

Écrire en L^AT_EX

Cours préparé par : H. MENIGHER

Chapitre 0 : Premier pas.

Chapitre 1 : Structure d'un document.

Chapitre 2 : Les caractères.

Chapitre 3 : Mise en forme de texte.

Chapitre 4 : Le mode mathématique.

Chapitre 5 : Les tableaux.

Chapitre 6 : Les images.

Chapitre 7 : Références et bibliographie.

Chapitre 0 : Premier pas

0.1. L^AT_EX.

0.2. Composition d'un document.

0.3. Installation.

0.4. Vocabulaire.

0.5. Structure d'un fichier L^AT_EX.

Il existe plusieurs catégories de logiciels pour gérer du texte.

Éditeur de texte : (Bloc-notes, Notepad++ ...)

- Destiné à la création et à l'édition de fichiers textes sans mise en forme des polices, des couleurs, de justification ...
- Les fichiers produits sont légers et exploitables par d'autres éditeurs de texte.

Traitement de texte : (Microsoft Word, OpenOffice ...)

- Produit une composition du texte à partir des données textuelles.
- Le format de sauvegarde retient les informations de composition : mise en forme, couleurs, polices ...
- Les fichiers sont plus gros.
- Les fichiers sont souvent moins exploitables par d'autres logiciels.

Logiciel de mise en page : (Adobe InDesign, QuarkXpress ...)

- Réalise une composition de très grande qualité typographique à partir de textes et d'images.
- Logiciels haut de gamme, donc souvent chers.
- Apprentissage difficile, souvent réservés à des professionnels.

0.1. L^AT_EX

L^AT_EX est un logiciel de composition de documents professionnel, gratuit et multi-plateforme permet de produire des documents de haute qualité typographique. Il permet facilement d'intégrer dans un document des formules mathématiques, des graphes, des schémas, etc.

Le système \LaTeX a été inventé par Leslie Lamport en 1983, sur base du système \TeX inventé par Donald Knuth à partir de 1977. \LaTeX , qui est d'ailleurs l'abréviation de Lamport \TeX , est devenu la méthode privilégiée d'écriture de documents (articles, rapports, livres, etc.) dans la communauté scientifique, en particulier du côté des sciences exactes et appliquées.

0.2. Composition d'un document

La rédaction d'un document avec \LaTeX consiste à écrire le code source du document. Il s'agit simplement d'un fichier texte, dont l'extension est `.tex`. Une fois ce fichier écrit, on le compile à l'aide du compilateur `pdflatex` pour obtenir un document final, au format PDF.

0.3. Installation

Pour pouvoir rédiger des documents en \LaTeX , deux éléments essentiels sont à avoir sur ton ordinateur :

- Une distribution est une collection de logiciels (des compilateurs, des convertisseurs, etc.) et de packages (fournissant des extensions à \LaTeX) ;
- Un éditeur est un programme qui permet de rédiger le code source des documents et propose des outils pour aider à la rédaction du document, notamment un correcteur orthographique.

Les distribution les plus courants : MiKTeX, TeXLive.

Les éditeur les plus courants : Winedt, TeXstudio.

0.4. Vocabulaire

préambule : Le préambule est la première partie de tout document \LaTeX . Il permet de spécifier plusieurs informations très importantes, comme par exemple le type de document à créer (rapport, livre...) ou encore les différents packages (extensions et fichiers de style) que l'on souhaite utiliser dans le document. On renseigne également dans le préambule le titre du document, l'auteur, la date ...

fichier source : Le fichier source (ou document \LaTeX), contient le texte du document et les commandes de mise en page. C'est un fichier texte, écrit avec n'importe quel éditeur de texte, avec l'extension `.tex`.

commande : Ce sont les instructions de base de \LaTeX qui permettent de changer des paramètres de mise en forme du texte, d'insérer un tableau ou une image. Les commandes sont identifiées par des noms qui commencent tous par un `\` (backslash).

environnement : Les environnements sont des commandes spéciales qui modifient gros blocs de texte. Ils permettent de mémoriser une mise en forme de texte et de l'appliquer ainsi à plusieurs endroits du documents. Il existe des environnements prédéfinis pour créer par exemple des listes, des tableaux, ... mais il est également possible de définir ses propres environnements.

La modification de la définition d'un environnement entraine la modification de la mise en forme du texte partout où l'on a utilisé cet environnement.

Les environnements commencent presque toujours par `\begin{environnement}` et se terminent par `\end{environnement}`, où le nom de l'environnement remplace `environnement`. La partie du document entre le `\begin{ }` et `\end{ }` est dans l'environnement choisi.

théorème : Afin de regrouper les structures similaires telles que les définitions, les propositions, les lemmes,..., \LaTeX propose un environnement générique qui permet de dériver ces structures avec ajout de numérotations.

fichier de style : C'est un fichier qui contient des définitions de commandes, de théorèmes ou d'environnements ou des commandes de mise en page de document. On regroupe souvent ces instructions dans un même fichier pour pouvoir les inclure plus facilement dans les différents documents que l'on rédige.

élément flottant : C'est une structure particulière de \LaTeX qui permet d'insérer un élément sans spécifier exactement son emplacement, celui-ci sera décider lors de la compilation afin de laisser le moins d'espace vide possible.

mode texte : On dit que \LaTeX est en mode texte s'il compile une portion de texte comme un paragraphe, un tableau, une énumération,... S'il n'est pas en mode texte, il est en mode mathématique. Certaines instructions ne peuvent pas être utilisées dans ce mode.

mode mathématique : \LaTeX se trouve en mode mathématique s'il compile une formule mathématique. Il utilise une mise en forme totalement différente de celle du mode texte. Par exemple, il ne respecte pas les espaces qui sont insérés.

0.5. Structure d'un fichier LaTeX

Tous les fichiers \LaTeX doivent commencer par la commande `\documentclass`. Cette commande indique à \LaTeX de quel type de document il s'agit, d'un livre, rapport, article, ...

Ensuite viennent les commandes qui vont changer le style du document ou charger d'éventuelles extensions avec la commande `\usepackage`. Cette partie est appelée le préambule.

Une fois tout cela fait, le corps du document peut enfin commencer. Le corps du document se trouve dans l'environnement `document`. Il commence donc par la commande `\begin{document}`.

Maintenant, vous pouvez saisir le texte de votre document et une fois cela terminé, il suffit d'utiliser la commande `\end{document}` qui signale la fin de l'environnement `document` et donc la fin du corps de votre document.

