

Introduction à \LaTeX

Table des matières

1	Généralités	2
1.1	Qu'est-ce que \LaTeX et pourquoi l'utiliser	2
1.2	La structure de \TeX et \LaTeX	2
1.3	Les bases du fonctionnement de \TeX et \LaTeX	3
2	Choisir et installer le matériel	3
2.1	\MiKTeX	4
2.2	L'éditeur de texte	4
2.3	Outils divers	4
3	Créer un document type	4
3.1	Ouvrir et sauver le fichier source \TeX	4
3.2	Structurer le fichier source	5
3.3	Déclarer les options du document et les package utilisés	5
3.4	Définir la mise en page globale du document	6
3.5	Structurer le document	7
3.6	Définir ses propres environnements	8
3.7	Définir ses propres macros	8
4	Utiliser le document <code>documentTM.tex</code>	9
4.1	Taper du texte	9
4.2	Compiler	10
4.3	Visualiser le résultat	10
4.4	Transformer le document dans le format désiré	10
5	Personnaliser le document <code>documentTM.tex</code>	10
5.1	Générer le titre	10
5.2	Modifier les entêtes et les pieds de pages	11
5.3	Créer la table des matières	11
5.4	Changer le style du texte	12
5.5	Créer une liste	12
5.6	Insérer du texte mathématique	12
5.7	Créer un tableau	13
5.8	Insérer une figure	14
5.9	Créer une note de bas de page	15
5.10	Générer une référence	15
5.11	Générer la bibliographie	16
5.12	Insérer les annexes	16
5.13	Pour aller plus loin	16
	Références	17

1 Généralités

1.1 Qu'est-ce que L^AT_EX et pourquoi l'utiliser

Le traitement de texte L^AT_EX n'est pas un WYSIWYG (abréviation de What You See Is What You Get) mais un langage de programmation qui utilise des balises et des macros.

Originellement, KNUTH a développé le Plain-T_EX¹ pour écrire des textes scientifiques et plus particulièrement mathématiques. Bien que cette version permet de tout faire, elle est souvent lourde et malaisée. C'est pourquoi LAMPORT a écrit une librairie de macros qui sont basées sur Plain-T_EX. Cette nouvelle version est appelée² L^AT_EX. D'autres apports ont permis de créer des versions encore plus élaborées, jusqu'à celle appelée³ L^AT_EX 2_ε et qui est utilisée aujourd'hui.

On aura compris que L^AT_EX est un traitement de texte destiné à écrire des documents à contenus scientifiques. Les avantages principaux sont les suivants :

- Il est très facile d'insérer des symboles et des formules mathématiques dans le texte. Par exemple

◇ © ✓ ‡ ¶ ⇔ ∘ χ

- Il est très facile de créer des raccourcis personnels (macros) pour obtenir des structures complexes qui se répètent. Par exemple, la matrice ci-contre s'obtient avec la commande

| \FuMa f_{xna}

si on a défini ce macro précédemment.

$$\begin{pmatrix} \frac{\partial f_1}{\partial x_1}(a) & \cdots & \frac{\partial f_1}{\partial x_n}(a) \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_1}(a) & \cdots & \frac{\partial f_n}{\partial x_n}(a) \end{pmatrix}$$

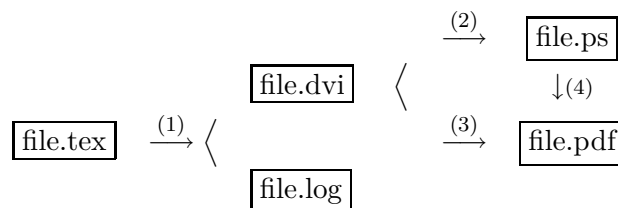
- Il est possible d'écrire de très longs textes (par exemple des livres de plusieurs centaines de pages) avec une stabilité garantie et des fichiers très légers.
- Il n'est pas nécessaire de se préoccuper des aspects esthétiques.
- Il est très facile de maîtriser et de changer l'aspect global du texte (taille de la page, grandeur de la police, écart entre les lignes et les paragraphes) tout en gardant une unité.
- Les numérotations sont automatiques et peuvent être utilisées en référence sans que des changements ne les affectent.
- Il est très facile de créer une table des matières, une bibliographie et un index.

L^AT_EX a également deux inconvénients majeurs :

- La création de tableaux est malaisée.
- L'insertion de figure est automatique et des souhaits précis peuvent être difficiles à réaliser.

