

Introduction à L^AT_EX

Introduction à l'utilisation des bases de données mathématiques
Méthodologie de la recherche
Formation au L^AT_EX

2^{ème} partie

Karl GROSSE-ERDMANN Aline GOULARD



2 mars 2017

③ Structures avancées

Les tableaux (1/2)

Exemple :

```
\begin{tabular}{l|c|c|}  
      & été & hiver\\  
\hline  
Paul   & 10  & 9\\  
\hline  
Pierre & 7   & 12\\  
\hline  
\end{tabular}
```

	été	hiver
Paul	10	9
Pierre	7	12

Les tableaux (2/2)

La lettre `l` écrite en argument en première ligne signifie que le texte de la première colonne sera aligné à gauche. Pour aligner à droite, on utilise `r` et pour centrer `c`. Il faut écrire autant de lettres qu'il n'y a de colonnes.

Les barres verticales `|` servent à tracer les lignes verticales tandis que les barres horizontales sont définies avec la commande `\hline`. Il suffit de doubler les commandes pour doubler le nombre de lignes à l'endroit voulu.

Les éléments des colonnes successives sont séparées par le caractère `&`. Les deux backslashes successifs `\\` indiquent la fin de la ligne.

Les tableaux (2/2)

La lettre `l` écrite en argument en première ligne signifie que le texte de la première colonne sera aligné à gauche. Pour aligner à droite, on utilise `r` et pour centrer `c`. Il faut écrire autant de lettres qu'il n'y a de colonnes.

Les barres verticales `|` servent à tracer les lignes verticales tandis que les barres horizontales sont définies avec la commande `\hline`. Il suffit de doubler les commandes pour doubler le nombre de lignes à l'endroit voulu.

Les éléments des colonnes successives sont séparées par le caractère `&`. Les deux backslashes successifs `\\` indiquent la fin de la ligne.

Les tableaux (2/2)

La lettre `l` écrite en argument en première ligne signifie que le texte de la première colonne sera aligné à gauche. Pour aligner à droite, on utilise `r` et pour centrer `c`. Il faut écrire autant de lettres qu'il n'y a de colonnes.

Les barres verticales `|` servent à tracer les lignes verticales tandis que les barres horizontales sont définies avec la commande `\hline`. Il suffit de doubler les commandes pour doubler le nombre de lignes à l'endroit voulu.

Les éléments des colonnes successives sont séparées par le caractère `&`. Les deux backslashes successifs `\\` indiquent la fin de la ligne.

Les matrices (1/4)

L'intérieur d'un environnement créant une matrice est similaire à celui d'un environnement `tabular`.

Les différents environnements de matrices sont à insérer dans un mode mathématique et chaque cellule sera donc interprétée comme une donnée mathématique.

Les matrices (1/4)

L'intérieur d'un environnement créant une matrice est similaire à celui d'un environnement `tabular`.

Les différents environnements de matrices sont à insérer dans un mode mathématique et chaque cellule sera donc interprétée comme une donnée mathématique.

Les matrices (2/4)

<pre>\begin{matrix} a & b \\ c & d \end{matrix}</pre>	$\begin{matrix} a & b \\ c & d \end{matrix}$
<pre>\begin{pmatrix} a & b \\ c & d \end{pmatrix}</pre>	$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$

Les matrices (3/4)

<pre>\begin{bmatrix} a & b \\ c & d \end{bmatrix}</pre>	$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$
<pre>\begin{Bmatrix} a & b \\ c & d \end{Bmatrix}</pre>	$\begin{Bmatrix} a & b \\ c & d \end{Bmatrix}$

Les matrices (4/4)

```
\begin{vmatrix}
a & b \\
c & d
\end{vmatrix}
```

$$\begin{vmatrix} a & b \\ c & d \end{vmatrix}$$

```
\begin{Vmatrix}
a & b \\
c & d
\end{Vmatrix}
```

$$\begin{Vmatrix} a & b \\ c & d \end{Vmatrix}$$

Aligner des équations (1/3)