Corps du document	Préambule	$\backslash\text{documentclass}\{\text{report}\}$ $\left\{ \begin{array}{l} \dots \\ \dots \\ \dots \end{array} \right.$ $\backslash\text{begin}\{\text{document}\}$ $\left\{ \begin{array}{l} \dots \\ \dots \\ \dots \end{array} \right.$ $\backslash\text{end}\{\text{document}\}$	Exemple :	$\backslash\text{documentclass}[12\text{pt},\text{a4paper}]\{\text{report}\}$ $\backslash\text{usepackage}[\text{utf8}]\{\text{inputenc}\}$ $\backslash\text{usepackage}[\text{francais}]\{\text{babel}\}$ $\backslash\text{setlength}\{\backslash\text{parskip}\}\{0.5\text{em}\}$ $\backslash\text{renewcommand}\{\backslash\text{baselinestretch}\}\{1.1\}$ $\backslash\text{begin}\{\text{document}\}$ Département de Mathématiques Première Année Master Proba $\backslash\text{end}\{\text{document}\}$

Chapitre 1 : Structure d'un document

1.1. Structure d'un fichier.
1.2. Caractères spéciaux.
1.3. Espaces et longueurs.
1.4. Classe de document.
1.5. Sectionnement.
1.6. Mise en page.
1.7. Table des matières.

1.1. Structure d'un fichier

Tout fichier \TeX est bâti selon le même canevas constitué :

- d'une commande `\documentclass` : obligatoire et unique, placer à la première ligne. Son rôle est de spécifier la "classe" de document qu'on souhaite rédiger.
- d'un préambule : c'est l'ensemble (éventuellement vide) de ce qui figure entre les commandes `\documentclass` et `\begin{document}`.
- d'un environnement `document` : celui-ci est obligatoire.
- d'un corps de document : c'est tout ce qui se trouve à l'intérieur de l'environnement `document`, donc entre les commandes `\begin{document}` et `\end{document}`.

1.2. Caractères spéciaux

Les caractères spéciaux, dits aussi réservés, de \LaTeX sont les caractères suivants :

Caractère	Rôle	S'obtient avec
{	délimiteur ouvrante	\{
}	délimiteur fermante	\}
%	commentaire	\%
#	macro personnelle	\#
\$	mode mathématique	\\$
^	exposant	\^
~	espace insécable	\~
&	tableau	\&
_	indice	_
\	commande	\textbackslash

1.3. Espaces et longueurs

1.3.1. Espaces

Le changement de ligne simple équivaut à une espace. Par contre, si on laisse une ou plusieurs lignes vides, cela indique un changement de paragraphe. On peut laisser plusieurs lignes vides, cela équivaut à une seule ligne vide. De même, si l'on met plusieurs espace, c'est comme s'il n'y en avait qu'une seule.

– **Commandes pour les espaces horizontaux fixes :**

<code>~</code>	espace insécable.
<code>\,</code>	espace fine.
<code>\:</code>	espace moyenne.
<code>\;</code>	espace large.
<code>\quad</code>	espace plus large (cadratin).
<code>\qquad</code>	espace encore plus large (double cadratin).
<code>\hspace*{2cm}</code>	exemple d'espace horizontal de largeur donnée.

– **Commandes pour les espaces horizontaux "élastiques"**

<code>\hfill</code>	remplit l'espace restant sur la ligne courante.
<code>\dotfill</code>	remplit l'espace restant sur la ligne courante par des pointillés.
<code>\hrulefill</code>	remplit l'espace restant sur la ligne courante par un filet horizontal.

– **Commandes pour les espaces verticaux fixes**

<code>\smallskip</code>	petit espace vertical.
<code>\medskip</code>	espace vertical moyen.
<code>\bigskip</code>	grand espace vertical.
<code>\vspace*{2cm}</code>	exemple d'espace vertical de hauteur donnée.

– **Commandes pour les espaces verticaux "élastiques"**

<code>\vfill</code>	remplit l'espace vertical restant sur la page courante.
---------------------	---

– **Autres espacements**

<code>\newline</code>	impose un changement de ligne à l'endroit du document où la commande est utilisée.
<code>\linebreak</code>	suggère à L ^A T _E X de couper la ligne à l'endroit du document où la commande est utilisée avec respect de la mise en forme du texte. Commande inverse : <code>\nolinebreak</code> .
<code>\newpage</code>	impose un changement de page à l'endroit du document où la commande est utilisée.
<code>\pagebreak</code>	suggère à L ^A T _E X de changer de page à l'endroit du document où la commande est utilisée avec respect de la mise en forme du texte. Commande inverse : <code>\nopagebreak</code> .

1.3.2. Longueurs

Le langage T_EX permet de spécifier des longueurs en utilisant plusieurs systèmes d'unités.

ex : la hauteur d'un "x" (minuscule) dans la police courante.

em : la largeur d'un "M" (majuscule) dans la police courante.

	Unité	en pt	en mm
sp	scale point	1.526×10^{-5}	5.363×10^{-6}
pt	point	1	0.3515
cc	cicéro	11.21	3.942
pc	pica	12	4.218
mm	millimètre	2.845	1
cm	centimètre	28.45	10
in	pouce	72.27	25.4

1.4. Classe de document

Syntaxe : `\documentclass[option]{class}`

où : `option` est un argument optionnel, et `classe` est un argument obligatoire.

Les différentes classes disponibles par défaut sont :

article pour des articles destinés à la publication et ne contenant que quelques pages.

report pour des documents un peu plus longs contenant plusieurs chapitres, comme des mémoires de thèse.

book pour de véritables livres, de plusieurs centaines de pages.

beamer pour faire des présentations utilisant la magnifique extension beamer.

Il existe aussi d'autres classes moins répandues :

slides pour faire des présentations sur transparents.

lettre pour faire des lettres au format français.

memoir pour écrire des mémoires, par exemple de fin d'étude.

Le choix de la classe va déterminer un certain nombre de paramètres par défaut, comme par exemple les marges.

1.4.1. Classe book

La classe book est prévue pour faire un livre. Les spécificités du livre sont les suivantes :

- il dispose d'une page de titre séparée, suivie d'une page blanche.
- il peut se décomposer en parties, chapitres, sections, sous-sections, sous-sous-sections, paragraphes et sous-paragraphes.
- les parties et chapitres commencent sur une page impaire.
- les marges sont assez grandes pour permettre une lecture aisée (par rapport à la quantité du texte).

1.4.2. Classe report

La classe report est similaire à la classe book, mais les chapitres ne commencent pas nécessairement en page impaire, et l'on ne peut pas utiliser certaines fonctions comme `\frontmatter`, `\mainmatter` et `\backmatter`. Par contre, elle dispose d'un environnement `abstract` permettant la mise en forme automatique d'un résumé.

1.4.3. Classe article

Par rapport à un livre, un article :

- a son titre sur la même page que le début du texte.
- a des marges plus étroites (mise en page dans un journal).
- a moins de subdivisions de texte : il n'y a pas de chapitre.

Quelques valeurs de l'argument `option` :

10pt, 11pt, 12pt	taille de police (valeur par défaut = 10pt).
a4paper, a5paper ...	taille de page (valeur par défaut = letterpaper = 11in×8.5in).
onecolumn, twocolumn	multiples colonnes (valeur par défaut = onecolumn).
landscape	mode paysage.
fleqn	alignement à gauche des formules mathématiques.

leqno numéros de formules sur le coté gauche au lieu de droite.

1.5. Sectionnement

Il existe différentes commandes qui permettent de délimiter les parties logiques d'un document. Celles-ci dépendent de la classe de document utilisée.