1.2 La structure de T_EX et L^AT_EX

Vu que L^AT_EX est un langage de programmation, il faut écrire un code et ensuite le faire interpréter par l'ordinateur et peut-être, dans une dernière phase, transformer le résultat dans le format désiré. Schématiquement, on a



¹Voir [3]

²Voir [4]

³Voir [2]

Le fichier source `file.tex` est un fichier ASCII. Il s'écrit à l'aide d'un éditeur de texte et contient le code. Sa taille est de quelques Ko et peut ainsi facilement être transmis (par exemple par e-mail).

Le passage (1) du fichier source au fichier `file.dvi` (device-independent) repose sur le programme \LaTeX à proprement parler. L'exécution de ce procédé, communément appelé **compiler** ou **typeset**, ne dépend pas de l'implémentation. Ce procédé est accompagné dans le `file.log` d'un protocole qui liste les éventuels problèmes rencontrés lors du typeset.

A partir du fichier `file.dvi`, on peut créer un fichier postscript `file.ps` (étape (2), langage des imprimantes) ou un fichier pdf `file.pdf` (étape (3)). Finalement, il est également possible de convertir un fichier postscript en fichier pdf : c'est l'étape (4). Pour plus de détails, voir la section 2.

1.3 Les bases du fonctionnement de \TeX et \LaTeX

Comme dans tout langage de programmation, certains symboles ont une signification particulière. Pour \LaTeX ce sont les suivants :

<code>%</code>	Le reste de la ligne est ignoré (utile pour écrire des commentaires)
<code>\</code>	Utilisé pour indiquer le début d'un macro (Exemple : <code>\FuMa</code>)
<code>{ }</code>	<code>{ }</code> et <code>}</code> sont des délimiteurs logiques
<code>\$</code>	Utilisé pour commencer et terminer un texte mathématique
<code>^</code>	Utilisé pour créer un exposant
<code>_</code>	Utilisé pour créer un indice
<code>&</code>	Séparateur de colonne dans les tableaux par exemple

Ensuite, les spécifications de ce langage sont les suivantes :

- (1) \LaTeX est **orienté en lignes** :
 - (a) Après un `%` le reste de la ligne est ignoré.
 - (b) Les lignes adjacentes sont interprétées comme *une* seule longue ligne.
 - (c) Un ou plusieurs espaces entre deux mots sont interprétés comme *un seul* espace.
 - (d) Une ligne ou plusieurs lignes vides sont interprétées comme *un seul* espace entre deux paragraphes.
- (2) \LaTeX utilise des **macros** et des **balises**. En les utilisant, on peut générer des chapitres, des sections, des paragraphes, des remarques, etc qui seront automatiquement numérotés. Comme déjà dit, on peut faire référence à ces numéros et la référence reste correcte même si la numérotation change.
- (3) \LaTeX distingue deux modes d'écriture : le **Text-Mode** et le **Math-Mode** : En Math-Mode \LaTeX décide automatiquement des aspects typographiques pour un rendu optimal.
Syntaxe : Texte `$`formule mathématique`$` Texte
- (4) En complément du Math-Mode normal, il existe le **Display-Mode** qui permet d'écrire des formules mathématiques compliquées et de les mettre en évidence sur une seule ligne, centrée.
Syntaxe : Texte `$$`formule`$$` Texte *ou mieux encore* Texte `\[`formule`\]` Texte

Vous trouverez une introduction beaucoup plus détaillée à \LaTeX dans [2] et [6].

2 Choisir et installer le matériel

La gratuité de \LaTeX constitue un avantage non négligeable. Voici une manière simple d'installer ce système sur

- WINDOWS : Télécharger⁴ proTeXt à l'adresse <http://www.tug.org/protext/>. Ouvrir le fichier interactif `protext-fr.pdf` et suivre les instructions à l'écran.

⁴Choisir le lien `download the self-extracting protext.exe file`.

- MACOS : Télécharger MacTeX à l'adresse <http://www.tug.org/mactex/>.
- LINUX : Visiter par exemple le lien <http://www.ac-poitiers.fr/math/tex/>.

Les différentes explications qui suivent concernent la distribution basée sur WINDOWS.

2.1 MIKTEX

MIKTEX est le principal programme qui permet de compiler⁵. Lors de son installation, il est conseillé de choisir la version *large* ou *complete*. Il est à placer sur votre disque dur (par exemple dans le dossier Programmes Files) et les bibliothèques sous C: \texmf\tex\latex.

