptsize}
\rput[b1](-0.94,8.52){$f$}
\end{scriptsize}
\end{pspicture*}
%\end{document}
```

Affichage



10.4 Tableau de variation avec PST+

Cela nécessite le module `pst-all`.

Il faut aussi

- que le fichier `tabvar.tex` soit présent dans le même répertoire que celui du fichier de travail ;
- ajouter dans le préambule la ligne de commande : `\input tabvar`

Ouvrir maintenant le logiciel PST+

Dans la fenêtre qui s'est ouverte, on voit la liste de ce qui peut être fait avec PST+ : Courbes, Suites, Tableau de variations, etc.

Cliquer sur le bouton Tableau de variations

Compléter alors le tableau comme indiqué ci-dessous.

Tableau						
	1	2	3	4	5	6
1 haut						
1 milieu	x	3		5		9
1 bas						
2 haut		11				8
2 milieu	f(x)		\		/	
2 bas				2		

Cliquer ensuite sur le bouton Aperçu en bas à droite, on visualise alors le résultat en pdf.

Fermer cette dernière fenêtre.

On voit alors, en bas le codage indiqué ci-dessous.

Les lignes commençant par un signe % indiquent des commentaires, permettant d'indiquer à l'utilisateur les modules à utiliser (en fait le module `pst-all` indiqué plus haut suffit). Ces commentaires indiquent aussi qu'il faut ajouter dans le préambule la commande `\input tabvar`.

Sélectionner alors le codage et l'insérer dans le fichier de travail (dans Texmaker), puis compiler, et l'on devrait voir le tableau de variations qui se trouve plus bas (affichage 1).

L'affichage centré du tableau dans l'affichage 1 est dû aux doubles dollars au début et à la fin du codage du tableau (voir ci-dessous). Si l'on souhaite un tableau aligné à gauche, avec moins d'espace au dessus et en dessous, comme dans l'affichage 2, il suffit de ne laisser qu'un dollar au début et un dollar à la fin.

Codage :

```
%\usepackage{pstricks,pst-plot,pst-text,pst-tree,
pst-eps,pst-fill,pst-node,pst-math}
%\input tabvar
$$\tabvar{%
\tx{x}&\tx{3}&&\tx{5}&&\tx{9}\cr
\tx{f(x)}&\txh{11}&\fd&\txb{2}&\fm&\txh{8}\cr
}$$
```

Affichage 1 :

x	3	5	9
$f(x)$	11	2	8

Affichage 2 :

x	3	5	9
$f(x)$	11	2	8

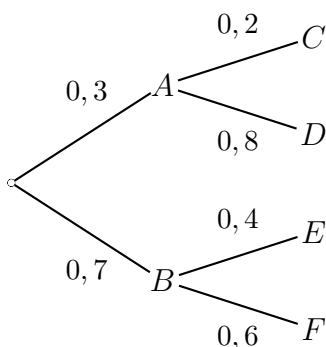
10.5 Arbres pondérés et module pstree

Pour l'exemple ci-dessous, le module `pstree` est nécessaire.

Codage (avec des commentaires insérés)

```
\psset{levelsep=20mm,treesep=10mm}
\pstree[treemode=R,nodesep=0mm]{\Tdot}
% gros point noir à la racine
% nodesep réglé à 0 pour que les deux branches
% issues du gros point soient collées au point
{
\pstree[nodesep=0.5mm]
% nodesep réglé à 0.5 mm pour que les branches
% ne soient pas collées aux lettres A, B, etc.
{\Tr{$A$}\taput{\small $0,3$}}
{
\Tr{$C$}\taput{\small $0,2$}
\Tr{$D$}\tbput{\small $0,8$}
}
\pstree[nodesep=0.5mm]
{\Tr{$B$}\tbput{\small $0,7$}}
{
\Tr{$E$}\taput{\small $0,4$}
\Tr{$F$}\tbput{\small $0,6$}
}
}
```

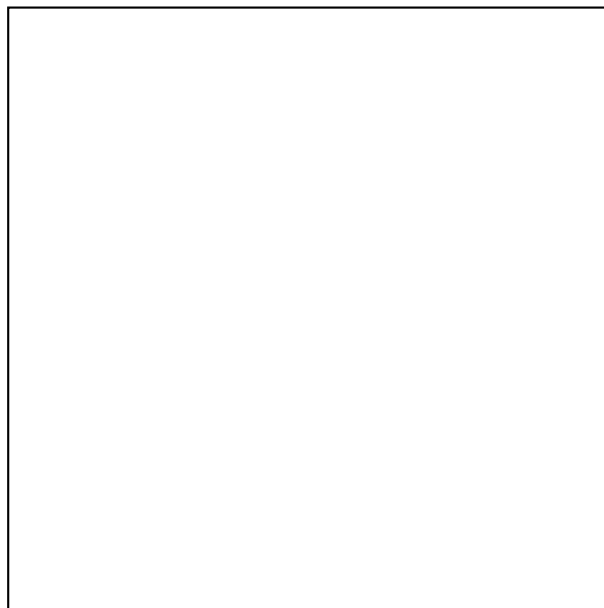
Affichage



Le logiciel PST+, vu au paragraphe 10.4 permet de produire le codage d'un arbre pondéré, mais le résultat n'est pas le même que ci-dessus.