On définit les titres par les commandes `\part`, `\chapter`, `\section`, `\subsection`, `\subsubsection`, `\paragraph` et `\subparagraph`. Celles-ci permettent de définir respectivement une partie, un chapitre, différents niveaux de sections et des paragraphes. Elles prennent toutes un paramètre qui est le texte du titre et une option qui est une version courte du titre. Cette dernière sera notamment utilisée dans la table des matières.

Enfin, toutes ces commandes existent en version étoilée. Dans ce cas, le titre ne sera pas numéroté et n'apparaîtra pas dans la table des matières.

1.6. Mise en page

1.6.1. Les marges

On détermine les marges en écrivant dans le préambule

```
\usepackage{geometry}
\geometry{left=ll, right=rr, top=tt, bottom=bb}
```

ou bien

```
\usepackage[left=ll, right=rr, top=tt, bottom=bb]{geometry}
```

où : `ll` : la marge gauche, `rr` : la marge droite, `tt` : la marge haut, `bb` : la marge bas.

1.6.2. En-tête et pied de page

Le contenu de l'en-tête et du pied de page d'un document varie selon le style de page. On dispose des styles :

- `empty` : laisse l'en-tête et le pied de page vides.
- `plain` : est utilisé par défaut pour les classes `article` et `report`. Il laisse l'en-tête vide et compose le numéro de page au milieu du pied de page.
- `headings` : employé par défaut pour la classe `book`. Il laisse vide le pied de page et compose l'extérieur de l'en-tête avec le numéro de page. L'intérieur de l'en-tête est composé avec l'unité de sectionnement courantes.

On peut changer le style de l'en-tête et pied de page au moyen des commandes `\pagestyle{style}` et `\thispagestyle{style}`. La première commande applique le style spécifié depuis la page courante jusqu'à la fin du document, tandis que la deuxième n'a d'effet que sur la page en cours.

Pour modifier le format par défaut de l'en-tête et du pied de page, charger le package `fancyhdr` et l'activez le :

```
\usepackage{fancyhdr}
\pagestyle{fancy}
```

et utiliser les commandes suivantes selon le besoin :

- `\lhead{text}` : zone gauche de l'en-tête.
- `\chead{text}` : zone centre de l'en-tête.
- `\rhead{text}` : zone droite de l'en-tête.

- `\lfoot{text}` : zone gauche du pied de page.
- `\cfoot{text}` : zone centre du pied de page.
- `\rfoot{text}` : zone droite du pied de page.
- `\leftmark` : nom de la section courante.
- `\rightmark` : numéro et nom de la sous-section courante.
- `\thepage` : numéro de la page courante.
- `\thechapter` : numéro du chapitre courant.
- `\thesection`, `\thesubsection` : numéro de la section, sous-section courante.

Pour changer le style de numéros de pages on utilise la commande `\pagenumbering{style}` où `style` est à choisir parmi :

`\arabic` : 1, 2, 3 ... `\Alph` : A, B, C ... `\alph` : a, b, c ...
`\Roman` : I, II, III ... `\roman` : i, ii, iii ...

1.7. Table des matières

Pour inclure une table des matières il suffit d'insérer la commande `\tableofcontents` à l'endroit où vous souhaitez afficher la table des matières dans votre document.

Lorsqu'un élément de structure est suivi de `*` (par exemple : `\chapter*{titre}`), il n'y a pas de référence vers celui-ci dans la table des matières. Pour forcer L^AT_EX à ajouter une entrée il suffit d'utiliser la commande `\addcontentsline{toc}{chapter}{titre}`.

Chapitre 2 : Les caractères

- 2.1. Style et taille de caractères.
- 2.2. Commande avancée : le package `extsizes`.
- 2.3. Approche alternative : le package `anyfontsize`.
- 2.4. Caractères accentués et francisation du document.

2.1. Style et taille de caractères

Une police de caractères est d'abord une famille (*family*) de différentes fontes. En interne, L^AT_EX installe trois familles de polices.

Famille romane : c'est la police utilisée par défaut.

Famille sans empattements (sans serif) : cette police sert occasionnellement, par exemple pour des titres.

Famille à écartement fixe : c'est une police où l'espacement entre les lettres est constant. On l'appelle aussi `typewriter`.

Chaque famille est déclinée en plusieurs graisses (*series*) : principale (medium), épaisse (**bold**).

Chaque graisse peut se subdiviser en plusieurs formes (*shapes*) : droite (up), italique (*italic*), penchée (*slanted*), petites capitales (SMALL CAPS).

Chacune de ces combinaisons peut se décliner en différentes tailles (*size*) : très petit à très grand.

Désigner une fonte de caractères revient donc à définir une combinaison : *encoding* – *family* –

series – *shapes* – *size*, où l’encodage (*encoding*) est celui désigné par le package `fontenc`.

Remarque : certaines combinaisons peuvent ne pas exister en fonction de la police utilisée.

2.1.1. Commandes de changement de style :

Commande déclarative	Commande à argument	Attribut	Type
<code>\rmfamily</code>	<code>\textrm{...}</code>	family	roman
<code>\sffamily</code>	<code>\textsf{...}</code>		sans serif
<code>\ttfamily</code>	<code>\texttt{...}</code>		type writer (machine à écrire)
<code>\mdseries</code>	<code>\textmd{...}</code>	series	medium (moyen)
<code>\bfseries</code>	<code>\textbf{...}</code>		bold (gras)
<code>\upshape</code>	<code>\textup{...}</code>	shape	up (en haut)
<code>\itshape</code>	<code>\textit{...}</code>		<i>italic (italique)</i>
<code>\slshape</code>	<code>\textsl{...}</code>		<i>slanted (incliné)</i>
<code>\scshape</code>	<code>\textsc{...}</code>		SMALL CAPS (PETITES CAPITALES)
<code>\normalfont</code>	<code>\textnormal{...}</code>		medium, up (moyen, en haut)

2.1.2. Commandes de changement de taille :

Commande déclarative	Environnement	Taille
<code>\tiny</code>	<code>tiny</code>	minuscule
<code>\scriptsize</code>	<code>scriptsize</code>	taille des indices
<code>\footnotesize</code>	<code>footnotesize</code>	taille des notes de bas de page
<code>\small</code>	<code>small</code>	petit
<code>\normalsize</code>	<code>normalsize</code>	normal
<code>\large</code>	<code>large</code>	grand
<code>\Large</code>	<code>Large</code>	très grand
<code>\LARGE</code>	<code>LARGE</code>	très très grand
<code>\huge</code>	<code>huge</code>	énorme
<code>\Huge</code>	<code>Huge</code>	vraiment énorme

2.2. Commande avancée : le package `extsizes`

Les classes standard support trois tailles de police : 10pt, 11pt et 12pt. La taille est définie par l’argument optionnelle de la classe, par exemple : `\documentclass[12pt]{report}`.

Si vous avez besoin d’une taille différente, utilisez le package `extsizes`. Il permet les tailles : 8pt, 9pt, 10pt, 11pt, 12pt, 14pt, 17pt, 20pt.