2.2 L'éditeur de texte

Les éditeurs de texte sont nombreux. Ils permettent d'écrire le fichier source `file.tex`. Son choix dépend des attentes et des goûts de l'utilisateur et de sa familiarité avec le langage L^AT_EX. La distribution PROTEXT installe l'éditeur TEXNICCENTER. Cet éditeur est tout à fait convenable bien qu'il ne supporte pas l'encodage UTF-8. En voici d'autres qu'il est possible d'installer :

- TEXMAKER : freeware (gratuit) très convivial (idéal pour débiter) téléchargeable à l'adresse <http://www.xm1math.net/texmaker/>. Il est à placer sur votre disque dur sous C: \Programmes Files\Texmaker. Ce freeware supporte l'encodage UTF-8. C'est cet éditeur qui est installé à Gambach.
- WINEDT : shareware (environ 50 Euros) convivial et puissant qui est téléchargeable à l'adresse <http://www.ac-poitiers.fr/math/tex/>. Il est à placer sur votre disque dur sous C: \Programmes Files\WinEdt. Ce freeware ne supporte pas l'encodage UTF-8.

2.3 Outils divers

Il existe de multiples autres outils qui peuvent être utiles. GSVIEW et GPL GSSCRIPT sont des freeware indispensables pour visualiser⁶ des fichiers postscript et les convertir⁷ en pdf. Ces deux outils font partie de la distribution `protext`.

On trouve une multitude d'autres outils sur le web. On citera notamment :

- <http://www.ams.org/tex/> : téléchargement de package divers (AMS L^AT_EX Author Package for author package documentation)
- <http://www.ctan.org/> : nombreuses explications sur l'utilisation de L^AT_EX et de leurs outils
- <http://www.xm1math.net/> : téléchargement d'outils divers comme PSTPLUS (dessin, etc)

3 Créer un document type

Cette section présente les étapes pour créer son propre document en partant de rien. Il est possible de sauter cette section et de télécharger le document `documentTM.tex` pour directement taper son texte.

3.1 Ouvrir et sauver le fichier source T_EX

Lancer l'éditeur c'est-à-dire le programme TEXNICCENTER ou TEXMAKER. Ouvrir un nouveau document et l'enregistrer dans son dossier personnel avec l'extension ".tex" (c'est-à-dire enregistrer le document ainsi : `file.tex`)

⁵C'est-à-dire d'effectuer le passage (1) présenté au paragraphe 1.2.

⁶C'est-à-dire voir le résultat de l'étape (2) présenté au paragraphe 1.2.

⁷C'est-à-dire effectuer le passage (4) présenté au paragraphe 1.2.

3.2 Structurer le fichier source

Dans un premier temps, taper les commandes obligatoires suivantes :

```
\documentclass{report}
\begin{document}

\end{document}
```

On peut également remplacer `report` par la classe `article` ou encore `book` selon la longueur du travail.

Pour des documents élaborés, il est préférable de structurer le fichier source avec des commentaires (précédés de `%`) pour pouvoir s'y retrouver. Par exemple ainsi :

```
%%%%
% Class
%%

\documentclass{report}

%%
% Package
%%

%%
% Mise en page
%%

%%
% Debut du document
%%

\begin{document}

%%
% Titre
%%

%%
% Tables des matières
%%

%%
% Texte
%%

%%
% Bibliographie
%%

%%
% Fin du document
%%

\end{document}
```

3.3 Déclarer les options du document et les package utilisés

On peut dans un premier temps choisir les options du document (taille du papier, taille de la police, recto-verso, etc) à l'aide de la commande `\documentclass[option1,option2,...]{report}`. Voir [2]

pour une liste complète des options. Par exemple

```
\documentclass[a4paper,french,12pt,fleqn]{report}
```

Déclarer ensuite à l'aide de la commande `\usepackage[option1,option2,...]{nomdupackage}` les packages utilisés. Ceci doit se faire **avant** la commande `\begin{document}`! Par exemple :

```
%%%%
% Packages
%%%%

%Langue
\usepackage[latin1]{inputenc} % gestion des accents
\usepackage[T1]{fontenc} % fonts
\usepackage[french]{babel} % choix de la langue (german ou english)
\usepackage{lmodern} % français moderne

%AMS
\usepackage{amsmath} % macros ams
\usepackage{amsfonts} % fonts ams
\usepackage{amssymb} % symboles ams

%Mise en page
\usepackage[top=2.5cm,bottom=2.5cm,left=2.5cm,right=2.5cm]{geometry} % Mise en page
\usepackage{fancyhdr} % Entêtes et pieds de page

%Définition de macros (environnements)
\usepackage{amsthm}

%Figure
\usepackage{graphicx}
\usepackage{wrapfig}

%Symboles mathématiques
\usepackage{latexsym}
```

Remarquez qu'en général, l'utilisation d'un macro requiert la déclaration d'un package... Il suffira donc d'ajouter les packages utiles à la liste qui précède.

3.4 Définir la mise en page globale du document

La déclaration du package `\usepackage[top=2cm,bottom=2cm,left=2cm,right=2cm]{geometry}` puis de la commande `\geometry{a4paper}` permet de définir le format et les marges de chaque page de texte.