Affichage

Blabla blabla blabla blabla blabla blabla blabla
 blabla blabla blabla blabla blabla blabla blabla
 blabla blabla blabla blabla blabla blabla blabla
 blabla blabla blabla blabla blabla blabla blabla
 blabla blabla blabla blabla blabla blabla blabla
 blabla blabla blabla blabla blabla.



11.2 Disposition en deux blocs et environnement *minipage*

On utilise ci-dessous l'environnement *minipage*, la commande `\hfill`¹ et une figure tracée avec l'environnement *pspicture* du module *psstricks*².

L'environnement *minipage* permet d'insérer du texte et d'autres environnements, comme des environnements de listes, dans un bloc de largeur donnée.

Dans l'exemple ci-dessous, le bloc de texte inséré dans l'environnement *minipage* est à gauche.

Précisons la signification des paramètres `[b] [2cm] [t] {15cm}`

La lettre **b** (pour *bottom*) indique que le bas de la boîte et le bas de la figure sont alignés, sinon, ce ne serait pas le cas.

2 cm est la hauteur du bloc, hauteur égale à la hauteur du carré à droite.

La lettre **t** (pour *top*) indique que, dans la boîte, le texte doit être aligné verticalement vers le haut, sinon, il serait centré verticalement dans la boîte.

En dessous de l'environnement *minipage* se trouve la commande `\hfill` qui permet de laisser un espace entre le texte et la figure, tout en permettant que l'ensemble occupe toute la largeur d'une marge à l'autre et soit donc aligné à gauche et à droite.

Codage

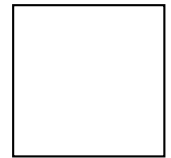
```
\begin{minipage}[b] [2cm] [t] {15cm}
  Blabla ... blabla blabla.
\end{minipage}
\hfill
\psset{xunit=1cm,yunit=1cm}
  \begin{pspicture}(0,0)(2,2)
    \psframe(0,0)(2,2)
  \end{pspicture}
```

1. Voir paragraphe 4.6.2

2. Voir paragraphe 10.3 sur PSTricks.

Affichage

Blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
 blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla blabla
 blabla blabla blabla blabla blabla blabla blabla.



12 Diverses choses

12.1 Structure du document : sections, sous sections, etc.

Comme on peut le lire dans ce document et dans la table des matières, le présent document est structuré en sections numérotées 1, 2, 3, etc. des sous sections numérotées par exemple 2.1, 2.2 etc., des sous-sections numérotées par exemples 2.1.1, 2.1.2, etc.

Ces numérotations ne sont pas faites avec des listes numérotées mais avec les commandes suivantes :
`\section` `\subsection` `\subsubsection`

Par exemple le présent paragraphe est une sous section et la commande du fichier source est indiquée ci-dessous.

```
\subsection{Structure du document : sections, sous sections, etc.}
```

Ce système permet de construire des documents longs, avec des possibilités de renvois à des numéros de paragraphes.

Il n'a pas paru souhaitable pour un stage d'initiation d'aller plus loin sur le sujet. Pour apprendre des précisions à ce sujet, voir la section 13 en fin de document.

12.2 Format d'encodage des fichiers : UTF-8, ISO-8859-15 – Logiciel Gedit

Le fichier dans lequel nous travaillons est encodé en UTF-8. C'est un format de texte permettant d'utiliser beaucoup d'alphabets (français, espagnol, tchèque, suédois etc.), et cet encodage est désormais conseillé. L'encodage en UTF-8 est utilisé par défaut dans Texmaker et dans beaucoup de logiciels désormais.