Les noms de classe de document sont légèrement différents des classes standard : `extbook`, `extarticle`, `extreport`.

2.3. Approche alternative : le package `anyfontsize`

Le premier argument de la commande `fontsize{ }{ }` du package `anyfontsize` fixe la taille de la police en pt, et le seconde fixe la hauteur minimale (en pt) de l’interligne.

Exemple : `{\fontsize{24}{28}\selectfont texte ... } texte ...`

2.4. Caractères accentués et francisation du document

Vous pouvez taper les accents de la manière suivante : `\accent lettre`. Par exemple : `\'a \^a \^a` donnera á à â.

Pour taper directement les accents sans avoir à les composer, charger le package `inputenc` avec l'option `utf8` : `\usepackage[utf8]{inputenc}`.

Pour franciser le fonctionnement de \LaTeX utiliser le package `babel` avec l'option `french` ou `francais`. Il traduit les titres et applique les règles de typographie française.

Chapitre 3 : Mise en forme de texte

- 3.1. Alignement des paragraphes.
- 3.2. Alinéa (indentation des paragraphes).
- 3.3. Espacement interligne et entre paragraphes.
- 3.4. Césure.
- 3.5. Structures de liste.
- 3.6. Listes personnalisées.

3.1. Alignement des paragraphes

Les paragraphes en \LaTeX sont habituellement entièrement justifiés (les lignes font toutes la même longueur). Si vous souhaitez changer la justification d'un paragraphe, \LaTeX met à votre disposition trois environnements, ainsi que des commandes équivalentes.

aligné à gauche :

`\begin{flushleft} texte \end{flushleft}` ou `{\raggedright texte \par}`.

aligné à droite :

`\begin{flushright} texte \end{flushright}` ou `{\raggedleft texte \par}`.

centré :

`\begin{center} texte \end{center}` ou `{\centering texte \par}`.

Remarques :

- `flush` pouvant se traduire par «chasser vers».
- les commandes `\raggedright` et `\raggedleft` font référence au côté opposé ; `ragged` signifie en typographie française «en drapeau» (drapeau droit pour l'alignement à gauche, drapeau gauche pour l'alignement à droite).

3.2. Alinéa (indentation des paragraphes)

En typographie, un alinéa est un retrait de la première ligne d'un paragraphe. La taille du retrait est déterminée par un paramètre appelé le `\parindent`. Sa longueur par défaut est fixée par la classe du document que vous employez. Il est possible de l'imposer en utilisant la commande `\setlength` avec la syntaxe : `\setlength{\parindent}{longueur}`.

- On indique un nouvel alinéa, ou paragraphe, en laissant une ou plusieurs lignes vides.
- On peut annuler l'alinéa en mettant la commande `\noindent` (commande inverse de `\indent`) en début de paragraphe.
- Au sein d'un alinéa, on peut faire un retour à la ligne en mettant deux anti-slaches `\\`.

3.3. Espacement interligne et entre paragraphes

L'extension (package) `setspace` permet d'augmenter l'interlignage, en introduisant l'environnement `spacing`, ou bien en utilisant la commande `\setstretch`.

Par exemple, pour avoir un interligne de 1.2 fois l'interligne normal on écrit :

```
\begin{spacing}{1.2} texte ... \end{spacing}
```

ou

```
\setstretch{1.2} texte ...
```

L'espace entre les paragraphes est contenu dans `\parskip`, qui pourrait être modifié au moyen de la commande `\setlength` avec la syntaxe : `\setlength{\parskip}{longueur}`.

3.4. Césure

La césure (coupure d'un mot en fin de ligne afin de respecter la justification et le gris typographique) est gérée automatiquement par LaTeX. Il peut toutefois arriver que la césure ne soit pas correcte (par exemple, la césure modifie la prononciation).

Pour indiquer l'endroit où peut avoir lieu la césure, on utilise la commande `\-`. Par exemple, l'instruction `auto\ -matique\ -ment` indique à LaTeX qu'il pourra couper «automatiquement» après le «o» ou bien après le «e».

Si l'on veut empêcher une césure, il suffit de mettre le mot à l'intérieur d'un `\mbox{...}`, par exemple `\mbox{automatiquement}` empêche la césure du mot «automatiquement».

3.5. Structures de liste

Les structures de liste en LaTeX sont simplement des environnements qui se présentent en trois formes différentes : `itemize`, `enumerate`, `description`.

itemize: permet d'afficher des listes à puces (étiquettes).	<code>\begin{itemize}</code>		
	<code>\item texte</code>		— texte
	<code>\item texte</code>		— texte
	<code>\item[(A)] texte</code>		(A) texte
	<code>\end{itemize}</code>		

enumerate: conçu pour représenter des listes numérotées.	<code>\begin{enumerate}</code>		
	<code>\item texte</code>		1. texte
	<code>\item texte</code>		2. texte
	<code>\item[(A)] texte</code>		(A) texte
	<code>\end{enumerate}</code>		

description: il vous permet de remplacer les puces par des expressions de votre choix.

<code>\begin{description}</code>	
<code>\item[expression] texte</code>	expression texte
<code>\item[expr] texte</code>	expr texte
<code>\item texte</code>	texte
<code>\item[1)] texte</code>	1) texte
<code>\end{description}</code>	

L^AT_EX vous permet d'insérer un environnement de liste dans une autre liste existante jusqu'à une profondeur de quatre.

3.6. Listes personnalisées

3.6.1. Listes numérotées personnalisées :

À tout objet que L^AT_EX numérote automatiquement, comme les pages, est associé un compteur qui contrôle la numérotation. Chaque compteur possède un format par défaut. De tels formats sont modifiables en utilisant des commandes internes de L^AT_EX.

Il existe quatre compteurs différents qui sont associés aux listes numérotées, chacun représentant les quatre niveaux possibles, ils s'appellent : **enumi**, **enumii**, **enumiii**, **enumiv**. Pour obtenir l'élément numéroté, employez simplement la commande `\the` suivie immédiatement (c'est-à-dire sans aucun espace) du nom du compteur, par exemple `\theenumi`.

Commandes pour effectuer des changements sur la mise en forme dn premier niveau :

<code>\renewcommand{\theenumi}{définition}</code>	modifier le style de numérotation
<code>\renewcommand{\labelenumi}{définition}</code>	modifier le format de numérotation

Exemple :

<code>\renewcommand{\theenumi}{\roman{enumi}}</code>	
<code>\renewcommand{\labelenumi}{-\theenumi-}</code>	
<code>\begin{enumerate}</code>	-i- texte
<code>\item texte</code>	-ii- texte
<code>\item texte</code>	1) texte
<code>\item[1)] texte</code>	
<code>\end{enumerate}</code>	

3.6.2. Listes à puces personnalisées :

Les listes à puces n'utilisent pas de compteurs. Ainsi pour personnaliser de telles listes, vous pouvez juste modifier les puces (étiquettes). Les puces sont accessibles via les commandes : `\labelitemi`, `\labelitemii`, `\labelitemiii`, `\labelitemiv`.