1. `align` ou `align*` (selon si on numérote ou non les équations)

```
\begin{align*}
\tanh(z) &= \frac{e^z - e^{-z}}{e^z + e^{-z}} \\
&= \frac{e^{2z} - 1}{e^{2z} + 1}
\end{align*}
```

$$\begin{aligned}\tanh(z) &= \frac{e^z - e^{-z}}{e^z + e^{-z}} \\ &= \frac{e^{2z} - 1}{e^{2z} + 1}\end{aligned}$$

Aligner des équations (2/3)

2. alignat

Il est prévu pour aligner des objets différents sur plusieurs colonnes. D'autre part, l'utilisation de `{ }` permet d'obtenir des espacements « normaux ».

$$1 = 2x + y + z \quad (1)$$

$$2 = x + 3y + z \quad (2)$$

$$3 = x + y + 4z \quad (3)$$

est obtenu en écrivant

```
\begin{alignat}{4}
1 & = {} & 2x & + {} & y & + {} & z \\
2 & = {} & x & + {} & 3y & + {} & z \\
3 & = {} & x & + {} & y & + {} & 4z
\end{alignat}
```

Aligner des équations (2/3)

2. alignat

Il est prévu pour aligner des objets différents sur plusieurs colonnes. D'autre part, l'utilisation de `{ }` permet d'obtenir des espacements « normaux ».

$$1 = 2x + y + z \tag{1}$$

$$2 = x + 3y + z \tag{2}$$

$$3 = x + y + 4z \tag{3}$$

est obtenu en écrivant

```
\begin{alignat}{4}
1 &= {} & 2x &+ {} & y &+ {} & z \\
2 &= {} & x &+ {} & 3y &+ {} & z \\
3 &= {} & x &+ {} & y &+ {} & 4z
\end{alignat}
```

Aligner des équations (2/3)

2. alignat

Il est prévu pour aligner des objets différents sur plusieurs colonnes. D'autre part, l'utilisation de `{ }` permet d'obtenir des espacements « normaux ».

$$1 = 2x + y + z \tag{1}$$

$$2 = x + 3y + z \tag{2}$$

$$3 = x + y + 4z \tag{3}$$

est obtenu en écrivant

```
\begin{alignat}{4}
1 & ={} & 2x & +{} & y & +{} & z \\
2 & ={} & x & +{} & 3y & +{} & z \\
3 & ={} & x & +{} & y & +{} & 4z
\end{alignat}
```

Aligner des équations (3/3)

Il est également possible d'aligner des équations à l'intérieur d'un environnement `equation` ou `equation*`. Il faut pour cela utiliser un des environnements suivants.

```
\begin{equation*}
\begin{split}
\tanh(z) &= \frac{e^z - e^{-z}}{e^z + e^{-z}} \\
&= \frac{e^{2z} - 1}{e^{2z} + 1}
\end{split}
\end{equation*}
```

$$\begin{aligned}\tanh(z) &= \frac{e^z - e^{-z}}{e^z + e^{-z}} \\ &= \frac{e^{2z} - 1}{e^{2z} + 1}\end{aligned}$$

4. alignat

Il est utilisé de la même manière que `split`.

Aligner des équations (3/3)

Il est également possible d'aligner des équations à l'intérieur d'un environnement `equation` ou `equation*`. Il faut pour cela utiliser un des environnements suivants.

3. `split`

```
\begin{equation*}
\begin{split}
\tanh(z) &= \frac{e^z - e^{-z}}{e^z + e^{-z}} \\
&= \frac{e^{2z} - 1}{e^{2z} + 1}
\end{split}
\end{equation*}
```

$$\begin{aligned}\tanh(z) &= \frac{e^z - e^{-z}}{e^z + e^{-z}} \\ &= \frac{e^{2z} - 1}{e^{2z} + 1}\end{aligned}$$

Il est utilisé de la même manière que `split`.

Aligner des équations (3/3)

Il est également possible d'aligner des équations à l'intérieur d'un environnement `equation` ou `equation*`. Il faut pour cela utiliser un des environnements suivants.