En ouvrant certains fichiers \LaTeX , on peut voir le message



On clique sur OK, et on travaille alors dans l'encodage ISO-8859-15, qui est un encodage fait pour les langues européennes, mais plus ancien, qui accepte moins de caractères que l'UTF-8.

Connaître l'encodage d'un fichier ou le modifier.

Un moyen possible est d'utiliser le logiciel Gedit qui est un éditeur de texte. On ouvre le fichier dans ce logiciel, puis, dans le menu Fichier, on choisit Enregistrer sous, et dans la fenêtre qui s'est ouverte, on voit le codage des caractères en bas à gauche.

Le fichier d'installation du logiciel Gedit est dans `partage prof` dans le répertoire `fichiers-installation-latex-et-autres`.

12.3 Utilisation de LaTeX dans d'autres logiciels

Le codage en \LaTeX est utilisé dans certains logiciels : dans GeoGebra, dans spip (avec lequel est fait le site de mathématiques du lycée), dans le logiciel de messagerie Thunderbird (en ajoutant le greffon Latex It!).

12.4 Quelques macrocommandes

Pour obtenir le symbole \mathbb{R} de l'ensemble des réels

Ajouter dans le préambule la ligne ci-dessous.

```
\newcommand{\R}{\text{rm}{I\kern-0.21emR}}
```

et, dans le fichier de travail, il suffit de saisir $\backslash R\{}$ pour obtenir le symbole \mathbb{R} .

Pour obtenir la lettre e de l'exponentielle en mode texte dans un environnement mathématiques

Ce qui suit nécessite module `amsmath`, à cause de la commande $\backslash text\{}$ qui fait partie de ce module.

Ajouter dans le préambule la ligne ci-dessous.

```
\newcommand{\e}{\text{e}}
```

Par exemple, on saisit $\backslash e^x$ et on obtient : e^x

12.5 Un préambule commun au lycée ?

Au cours de ce stage le contenu du préambule d'un fichier \LaTeX a été abordé à plusieurs reprises, tout d'abord au paragraphe 1.1, puis tout au long du stage, chaque fois qu'un module devait être ajouté au préambule.

Finalement le préambule est un peu une boîte à outils : on saisit la liste des outils dont on a besoin, par exemple `amsmath` pour les écritures mathématiques, `pst-all` pour PSTricks, etc.

Dans certaines universités sont adoptés des préambules communs pour faciliter les échanges de documents, et il pourrait être intéressant de convenir ensemble au lycée Bellepierre d'un préambule commun.

13 Après le stage

13.1 Livres sur LaTeX

LaTeX Apprentissage, guide et référence, 2^e édition, B. DESGRAUPES, Vuibert 2003

LaTeX pour l' impatient : lien vers une page web consacrée à cet ouvrage ci-dessous.

http://www.h-k.fr/liens/tp/latex_pour_l_impatient.html

13.2 Documentation sur Internet

13.2.1 Le site de documentation de LaTeX et Texmaker

<http://www.xm1math.net/doculatem/index.html>

Ce site est d'une présentation très claire, il convient bien, à mon avis à un débutant, il contient beaucoup d'exemples, que l'on peut récupérer par copier-coller ou par des fichiers *.tex que l'on peut enregistrer.

13.2.2 Le fichier pdf "LatexPourProfMaths" par Arnaud GAZAGNES

Ce fichier est téléchargeable sur cette page web :

<http://math.univ-lyon1.fr/irem/spip.php?article340>

Pour un professeur de mathématiques, c'est la meilleure documentation que je connaisse, très complète, rédigée comme un mode d'emploi efficace et pratique, avec beaucoup d'exemples.

Le lien ci-dessus est d'ailleurs donné sur les site de documentation de Texmaker sur LaTeX, dans la rubrique "Autres ressources".

13.2.3 Documentation sur TikZ

Aller sur le site indiqué ci-dessous. On peut y télécharger un fichier pdf, version intégrale du livre *TikZ pour l' impatient*. C'est une bonne documentation pour débiter.

Ce site propose aussi des pages d'exemples, et des pages qui génèrent le code pour tracer des tableaux de variations et des arbres, comme cela a été fait au paragraphes 9.3 et 9.4.

<http://math.et.info.free.fr/TikZ/>

On trouve le manuel de référence de TikZ intitulé *TikZ & PGF Manual for Version 2.00* à l'adresse ci-dessous.