Exemple :

<code>\renewcommand{\labelitemi}{*}</code>	
<code>\begin{itemize}</code>	* texte
<code>\item texte</code>	* texte
<code>\item texte</code>	– texte
<code>\item[--] texte</code>	
<code>\end{itemize}</code>	

Chapitre 4 : Le mode mathématiques

4.1. Les deux modes mathématiques.
4.2. Les environnements mathématiques.
4.3. Fonctions usuelles.
4.4. Les délimiteurs.
4.5. Style et taille des caractères.
4.6. Les matrices.
4.7. L'environnement <code>theorem</code> .

Pour écrire des mathématiques, les extensions (packages) `amsmath` et `amssymb` sont très utilisées, vous pouvez inclure systématiquement.

4.1. Les deux modes mathématiques

En dehors du cas des environnements (équations), il existe deux modes pour écrire des mathématiques.

4.1.1. Le mode en ligne : les symboles mathématiques sont mêlés au texte ; une telle formule commence par `$` et se termine par `$`, ou commence par `\(` et finit par `\)`.

4.1.2. Le mode hors ligne : les éléments mathématiques sont centrés sur une nouvelle ligne ; une telle formule commence par `\[` et se termine par `\]`, ou bien s'écrit dans l'environnement `displaymath`.

4.2. Les environnements mathématiques

L'équation est automatiquement numérotée par \LaTeX et, en ajoutant une étiquette avec `\label{key}`, on peut faire référence plus loin dans le document avec la commande `\eqref{key}`.

4.2.1. Pour une équation sur une seule ligne : l'environnement `equation`

Exemple :

```
\begin{equation}
(a+b)^3=a^3+3a^2b+3ab^2+b^3 \label{Newton3}
\end{equation}
L'équation \eqref{Newton3} ...
```

$$(a+b)^3 = a^3 + 3a^2b + 3ab^2 + b^3 \quad (1.1)$$

L'équation (1.1) ...

4.2.2. Pour une équation sur plusieurs lignes :

- On utilise `\\` pour sauter la ligne.
- Il est possible d'insérer un texte au sein des calculs grâce à la commande `\intertext{text}`.
- Pour retirer la numérotation d'une certaine ligne ajouter `\notag` ou `nonumber` dans cette ligne.

► **align pour les équations alignées :**

- On utilise `&` pour aligner sur le symbole écrit après.

Exemple :

```
\begin{align}
(a+b)^3 &= (a+b)(a+b)^2 \notag \\
\end{align}
```

```

&= (a+b)(a^2+2ab+b^2) \notag \\
\intertext{en développant}
&= a^3+3a^2b+3ab^2+b^3
\end{align}

```

$$\begin{aligned}
 (a+b)^3 &= (a+b)(a+b)^2 \\
 &= (a+b)(a^2+2ab+b^2)
 \end{aligned}$$

en développant

$$= a^3 + 3a^2b + 3ab^2 + b^3 \quad (1.2)$$

Exemple : (complexe)

Ici Nous organisons les équations dans trois colonnes. \LaTeX suppose que chaque équation se compose de deux parties séparées par `&`. De même que chaque équation est séparée de celle précédemment par un `&`.

```

\begin{align}
x \&= y \qquad \& \quad w \&= z \qquad \& \quad a \&= b+c \\
2x \&= -y \qquad \& \quad 3w \&= \frac{1}{2}z \qquad \& \quad a \&= b \\
-4 + 5x \&= 2+y \qquad \& \quad w+2 \&= -1+w \qquad \& \quad ab \&= cb
\end{align}

```

$$x = y \qquad w = z \qquad a = b + c \quad (1.3)$$

$$2x = -y \qquad 3w = \frac{1}{2}z \qquad a = b \quad (1.4)$$

$$-4 + 5x = 2 + y \qquad w + 2 = -1 + w \qquad ab = cb \quad (1.5)$$

► **gather pour les équations non alignées :**

– L'utilisation du symbole `&` entraine une erreur.

Exemple :

```

\begin{gather}
(a+b)^3 = (a+b)(a^2+2ab+b^2) \notag \\
= a^3+3a^2b+3ab^2+b^3
\end{gather}

```

$$\begin{aligned}
 (a+b)^3 &= (a+b)(a^2+2ab+b^2) \\
 &= a^3 + 3a^2b + 3ab^2 + b^3
 \end{aligned} \quad (1.6)$$

4.2.3. Pour une équation très longue : l'environnement `multline`

– On utilise `\\` à l'endroit où nous voulons couper l'équation.

Remarque :

Les environnements précédents admettent les versions étoilées `equation*`, `align*`, `gather*` et `multline*` qui permettent de retirer la numérotation.

Exemple :

```

\begin{multline}
(a+b+c+d)^3 = a^3 + b^3 + c^3 + d^3 + \\
3a^2b + 3a^2c + 3a^2d + 3b^2a + 3b^2c + 3b^2d + \\
3c^2a + 3c^2b + 3c^2d + 3d^2a + 3d^2b + 3d^2c + \\
6abc + 6abd + 6acd + 6bcd
\end{multline}

```

$$\begin{aligned}
 (a + b + c + d)^3 = & a^3 + b^3 + c^3 + d^3 + \\
 & 3a^2b + 3a^2c + 3a^2d + 3b^2a + 3b^2c + 3b^2d + \\
 & 3c^2a + 3c^2b + 3c^2d + 3d^2a + 3d^2b + 3d^2c + \\
 & 6abc + 6abd + 6acd + 6bcd \quad (1.7)
 \end{aligned}$$

4.2.4. L'environnement cases :

- Rend plusieurs lignes avec une accolade gauche extensible.
- Il doit apparaître dans le style mathématique.

Exemple :

$$\begin{array}{l}
 \text{\texttt{\$}\texttt{\backslash begin{cases}} \\
 \text{\texttt{p(x)y' + q(x)y = f(x)} \texttt{\backslash\backslash} \\
 \text{\texttt{y(0) = y_0}} \\
 \text{\texttt{\backslash end{cases}}\texttt{\$}}
 \end{array}
 \left|
 \begin{array}{l}
 p(x)y' + q(x)y = f(x) \\
 y(0) = y_0
 \end{array}
 \right.$$

Exemple :

$$\begin{array}{l}
 \text{\texttt{\$|x|=\backslash begin{cases}} \\
 \text{\texttt{x, \& x \backslash geq 0 \backslash\backslash}} \\
 \text{\texttt{-x, \& x < 0}} \\
 \text{\texttt{\backslash end{cases}}\texttt{\$}}
 \end{array}
 \left|
 \begin{array}{l}
 |x| = \begin{cases} x, & x \geq 0 \\ -x, & x < 0 \end{cases}
 \end{array}
 \right.$$

Remarques :

- \LaTeX ignore l'espace entre les symboles dans le mode mathématique.
- Aucune ligne vide n'est autorisée dans les environnements mathématiques (à l'exception l'environnement `cases`).