3. `split`

```
\begin{equation*}
\begin{split}
\tanh(z) &= \frac{e^z - e^{-z}}{e^z + e^{-z}} \\
&= \frac{e^{2z} - 1}{e^{2z} + 1}
\end{split}
\end{equation*}
```

$$\begin{aligned}\tanh(z) &= \frac{e^z - e^{-z}}{e^z + e^{-z}} \\ &= \frac{e^{2z} - 1}{e^{2z} + 1}\end{aligned}$$

4. `aligned`

Il est utilisé de la même manière que `split`.

Les accolades

Dans un mode mathématique, il est possible d'utiliser des accolades pour les systèmes, les fonctions définies par morceaux, ...

On insère les expressions dans un environnement `cases` à l'intérieur duquel la structure est identique à celle d'un tableau.

Exemple :

$$f(x) = \begin{cases} \frac{e^{2x}-1}{e^{2x}+1} & \text{si } x \geq 0 \\ 0 & \text{sinon} \end{cases}$$

sera obtenu en écrivant

```
\[f(x)=
\begin{cases}
\frac{e^{2x}-1}{e^{2x}+1}&\text{si }x\geqslant 0\\
0 & \text{sinon}
\end{cases}
\end{cases}\]
```

Les accolades

Dans un mode mathématique, il est possible d'utiliser des accolades pour les systèmes, les fonctions définies par morceaux, ...
On insère les expressions dans un environnement `cases` à l'intérieur duquel la structure est identique à celle d'un tableau.

Exemple :

$$f(x) = \begin{cases} \frac{e^{2x}-1}{e^{2x}+1} & \text{si } x \geq 0 \\ 0 & \text{sinon} \end{cases}$$

sera obtenu en écrivant

```
\[f(x)=
\begin{cases}
\frac{e^{2x}-1}{e^{2x}+1}&\text{si }x\geqslant 0\\
0 & \text{sinon}
\end{cases}
\end{cases}\]
```

Les accolades

Dans un mode mathématique, il est possible d'utiliser des accolades pour les systèmes, les fonctions définies par morceaux, ...
On insère les expressions dans un environnement `cases` à l'intérieur duquel la structure est identique à celle d'un tableau.

Exemple :

$$f(x) = \begin{cases} \frac{e^{2x}-1}{e^{2x}+1} & \text{si } x \geq 0 \\ 0 & \text{sinon} \end{cases}$$

sera obtenu en écrivant

```
\[f(x)=  
\begin{cases}  
\frac{e^{2x}-1}{e^{2x}+1}&\text{si }x\geqslant 0\\  
0&\text{sinon}  
\end{cases}\]
```

Left right (1/4)

Quand on écrit des parenthèses, des accolades, des séparateurs, . . . , on souhaite que ceux-ci aient une taille suffisamment grande par rapport à ce qui y est inséré.

Pour cela, il existe les commandes `\left<` et `\right<`, où à la place de `<`, on insère le caractère délimiteur `\{`, `\}`, `[`, `]`, `|`, `\|`, . . .

Si l'un des deux est présent, l'autre doit obligatoirement l'être aussi. Si un délimiteur ne doit pas apparaître, on utilise le point `.` comme caractère.

Left right (1/4)

Quand on écrit des parenthèses, des accolades, des séparateurs, ..., on souhaite que ceux-ci aient une taille suffisamment grande par rapport à ce qui y est inséré.

Pour cela, il existe les commandes `\left*` et `\right*`, où à la place de `*`, on insère le caractère délimiteur `\{, \}`, `[,]`, `|, \|`, ...

Si l'un des deux est présent, l'autre doit obligatoirement l'être aussi. Si un délimiteur ne doit pas apparaître, on utilise le point (`.`) comme caractère.

Left right (1/4)

Quand on écrit des parenthèses, des accolades, des séparateurs, ..., on souhaite que ceux-ci aient une taille suffisamment grande par rapport à ce qui y est inséré.

Pour cela, il existe les commandes `\left*` et `\right*`, où à la place de `*`, on insère le caractère délimiteur `\{, \}`, `[,]`, `|, \|`, ...