La mise en page globale du document s'obtient avec la commande `\pagestyle{style}` permet de fixer le style de page (numérotation, etc). Les styles possibles sont `empty` (vide), `plain` (numéro de page centré en pied de page) ou encore `fancy`. Ce dernier style requiert le package `fancyhdr` et permet de créer une entête et des bas de page personnalisés.

On a ainsi par exemple

```

%%%%
% Packages
%%%%

%Mise en page
\usepackage[top=2.5cm,bottom=2.5cm,left=2.5cm,right=2.5cm]{geometry} % Mise en page
\usepackage{fancyhdr} % Entêtes et pieds de page

%%%%
% Mise en page (A ADAPTER)
%%%%

% marges
\geometry{a4paper}

%Entete et bas de page
\renewcommand{\headrulewidth}{.4pt} % largeur du filet de l'entête (Opt=pas de filet)
\fancyhead[R]{} % entête droite
\fancyhead[L]{} % entête gauche
\fancyhead[C]{} % entête centrée
\renewcommand{\footrulewidth}{.4pt} % largeur du filet du pied de page (Opt=pas de filet)
\fancyfoot[R]{} % pied de page à droite
\fancyfoot[L]{} % pied de page à gauche
\fancyfoot[C]{} % pied de page au centre

%Paragraphe
\setlength{\parindent}{0pt} % indentation du paragraphe
\setlength{\parskip}{5pt plus1pt minus1pt} %espace entre chaque paragraphe

%Numerotation
\setcounter{secnumdepth}{2} % profondeur de la numérotation

%%%%
% Debut du document
%%%%

\begin{document}

%%%%
% Style de page (A ADAPTER)
%%%%

\pagestyle{fancy}

```

Le numéro de la page s'obtient avec la commande `\thepage`. Les commandes `\leftmark` et `\rightmark` contiennent le nom du chapitre respectivement de la section. Pour des mises en page plus élaborées, voir les références.

3.5 Structurer le document

La commande `\chapter{}` produit un chapitre numéroté ayant pour titre ce qui figure entre les accolades.

Les sections, sous-sections, ... s'obtiennent avec les commandes `\section`, `\subsection`, etc et se numérotent *automatiquement*.

Pour obtenir un chapitre, une section, etc non numéroté, ajouter le symbole `*` entre la commande et l'accolade ouvrante. Dans ce cas, le chapitre ou la section n'apparaît pas dans la table des matières. Pour y remédier, on peut utiliser la commande `\addcontentsline{toc}{Niveau d'insertion dans la table}{Nom apparaissant dans la table}`.

Par exemple :

```

\chapter{Premier chapitre}

\section{Première section du premier chapitre}

\section{Deuxième section du premier chapitre}

\section*{Attention: pas numérotée}
\addcontentsline{toc}{section}{Attention}

\chapter{Deuxième chapitre}

```

Finalement, les commandes `\linebreak`, `\pagebreak`, `\newpage` permettent de réaliser les sauts suggéré par la commande.

3.6 Définir ses propres environnements

On peut également définir et employer des environnements qui correspondent un peu près aux styles des WYSIWYG. Par exemple, le package `{amsthm}` permet de définir de nouveaux environnements à l'aide de la commande `\newtheorem{env-name}{label-text}[counter]` dont les différentes options se gèrent avec `\theoremstyle{style}`. Il faudra alors taper `\begin{env-name} ... \end{env-name}` dans le texte pour obtenir le `env-name` (par exemple une définition) désiré. Ce dernier sera alors *automatiquement* numéroté!

```

%%%%
%% Environnements
%%%%

{\theoremstyle{plain}
\newtheorem{thm}{Théorème}[section]
}

{\theoremstyle{definition}
\newtheorem{defn}[thm]{Définition}
}

%%%%
%% Debut du document
%%%%

\begin{document}

%%%%
%% Texte
%%%%
La définition d'une fonction est la suivante:
\begin{defn}
Une fonction  $f:A\to B$  est une correspondance qui à chaque élément de  $A$  associe exactement
un élément de  $B$ .
\end{defn}

```

Voir [2, pages 251-55] pour une liste complète des options et des exemples.

3.7 Définir ses propres macros

Un des avantages de L^AT_EX 2_ε se manifeste dans cette partie.