4.3. Fonctions usuelles

4.3.1. Fonctions de base :

Indice : <code>nombre_{indice}</code>	Racine : <code>\sqrt[ordre]{nombre}</code>
Exposant : <code>nombre^{exposant}</code>	Limite : <code>\lim_{var \rightarrow nbr}</code>

Exemple :

$$\begin{array}{l}
 \text{\texttt{\backslash[u_{n_0} , e^{x^2} , \sqrt[3]{27} , \sqrt{9} , \lim_{x \rightarrow x_0} f(x) ,}} \\
 \text{\texttt{\backslash lim u_n \backslash}}
 \end{array}$$

$$u_{n_0} \quad , \quad e^{x^2} \quad , \quad \sqrt[3]{27} \quad , \quad \sqrt{9} \quad , \quad \lim_{x \rightarrow x_0} f(x) \quad , \quad \lim u_n$$

Remarque : D'autres fonctions mathématiques sont regroupées dans `TeXstudio` au menu `Maths` → `Math Functions`.

4.3.2. Grands opérateurs :

Fraction : <code>\frac{numérateur}{dénominateur}</code>	Somme : <code>\sum_{min}^{max}</code>
Intégrale : <code>\int_{min}^{max}</code>	Produit : <code>\prod_{min}^{max}</code>

Exemple :

$$\begin{array}{l}
 \text{\texttt{\backslash[\frac{f(x)}{g(x)} , \int_0^{+\infty} f(x)dx , \sum_{i=0}^n u_i ,}} \\
 \text{\texttt{\backslash prod_{i=0}^{+\infty} x_i \backslash}}
 \end{array}$$

$$\frac{f(x)}{g(x)} \quad , \quad \int_0^{+\infty} f(x)dx \quad , \quad \sum_{i=0}^n u_i \quad , \quad \prod_{i=0}^{+\infty} x_i$$

► Commandes de style :

– `\limits` : pour le style display. – `\nolimits` : pour le style texte.

Exemple :

```
\[ \sum\nolimits_{i=0}^n u_i \quad , \quad \prod\nolimits_{i=0}^{+\infty} x_i \]
```

$$\sum_{i=0}^n u_i \quad , \quad \prod_{i=0}^{+\infty} x_i$$

Exemple :

```
$ \int_0^{+\infty} f(x)dx \quad , \quad \int\limits_0^{+\infty} f(x)dx \quad , \\ \sum_{i=0}^n u_i \quad , \quad \sum\limits_{i=0}^n u_i \quad , \quad \prod_{i=0}^{+\infty} x_i \quad , \\ \prod\limits_{i=0}^{+\infty} x_i $
```

$$\int_0^{+\infty} f(x)dx \quad , \quad \int_0^{+\infty} f(x)dx \quad , \quad \sum_{i=0}^n u_i \quad , \quad \sum_{i=0}^n u_i \quad , \quad \prod_{i=0}^{+\infty} x_i \quad , \quad \prod_{i=0}^{+\infty} x_i$$

4.4. Les délimiteurs

La taille des symboles (les parenthèses par exemple) qui entourent une expression mathématique peut être ajustée de deux façons.

La première est automatique, on fait précéder le symbole de gauche de la commande `\left` et celui de droite de la commande `\right`.

Si l'on ne veut qu'un seul délimiteur, on utilise « . » de l'autre côté.

Exemple :

```
\[ (\frac{a}{b}) \quad , \quad \left(\frac{a}{b}\right) \quad , \quad \left(\frac{a}{b}\right). \]
```

$$\left(\frac{a}{b}\right) \quad , \quad \left(\frac{a}{b}\right) \quad , \quad \left(\frac{a}{b}\right)$$

La seconde méthode consiste à indiquer à \LaTeX quelle taille utiliser à l'aide des commandes `\big`, `\Big`, `\bigg`, `\Bigg` (par ordre croissant).

Les principaux délimiteurs ajustables sont : (,) , [,] , | , || , < , >

4.5. Style et taille des caractères

4.5.1. Commandes de changement de style :

`\mathrm` , `\mathit` , `\mathbf` , `\mathsf` , `\mathtt`.

4.5.2. Commandes de changement de taille :

- `\displaystyle` : taille des formules en mode math isolé.
- `\textstyle` : taille des formules en mode texte.
- `\scriptstyle` : taille des indices.
- `\scriptscriptstyle` : taille des indices des indices.

4.5.3. Quelques fontes mathématiques :

Sont des packages qui permettent d'écrire quelques symboles mathématiques.

Commande	Résultat	Package requis
<code>\mathcal{ABC}</code>	\mathcal{ABC}	/
<code>\mathscr{ABC}</code>	\mathscr{ABC}	<code>mathrsfs</code>
<code>\mathbb{ABC}</code>	\mathbb{ABC}	<code>amsmath</code>
<code>\mathds{ABC1}</code>	$\mathds{ABC1}$	<code>dsfont</code>

4.6. Les matrices

Les environnements matrice sont : `matrix`, `pmatrix`, `bmatrix`, `Bmatrix`, `vmatrix`, `Vmatrix`.

La différence entre les différents types de matrice est le style de délimiteurs qui entourent la matrice.

environnement	délimiteur	environnement	délimiteur	environnement	délimiteur
<code>matrix</code>	\cdots	<code>bmatrix</code>	$[\cdots]$	<code>vmatrix</code>	$ \cdots $
<code>pmatrix</code>	(\cdots)	<code>Bmatrix</code>	$\{\cdots\}$	<code>Vmatrix</code>	$\ \cdots\ $

Chaque ligne d'une matrice se termine par `\\` et chaque colonne se termine par `&` (esperluette).

Exemple :

```
\[ \det A=\begin{vmatrix}
a & b & c \\
d & e & f \\
g & h & i
\end{vmatrix} \]
```

$$\det A = \begin{vmatrix} a & b & c \\ d & e & f \\ g & h & i \end{vmatrix}$$

4.7. L'environnement theorem

La commande `\newtheorem` permet de créer et numéroter des théorèmes, des définitions, des propositions ...

Syntaxe : `\newtheorem{nom}[num]{titre}[division]`

où : – `nom` : nom de l'environnement.

– `num` : indique comment s'organise la numérotation.

– `titre` : nom du théorème.

– `division` : le niveau de sectionnement qui veut numéroter par le.

Exemple :

```
% au préambule
\newtheorem{theorem}{Théorème}[chapter]
\newtheorem{definition}[theorem]{Définition}
% au corps du document
\begin{definition}
  texte texte ...
\end{definition}
\begin{theorem}
  texte texte ...
\end{theorem}
```

Définition 1.1. *texte texte ...*

Théorème 1.2. *texte texte ...*

Chapitre 5 : Les tableaux

5.1. Insertion d'un tableau.
5.2. Filets.
5.3. Largeurs des colonnes.
5.4. Cellule sur plusieurs colonnes.
5.5. Cellule sur plusieurs lignes.
5.6. Cellule coupée par une barre oblique.
5.7. Rotation du texte. 5.8. Titre.