<http://www.texample.net/media/pgf/builds/pgfmanualCVS2012-11-04.pdf>

13.2.4 Documentation sur PSTricks

Un document d'initiation

LaTeX et PSTricks par M. VAN GARREL et A. DANALET de l'école polytechnique de Lausanne.

Un bon document en français pour s'initier à PSTricks.

lien vers ce document pdf

Le document de référence

Le document de référence, en anglais, est indiqué ci-dessous ; son auteur est le créateur de PSTricks.

PSTricks : PostScript macros for Generic TeX. User's Guide par Timothy VAN ZANDT

Lien vers le site de PSTricks : <http://tug.org/PSTricks/main.cgi/>

On accède au document de référence en cliquant « Documentation » à gauche, et en choisissant Complete PSTricks user guide : as one PDF (by Denis Girou ...)

C'est une compilation de documentations de différents modules de PSTricks, dont `pstree`.

13.3 Installations sous Windows

Les étapes essentielles d'installation sont d'installer d'abord le système \LaTeX puis d'installer un logiciel éditeur de texte et compilateur.

Il existe différentes distributions de \LaTeX , comme TeXLive ou MiKTeX, ici c'est MiKTeX qui est proposé.

Le logiciel éditeur de texte et compilateur proposé ci-dessous sera Texmaker, le logiciel utilisé pendant le stage.

13.3.1 Installation MiKTeX avec proTeXt

Créer d'abord un répertoire sur le bureau nommé par exemple `extraction-protex`.

Ouvrir le fichier `ProTeXt-3.1.3-060313`

Choisir comme répertoire de destination le répertoire créé ci-dessus (nommé `extraction-protex` ici).

Utiliser le bouton (parcourir).

Cliquer sur le bouton .

Dans le répertoire `extraction-protex`, lancer `setup`.

Une petite fenêtre nommée `proTeXt` s'ouvre. À côté de la mention `MiKTeX` cliquer sur le bouton

Dans la fenêtre `Copying conditions` cocher `I accept ...`

Bouton

Fenêtre `Package set`

Choisir `Basic MiKTeX` (5 min, mais installation partielle, mais les futurs téléchargements de modules complémentaires se téléchargent très rapidement) ou `Complete MiKTeX` (10 à 20 minutes, mais installation complète)

Cocher (laisser) `Anyone who use ...` ou `Only for ...`

Fenêtre `Settings`

Preferred paper : une liste déroulante propose `A4` ou `Letter`

Install missing package on the fly : une liste déroulante propose `Ask me first` ou `Yes` ou `No`

Fenêtre `Information` : récapitule les options choisies

Bouton

10 à 20 min plus tard

Fenêtre Executing The main task is being executed

Fenêtre Completing the MiKTeX Setup Wizard

Bouton

Fermer la petite fenêtre de départ proTeXt en cliquant sur sauf si l'on veut installer TeXstudio au lieu de Texmaker.

13.3.2 Installation Texmaker

Ouvrir fichier `texmakerwin32_install`

1^{re} fenêtre : cliquer sur le bouton en bas à droite

2^e fenêtre : cliquer sur le bouton en bas à droite

10 secondes

3^e fenêtre : cliquer sur le bouton en bas à droite

13.3.3 Installation PST plus

Ouvrir fichier `pstpluswin32_install`

1^{re} fenêtre : cliquer sur le bouton en bas à droite

2^e fenêtre : cliquer sur le bouton en bas à droite

5 secondes

3^e fenêtre : cliquer sur le bouton en bas à droite

13.3.4 Installation Gedit

Ouvrir fichier `gedit-setup-2.30.1-1`

1^{re} fenêtre : cliquer sur le bouton en bas

2^e fenêtre : cocher `I accept ...`, puis cliquer bouton en bas

3^e fenêtre Select Destination Location : cliquer sur le bouton en bas

4^e fenêtre Select Start Menu Folder : cliquer sur le bouton en bas

5^e fenêtre Select Additionnal Tasks : cocher option choisie, puis cliquer sur le bouton en bas

6^e fenêtre Ready to Install : cliquer sur le bouton en bas

5 secondes

7^e fenêtre Completing ... : cliquer sur le bouton en bas

5 secondes

Cliquer sur le bouton en bas à droite