On peut définir des macros à l'aide des commandes `\def` et `\newcommand`. La différence entre ces deux commandes réside dans le fait que `\newcommand` peut avoir un ou plusieurs paramètres. Par exemple, avec

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Macros personnels (A ADAPTER)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

\def\R{\mathbb{R}}

\newcommand{\FuMa}[4]{\left(\begin{array}{ccc}%
\frac{\partial #1_1}{\partial #2_1}(\#4) & \cdots & \frac{\partial #1_1}{\partial #2_n}(\#4)\cr
\vdots & & \vdots\cr
\frac{\partial #1_{\#3}}{\partial #2_1}(\#4) & \cdots & \frac{\partial #1_{\#3}}{\partial #2_{\#3}}(\#4)
\end{array}\right)}

```

En tapant `\R` dans le texte, on obtient \mathbb{R} . En tapant `\FuMa{f}{x}{n}{a}`, on obtient la matrice de la page 2.

4 Utiliser le document `documentTM.tex`

Cette section présente les manipulations de base pour utiliser le document `documentTM.tex`.

4.1 Taper du texte

Taper du texte dans la partie correspondante. Par exemple

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Texte (A ADAPTER)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Voici mes première ligne en \LaTeX

```

La commande `\LaTeX` est un macro qui génère automatiquement le texte " \LaTeX ". D'une manière générale, les balises `\begin{env-name} ... \end{env-name}` permettent de faire ce que suggère l'`env-name` pour le texte qui se situe entre les deux balises. Par exemple

```

\begin{center}
Ce texte sera centré.
\end{center}

```


Pour créer ses propres environnements, voir le paragraphe 3.6

Conseil

Chaque fois que vous ouvrez un délimiteur `{` ou un environnement `\begin{env-name}`, refermez-le immédiatement avec `}` ou `\end{env-name}` et tapez ensuite votre texte entre.

Pour créer des chapitres, des sections, des paragraphes, insérer des sauts de lignes, de pages, etc, se référer au paragraphe 3.5.

4.2 Compiler

Pour compiler le texte, cliquez simplement sur le bouton  de votre éditeur.

Conseil
Il faut très souvent compiler !

Il y a deux avantages majeurs :

- Votre fichier est automatiquement sauvé chaque fois que vous compilez.
- On repère plus facilement les éventuelles erreurs.

Les éventuelles erreurs de syntaxe sont mentionnées dans le fichier `file.log`. En principe, le numéro de la ligne où une faute a été repérée est indiqué. Il s'agit très souvent de l'oubli de délimiteur ou de l'oubli de `$` en Math-Mode. Les corriger puis compiler à nouveau.

4.3 Visualiser le résultat

Le résultat se visionne avec le YAP dans le `file.dvi`. Une commande dans l'éditeur (*voir Dvi* pour TEXMAKER) permet d'ouvrir ce petit logiciel et de visualiser le résultat.

4.4 Transformer le document dans le format désiré

Suivant votre éditeur, vous pouvez transformer votre `file.dvi` dans un format plus agréable. Plus précisément :

- **dvi to ps**
Le format ps (postscript) est le langage d'impression. Cette transformation se fait avec le bouton correspondant dans votre éditeur et le résultat se visualise avec GHOSTVIEW.
- **ps to pdf**
Le format pdf est un format passe-partout. Cette transformation se fait avec le bouton correspondant dans votre éditeur de texte, avec GHOSTVIEW ou avec ACROBAT DISTILLER. Le résultat se visualise avec ACROBAT READER.
- **dvi to pdf**
Cette transformation se fait avec le bouton correspondant dans votre éditeur de texte mais est à éviter si votre document contient des images au format eps (incompatibilité).

Remarquez que vous pouvez configurer ces différents éléments dans votre éditeur de texte. Pour TEXMAKER, la configuration s'effectue à partir du menu *Options*.

5 Personnaliser le document `documentTM.tex`

Voici quelques indications sommaires qui vous permettent de personnaliser le document `documentTM.tex`.

5.1 Générer le titre

Le titre se génère automatiquement avec la commande `\maketitle` qui doit être placée après la commande `\begin{document}`. Les options du titre s'obtiennent avec les commandes `\author{}`, `\title{}` et `\date{}` qui doivent elles être placées avant la commande `\begin{document}`. Par exemple :

```

%%
% Commandes de Titre (A ADAPTER)
%%

\author{Fabien Augsburg}
\date{\today}
\title{Introduction à \LaTeX}

%%
% Debut du document
%%

\begin{document}

%%
% Titre
%%

\maketitle

```

5.2 Modifier les entêtes et les pieds de pages

Comme expliqué dans la section 3.4, le package `fancyhdr` permet de personnaliser les entêtes et les pieds de page. Une mise en page particulière compatible avec un travail de maturité est déjà proposée dans le document `documentTM.tex`. Il suffit d'adapter la partie mise en page dans le fichier source avec son nom :

```

%%
% Mise en page (A ADAPTER)
%%

%Entete et bas de page
\renewcommand{\headrulewidth}{.4pt}
\fancyhead[R]{\thepage}
\fancyhead[L]{\textnormal{\leftmark}}
\fancyhead[C]{}
\renewcommand{\footrulewidth}{.4pt}
\fancyfoot[R]{Fabien Augsburg} % A adapter ici
\fancyfoot[L]{Travail de maturité}
\fancyfoot[C]{}