5.1. Insertion d'un tableau

On insère un tableau avec l'environnement `tabular` qui prend un paramètre qui décrit les colonnes du tableau. chaque colonne est définie à l'aide d'une lettre qui décrit l'alignement horizontal du texte dans cette colonne. Les valeurs possibles sont :

- `l` : texte aligné à gauche,
- `c` : texte centré,
- `r` : texte aligné à droite.

On sépare les colonnes par `&` et on sépare les lignes par `\\`.

Exemple :

<code>\begin{tabular}{lc}</code>	
<code>Module & Coefficient \\</code>	Module Coefficient
<code>\LaTeX & 2 \\</code>	L ^A T _E X 2
<code>Analyse & 5</code>	Analyse 5
<code>\end{tabular}</code>	

5.2. Filets

Les filets verticaux sont définis dans l'option de l'environnement `tabular`, il suffit d'insérer `|`. Les filets horizontaux sont définis avec les données du tableau. La commande `\hline` insère un filet horizontal.

Exemple :

<code>\begin{tabular}{l c}</code>	
<code>Module & Coefficient \\</code>	Module Coefficient
<code>\hline</code>	
<code>\LaTeX & 2 \\</code>	L ^A T _E X 2
<code>Analyse & 5</code>	Analyse 5
<code>\end{tabular}</code>	

On peut avoir des filets horizontaux partiels qui ne s'étendent que sur certaines colonnes à l'aide de la commande `\cline`, qui prend en paramètre les colonnes sur lesquelles le filet doit s'étendre.

Exemple :

<code>\begin{tabular}{ l c c c }</code>
<code>\hline</code>
<code>Module & Coefficient & Crédit & Évaluation \\ \hline</code>

```
\LaTeX & 2 & 2 & Examen \\
\cline{1-2}\cline{4-4}
Analyse & 5 & 10 & Examen \\ \hline
\end{tabular}
```

Module	Coefficient	Crédit	Évaluation
L ^A T _E X	2	2	Examen
Analyse	5	10	Examen

5.3. Largeurs des colonnes

Pour fixé la largeur d'une colonne on utilise le descripteur `p{length}`.

Exemple :

```
\begin{tabular}{|p{4cm}|p{3cm}|}
```

pour tracer un tableau de deux colonnes de largeur 4cm et 3cm respectivement.

Pour répéter un descripteur plusieurs fois utiliser le descripteur `*{nbr}{desc}` qui va répéter `nbr` fois le descripteur `desc`.

Exemple :

```
\begin{tabular}{*{3}{|c}*{2}{|l|}}
```

équivaux à

```
\begin{tabular}{|c|c|c|l|l|}
```

Exemple :

```
\begin{tabular}{*{3}{|p{2cm}}|}
```

équivaux à

```
\begin{tabular}{|p{2cm}|p{2cm}|p{2cm}|}
```

Astuce : Alignement de texte dans une colonne de largeur fixe.

Package requis : `\tabularx`

Exemple :

```
\begin{tabular}{|>\centering\arraybackslash p{3cm}|>\raggedleft
\arraybackslash p{3cm}{2cm}|}
```

pour centrer le texte de la première colonne et aligner à droite le texte de la deuxième colonne.

5.4. Cellule sur plusieurs colonnes

Syntaxe : `\multicolumn{nbr}{desc}{texte}`

- `nbr` : le nombre de colonnes sur lesquelles la cellule s'étend,
- `desc` : la description de la cellule,
- `texte` : le texte de la cellule.

Exemple :

```
\begin{tabular}{|l|c|c|c|}
\hline
Module & Coefficient & Crédit & Évaluation \\
\hline
\LaTeX & \multicolumn{2}{c|}{2} & Examen \\
\hline
Analyse & 5 & 10 & Examen \\
\hline
\end{tabular}
```

Module	Coefficient	Crédit	Évaluation
L ^A T _E X	2		Examen
Analyse	5	10	Examen

5.5. Cellule sur plusieurs lignes :

Package requis : `multirow`

Syntaxe : `\multirow{nbr}{desc}{texte}`

- `nbr` : le nombre de lignes sur lesquelles la cellule s'étend,
- `desc` : la largeur de la cellule,
- `texte` : le texte de la cellule.

Exemple :

```
\begin{tabular}{|l|c|c|c|}
\hline
Module & Coefficient & Crédit & Évaluation \\
\hline
LaTeX & 2 & 2 & \multirow{2}{2cm}{Examen} \\
\cline{1-3}
Analyse & 5 & 10 & \\
\hline
\end{tabular}
```

Module	Coefficient	Crédit	Évaluation
LaTeX	2	2	Examen
Analyse	5	10	

5.6. Cellule coupée par une barre oblique :

Package requis : `slashbox`

Syntaxe : `\backslashbox{texte1}{texte2}`

Exemple :

```
\begin{tabular}{|c|c|c|}
\hline
\backslashbox{Niveau}{Spécialité} & Analyse & EDP \\
\hline
Licence & 57 & 65 \\
\hline
Master & 34 & 31 \\
\hline
\end{tabular}
```

<div style="text-align: center;">\backslashbox{Niveau}{Spécialité}</div>	Analyse	EDP
Licence	57	65
Master	34	31

5.7. Rotation du texte :

Package requis : `graphicx`

Syntaxe : `\rotatebox{nbr}{texte}`

- `nbr` : angle de rotation (en degrés),
- `texte` : le texte que vous souhaitez faire pivoter.

Exemple :

```
\begin{tabular}{|c|c|c|}
\cline{2-3}
\multicolumn{1}{c|}{} & \textbf{Module} & \textbf{Coefficient} \\
\hline
\multirow{3}{3mm}{\rotatebox{90}{Licence}} & Topologie & 2 \\
\hline
\end{tabular}
```

```

\cline{2-3}
& Analyse & 4 \\
\cline{2-3}
& Algèbre & 4 \\
\hline

```

	Module	Coefficient
Licence	Topologie	2
	Analyse	4
	Algèbre	4

5.8. Titre :

Pour ajouter un titre à un tableau, il faut le placer dans un environnement `table` et utiliser la commande `\caption`.

Exemple :

```

\begin{table}[H]
\centering
\begin{tabular}{|c|c|}
\hline
\bfseries Module & Coefficient \\
\hline
\LaTeX & 2 \\
Analyse & 5 \\
\hline
\end{tabular}
\caption{Coefficients des modules.}
\end{table}

```

Module	Coefficient
\LaTeX	2
Analyse	5

TABLE 1.1 – Coefficients des modules.

On peut insérer une liste de tableaux à l'aide de la commande `\listoftables`

Chapitre 6 : Les images

- 6.1. Insérer une image.
- 6.2. Spécifier la taille de l'image.
- 6.3. Rotation de l'image.
- 6.4. Texte autour l'image.
- 6.5. Titre de l'image.

6.1. Insérer une image

Package requis : `graphicx`.

Syntaxe : `\includegraphics{nom.ext}`

- `nom` : nom de l'image,
- `ext` : l'extension de l'image (`png` ou `jpg`).

6.2. Spécifier la taille de l'image :

On peut spécifier la taille de l'image par des arguments optionnels.