```

5.3 Créer la table des matières

En tapant la commande `\tableofcontents` dans le texte, la table des matières est automatiquement créée. Il faut cependant compiler au minimum deux fois (sans erreur) pour qu'elle s'insère à l'endroit où la commande a été placée. La profondeur de la table des matières peut être déterminée avec la commande `\setcounter{tocdepth}{valeur}`. Par exemple

```

%%
% Tables des matières
%%

\setcounter{tocdepth}{2} \tableofcontents

```

5.4 Changer le style du texte

Différentes commandes permettent de changer localement le corps du texte et sa justification. Le code suivant

```
Je peux écrire en \textbf{gras}, en \textit{italique}, en {\scshape petites capitales} ou encore plus {\large grand}.

\begin{center}
Mais le summum, c'est de \emph{mettre en évidence} en centrant.
\end{center}
```

engendre le texte suivant

```
Je peux écrire en gras, en italique, en PETITES CAPITALES ou encore plus grand.
           Mais le summum, c'est de mettre en évidence en centrant.
```

Cet exemple est non exhaustif. Pour d'autres changements, consulter les menus de votre éditeur ou les références.

5.5 Créer une liste

Pour créer une liste, il suffit d'employer l'environnement `itemize` (liste non numérotée) ou `enumerate` (liste numérotée). Chaque élément de la liste doit alors être précédé de la commande `\item`. Par exemple, la liste de la page 10 a été engendrée par le code suivant :

```
\begin{itemize}
\item Votre fichier est automatiquement sauvé chaque fois que vous compilez.
\item On repère plus facilement les éventuelles erreurs.
\end{itemize}
```

Dans l'éditeur TEXMAKER, ces commandes peuvent s'obtenir dans le menu *LaTeX*.

Pour les variantes avec des symboles particuliers, voir les références.

5.6 Insérer du texte mathématique

Pour rappel, une formule insérée dans du texte s'obtient avec le Math-Mode, c'est-à-dire avec la syntaxe `texte $formule$ texte`. Une formule mise en évidence s'obtient avec le Display-Mode, c'est-à-dire avec la syntaxe `texte \[formule\] texte`.

Par exemple le code suivant

```
Les fonctions affines ont pour équation $f(x)=mx+h$. Ainsi
\[f(x)=3x-2\]
est un exemple de fonction affine.
```

engendre le texte suivant

```
Les fonctions affines ont pour équation  $f(x) = mx + h$ . Ainsi
           
$$f(x) = 3x - 2$$

est un exemple de fonction affine.
```

Il faut bien faire attention à l'utilisation des délimiteurs. Par exemple, en Math-Mode,

`a_n+1` va produire $a_n + 1$ `a^10` va produire a^{10}
`a_{n+1}` va produire a_{n+1} `a^{10}` va produire a^{10}

Conseil

Chaque fois que vous ouvrez un délimiteur `{` ou `$`, refermez-le immédiatement avec `}` ou `$` et tapez ensuite votre texte entre.

De plus, beaucoup d'expressions mathématiques ne s'insèrent pas bien dans un texte :

la fraction `$$\frac{a+b}{c}c$` est va produire la fraction $\frac{a+b}{c}$ est
`$c= \lim_{x \to +\infty} \sin(x)$` va produire $c = \lim_{x \rightarrow +\infty} \sin(x)$

Pour la première expression, il est judicieux d'écrire la fraction avec un autre symbole. Pour la seconde, elle se révèle plus agréable en Display-Mode :

la fraction `$$ (a+b)/c $` est va produire la fraction $(a + b)/c$ est
`\[c= \lim_{x \to +\infty} \sin(x)\]` va produire $c = \lim_{x \rightarrow +\infty} \sin(x)$

On note que la fonction sinus s'obtient avec le macro `\sin`. De tels macros existent pour toutes les fonctions. Dans TEXMAKER , voir le menu *Math*.

Dans le Display-Mode, il est également possible d'aligner des expressions. Pour y parvenir, l'environnement `align` est à utiliser (et non l'environnement `eqnarray`). L'alignement se fait avec le caractère spécial `&` et la fin d'une ligne se marque avec un double backslash `\\`.