- `width=longueur` : largeur de l'image.
- `height=longueur` : hauteur de l'image.
- `keepaspectratio=true` : pour garder le ratio largeur/hauteur.

Exemple :

```
\includegraphics[width=3cm]{univ.png}
\includegraphics[width=3cm,height=2cm]{univ.png}
```

6.3. Rotation de l'image :

Syntaxe : `\includegraphics[angle=nbr]{nom}`

- `nbr` : l'angle de rotation (en degrés).
- `nom` : nom de l'image.

Exemple :

```
\includegraphics[width=3cm,angle=45]{univ.png}
\includegraphics[width=3cm,angle=-45]{univ.png}
```

6.4. Texte autour l'image

On utilise l'environnement `wrapfigure` du package `wrapfig`.

syntaxe : `\begin{wrapfigure}[nbr]{pos}{larg}`
`\includegraphics{nom}`
`\end{wrapfigure}`

- `nbr` : nombre de lignes de texte se trouvant à la même hauteur que l'image.
- `pos` : le côté où il faut placer l'image (`l` = à gauche, `r` = à droite)
- `larg` : la largeur de la zone de laquelle l'image doit insérée.
- `nom` : nom de l'image.

Exemple :

```
\begin{wrapfigure}[3]{r}{1.5cm}
\includegraphics[width=1.5cm]{univ.png}
\end{wrapfigure}
texte ...
```

texte texte texte texte texte texte texte	univ.png
texte texte texte texte texte texte texte	
texte texte texte texte texte texte texte	
texte texte texte texte texte texte texte	

6.5. Titre de l'image :

Pour ajouter un titre à l'image, il faut la placer dans un environnement `figure` et utiliser la commande `\caption`.

Exemple :

```
\begin{figure}[H]
\centering
\includegraphics[width=1.5cm]{univ.png}
\caption{Université de Jijel \label{univ}}
\end{figure}
```

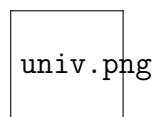


FIGURE 1.1 – Université de Jijel

On peut insérer une liste de figures à l'aide de la commande `\listoffigures`

Liste des tableaux

1	Coefficients des modules	20
---	--------------------------------	----

Table des figures

1	Université de Jijel	21
---	---------------------------	----

Chapitre 7 : Références et bibliographie

7.1. Références croisées.

7.2. Bibliographie.

7.1. Références croisées

Les références croisées permettent de poser des étiquettes à certains endroits (une figure, un tableau, une équation ...) et puis d'y faire référence automatiquement. L'étiquette est définie avec la commande `\label`. On y fait ensuite référence avec `\ref`, `\eqref`, ou avec `\pageref` pour faire référence à la page contenant cet élément.

Pour les figures et les tables, la commande `\label` doit être utilisée dans l'argument de la commande `\caption`.

Exemple :

La figure~\ref{univ} de la page~\pageref{univ} illustre le logo de l'université de Jijel.

La figure 1.1 de la page 22 illustre le logo de l'université de Jijel.

On fait chaque fois précéder les commandes `\ref`, `\eqref` et `\pageref` d'un tilde (~). Ceci permet d'éviter une césure entre le mot et le numéro.

7.2. Bibliographie :

Une bibliographie est insérée avec l'environnement `thebibliography`. Chaque élément est inséré avec la commande `\bibitem` qui prend comme paramètre une étiquette (pour faire référence à l'élément bibliographique).

L'environnement `thebibliography` prend en option la plus longue étiquette. Celle-ci sera utilisée pour aligner les différents éléments de la bibliographie.

Une fois une bibliographie insérée, on peut citer une référence en utilisant la commande `\cite`. Celle-ci prend en paramètre l'étiquette de l'entrée que vous souhaitez référencer. On peut également en donner plusieurs séparées par des virgules.

Exemple :

Pour plus de détaille consulter `\cite{fonct}`.

```
\begin{thebibliography}{00}
```

```
\bibitem{comp} \textbf{Alain Yger}: \textit{Analyse complexe}, Ellipses  
Édition Marketing, Paris cedex, 2014.
```

```
\bibitem{fonct} \textbf{Francis Hirsch} et \textbf{Gilles Lacombe}: \textit{Éléments d'analyse fonctionnelle}, 2ème édition, Dunod, Paris, 1999.
```

```
\end{thebibliography}
```

Pour plus de détaille consulter [2].

Bibliographie

[1] **Alain Yger** : *Analyse complexe*, Ellipses Édition Marketing, Paris cedex, 2014.

[2] **Francis Hirsch** et **Gilles Lacombe** : *Éléments d'analyse fonctionnelle*, 2ème édition, Dunod, Paris, 1999.

Maths +

Code 1 :

```
% au préambule: \DeclareMathOperator{\argch}{argcosh}
\[ \ln 1 \overset{\textit{déf}}{=} 0 \quad , \quad \quad
\cos x \underset{x \rightarrow 0}{\sim} 1 - \frac{x^2}{2} \quad , \quad \quad
|u_n| \xrightarrow[n \geq n_0]{\text{décroit}} 0 \quad , \quad \quad
\argch x = \ln \left( x + \sqrt{1+x^2} \right) \quad ]
```

$$\ln 1 \stackrel{\text{déf}}{=} 0 \quad , \quad \cos x \underset{x \rightarrow 0}{\sim} 1 - \frac{x^2}{2} \quad , \quad |u_n| \xrightarrow[n \geq n_0]{\text{décroit}} 0 \quad , \quad \operatorname{argcosh} x = \ln \left(x + \sqrt{1+x^2} \right)$$

Code 2 :

```
% au préambule: \newcommand{\dx}[1]{\,\,\textup{d}\,#1}
\[ \int f(x) \dx{x} \quad , \quad \quad
\sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq m}} a_{ij} \quad , \quad \quad
\iint\limits_{\substack{1 \leq x \leq 2 \\ 0 \leq y \leq x^2}} f(x,y) \dx{x} \dx{y} \quad ]
```

$$\int f(x) dx \quad , \quad \sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq m}} a_{ij} \quad , \quad \iint\limits_{\substack{1 \leq x \leq 2 \\ 0 \leq y \leq x^2}} f(x,y) dx dy$$

Code 3 :

```
\begin{align*}
(\sin x + \cos x)^2 &= \underbrace{\sin^2(x) + \cos^2(x)}_{1} + \overbrace{2 \sin x \cos x}^{\sin(2x)} \\
&= 1 + \sin(2x)
\end{align*}
```

$$\begin{aligned} (\sin x + \cos x)^2 &= \underbrace{\sin^2(x) + \cos^2(x)}_1 + \overbrace{2 \sin x \cos x}^{\sin(2x)} \\ &= 1 + \sin(2x) \end{aligned}$$

Code 4 :

```
% au préambule: \usepackage{amsthm}
% au corps du document:
\renewcommand{\proofname}{\textbf{\textup{Preuve}}}
%%
\begin{proof}
  texte texte ...
\end{proof}
```

Preuve. texte texte ...

□

Hammoud MENIGHER

menigher.hammoud@gmail.com