Par exemple le code suivant

```
\begin{align*}
f(x)&=ax^2+bx+c\\
&=a(x-x_S)^2+y_S
\end{align*}
```

engendre le texte suivant

$$\left| \begin{array}{l} f(x) = ax^2 + bx + c \\ \quad = a(x - x_S)^2 + y_S \end{array} \right.$$

Si on désire que les lignes soient numérotées, il suffit de remplacer l'environnement `align*` par `align`.

5.7 Créer un tableau

L'environnement `tabular` permet de créer un tableau. Cependant la création d'un tableau demande une réflexion préalable ce qui en fait certainement un des points faibles de \LaTeX . Il faut en effet déterminer a priori l'alignement de chaque colonne (et donc indirectement leur nombre) dans le préambule `\begin{tabular}{r|l|c|c}` de l'environnement. Ici, on commence un tableau à 4 colonnes, la première étant alignée à droite, la deuxième à gauche et les deux dernières sont centrées mais séparées d'un filet vertical.

L'écriture se fait alors en ligne, la commande `&` permettant de passer à la colonne suivante et la commande `\cr` permettant de passer à la ligne suivante. Pour créer un filet horizontal, il faut insérer la commande `\hline` au début de la ligne.

Par exemple, le code suivant

```

Le tableau suivant
\begin{center}
\begin{tabular}{r|l}
{Avantages} & {Désavantages} \cr \hline
Nombreux & Faibles
\end{tabular}
%\caption[Avantages et désavantages de \LaTeX]{Avantages et désavantages}
\end{center}
permet de

```

engendre le texte suivant

```

Le tableau suivant


| Avantages | Désavantages |
|-----------|--------------|
| Nombreux  | Faibles      |


permet de

```

Pour ajouter une légende à un tableau, il faut emboîter l'environnement `tabular` dans un environnement `table`. On peut ensuite utiliser la commande `\caption[Légende dans la table des tableaux]{Légende dans le texte}` pour insérer la légende voulue. Le tableau est alors numéroté automatiquement.

Les tableaux en Math-Mode s'obtiennent avec l'environnement `array`. Pour des tableaux plus compliqués, consulter les références.

5.8 Insérer une figure

Pour insérer une figure avec les manipulations suivantes, il faut déclarer le package `graphicx` en préambule. Il est souvent pratique de regrouper les fichiers des images insérées dans un dossier. Si ce dossier est appelé `fig` et est un sous-dossier du dossier dans lequel le document `DocumentTM.tex` est placé, il suffit de placer la commande `\graphicspath{./fig/}` à l'endroit indiqué.

Avec la commande `\includegraphics[dimension]{nomfichierimage}`, on insère alors facilement une image de format `eps` ou `jpg`. Cette commande doit être emboîtée dans un environnement `figure`. On peut ensuite utiliser la commande `\caption[Légende dans la table des figures]{Légende dans le texte}` pour insérer la légende voulue. La figure est alors numéroté automatiquement.

Par exemple, si le sous-dossier `fig` contient l'image `canal.eps`, le code

```

%%%%
% Packages
%%%%

%Figure
\usepackage{graphicx}

%%%%
% Debut du document
%%%%

\begin{document}

%%%%
% Chemin des figures (A ADAPTER)
%%%%

\graphicspath{./fig/}

```

```

%%%%
% Texte
%%%%

\begin{figure}[!ht]
\begin{center}
\includegraphics[width=6cm]{canal.eps}% height=3cm si on veut spécifier la hauteur
\end{center}
\end{figure}

```

On peut également insérer d'autres formats, modifier certains paramètres de l'image (écriture, etc) à partir de L^AT_EX. Par exemple, le package `wrapfigure` permet d'habiller une image par du texte. Pour plus d'informations, voir les références.

5.9 Créer une note de bas de page

Pour créer une note de bas de page, taper la commande `\footnote{Texte de la note de bas de page}` dans le texte juste après le mot auquel la note fait référence. Les notes sont numérotées automatiquement.

Par exemple, le code suivant

```

Le mathématicien \textsc{Euler}\footnote{\textsc{Leonhard Paul Euler}, mathématicien et physicien
suisse, 1707--1783} fit d'importantes découvertes dans des domaines variés.

```

engendre le texte suivant

```

Le mathématicien EULERa fit d'importantes découvertes dans des domaines variés.


---


aLEONHARD PAUL EULER, mathématicien et physicien suisse, 1707–1783

```

5.10 Générer une référence

Pour faire référence à une section, une définition, une expression mathématique, un tableau, une figure, une page, il faut référencer l'objet à l'aide de la commande `\label{label-name}` après la commande `\{section}` ou `\begin{env-name}`. Ensuite, la commande `\ref{label-name}` placée à l'endroit où doit figurer la référence permet d'obtenir le numéro de la section, la définition, etc. Pour obtenir le numéro de la page à laquelle se situe la référence, utiliser la commande `\ref{label-name}`. Par exemple, le code suivant

```

Une fonction affine \label{fctaffine} est une fonction d'équation  $f(x)=mx+h$ .
...
La fonction  $f$  est une fonction affine (voir définition de la page \pageref{fctaffine}) et admet
donc un seul zéro.

```

engendre le texte suivant

```

Une fonction affine est une fonction d'équation  $f(x) = mx + h$ .
...
La fonction  $f$  est une fonction affine (voir définition de la page 15) et admet donc un seul zéro.

```

Remarquez qu'il faut au moins compiler deux fois pour que la référence apparaisse !

5.11 Générer la bibliographie

La bibliographie se crée avec l’environnement `\begin{thebibliography}{99}`. Elle se place généralement après le corps de texte mais en tous les cas toujours **avant** la commande `\end{document}`!. Chaque entrée de la bibliographie doit être précédée par la commande `\bibitem{label1}`. Le style de la bibliographie peut être modifié à l’aide de la commande `\bibliographystyle{style}`. Les styles possibles les plus courants sont `plain`, `alpha` ou `abbrv`.

Par exemple, la bibliographie de ce document est créée à l’aide des lignes de codes suivantes :

```
\bibliographystyle{alpha}
\begin{thebibliography}{99}
\bibitem{teximpatient} {\sc Abrahams, Hargreaves \& Berry}, \emph{\TeX pour l'Impatient}, 2003-4.
Disponible à l'adresse {\tt http://www.tex.ac.uk/tex-archive/info/impatient/fr/fbook.pdf}

\bibitem{lc} {\sc Goossens \& MittelBach}, \emph{The \LaTeX\ Companion}, Addison-Wesley Publishing
Company, New York, 2004.

\bibitem{knu} {\sc Knuth}, \emph{Le \TeX book}, Vuibert, 2003.

\bibitem{lam} {\sc Lamport}, \emph{\LaTeX\ (second edition)}, Addison-Wesley Publishing Company.

\bibitem{texoser} {\sc Lozano}, \emph{Tout ce que vous avez toujours voulu savoir sur \LaTeX sans
jamais oser le demander}, 2007. Disponible à l'adresse {\tt http://cours.enise.fr/info/latex/}

\bibitem{cil} {\sc Oetiker \& al}, \emph{Une courte (?) introduction à \LaTeXe\ }, 2004. Disponible
à l'adresse \texttt{http$://$mirror.switch.ch/ftp/mirror/tex/info/lshort/french}
\end{thebibliography}
```

Pour faire référence à un livre, il suffit de taper la commande `\cite[texte facultatif]{label}` dans le texte. Par exemple, la référence de la dernière ligne du paragraphe 3.6 de la page 8 est obtenue avec le code suivant :

```
Voir \cite[pages 251-55]{lc} pour une liste compl'ete des options et des exemples.
```

5.12 Insérer les annexes

Différents annexes peuvent être créés automatiquement. Les plus simples à obtenir sont les listes des tableaux et les listes des figures. Il suffit d’insérer les commandes `\listoftables` et `\listoffigures` à l’endroit où on veut voir apparaître la table. Il faut cependant noter que seuls les tableaux et figures légendés avec la commande `\caption` apparaîtront dans les listes.

Pour créer un index ou insérer d’autres annexes, consulter les références.

5.13 Pour aller plus loin

Ce document est une première introduction grossière à L^AT_EX. Si vous voulez faire quelque chose de plus spécifique, référez-vous aux ouvrages suivants qui figurent ci-dessous.

Bibliographie

- [1] ABRAHAMS, HARGREAVES & BERRY, *TeX pour l'Impatient*, 2003-4. Disponible à l'adresse <http://www.tex.ac.uk/tex-archive/info/impatient/fr/fbook.pdf>
- [2] GOOSSENS & MITTELBACH, *The \LaTeX Companion*, Addison-Wesley Publishing Company, New York, 2004.
- [3] KNUTH, *Le TeXbook*, Vuibert, 2003.
- [4] LAMPORT, *\LaTeX (second edition)*, Addison-Wesley Publishing Company.
- [5] LOZANO, *Tout ce que vous avez toujours voulu savoir sur \LaTeX sans jamais oser le demander*, 2007. Disponible à l'adresse <http://cours.enise.fr/info/latex/>
- [6] OETIKER & AL, *Une courte (?) introduction à $\text{\LaTeX} 2_{\epsilon}$* , 2004. Disponible à l'adresse <http://mirror.switch.ch/ftp/mirror/tex/info/lshort/french>