Si l'un des deux est présent, l'autre doit obligatoirement l'être aussi. Si un délimiteur ne doit pas apparaître, on utilise le point `(.)` comme caractère.

Left right (2/4)

Exemples :

$$\left] \frac{1}{n}, 1 \right[\quad \text{et} \quad \left\{ \frac{1}{n} \mid n \in \mathbb{N} \right\}$$

sont obtenus avec

```
\left] \frac{1}{n}, 1 \right[  
\]
```

et

```
\left\{ \left( \frac{1}{n} \right) \mid n \in \mathbb{N} \right\}  
\right\}
```

Left right (2/4)

Exemples :

$$\left] \frac{1}{n}, 1 \right[\quad \text{et} \quad \left\{ \frac{1}{n} \mid n \in \mathbb{N} \right\}$$

sont obtenus avec

```
\left] \frac{1}{n}, 1 \right[
```

et

```
\left\{ \left. \frac{1}{n} \right| n \in \mathbb{N} \right\}
```

Left right (3/4)

Le résultat n'est pas toujours parfait. Dans ce cas, il vaut mieux utiliser les délimiteurs de taille fixe tels que `\big`, `\Big`, `\bigg`, `\Bigg`, ...

Left right (4/4)

Exemples :

$$\left] \frac{1}{n}, 1 \right[\quad \text{et} \quad \left\{ \frac{1}{n} \mid n \in \mathbb{N} \right\}$$

sont obtenus avec

```
\left[\frac{1}{n},1\right[
```

et

```
\left\{\frac{1}{n}\mid n\in\mathbb{N}\right\}
```

Left right (4/4)

Exemples :

$$\left] \frac{1}{n}, 1 \right[\quad \text{et} \quad \left\{ \frac{1}{n} \mid n \in \mathbb{N} \right\}$$

sont obtenus avec

```
\[  
\Big]\frac{1}{n},1\Big[  
\]
```

et

```
\[  
\Big\{\frac{1}{n}\Big|n\in\mathbb{N}\Big\  
\]
```

Les énumérations (1/2)

Forme générale :

```
\begin{type de l'énumération}  
\item énumération 1  
...  
\item énumération n  
\end{type de l'énumération}
```

Les différents types d'énumération sont : *itemize*, *enumerate*, *description*.

Les énumérations (1/2)

Forme générale :

```
\begin{type de l'énumération}  
\item énumération 1  
...  
\item énumération n  
\end{type de l'énumération}
```

Les différents types d'énumération sont : `itemize`, `enumerate`, `description`.

Les énumérations (2/2)

<pre>\begin{itemize} \item ligne 1 \item ligne 2 \end{itemize}</pre>	<ul style="list-style-type: none">- ligne 1- ligne 2
<pre>\begin{enumerate} \item ligne 1 \item ligne 2 \end{enumerate}</pre>	<ol style="list-style-type: none">1. ligne 12. ligne 2
<pre>\begin{description} \item[donnéeA] ligne 1 \item[donnéeB] ligne 2 \end{description}</pre>	<p>donnéeA ligne 1</p> <p>donnéeB ligne 2</p>

Les énumérations personnalisées (1/2)

Il est possible de changer le symbole servant à l'énumération et de mettre presque ce que l'on veut. Il suffit pour cela de mettre entre crochet le symbole pour chaque élément.

Exemple :

```
\begin{itemize}
\item[$\rightarrow$] ligne 1
\item[$\leftarrow$] ligne 2
\end{itemize}
```

donnera

```
→ ligne 1
← ligne 2
```

Les énumérations personnalisées (1/2)

Il est possible de changer le symbole servant à l'énumération et de mettre presque ce que l'on veut. Il suffit pour cela de mettre entre crochet le symbole pour chaque élément.

Exemple :

```
\begin{itemize}
\item[$\rightarrow$] ligne 1
\item[$\leftarrow$] ligne 2
\end{itemize}
```

donnera

→ ligne 1

← ligne 2

Les énumérations personnalisées (1/2)

Il est possible de changer le symbole servant à l'énumération et de mettre presque ce que l'on veut. Il suffit pour cela de mettre entre crochet le symbole pour chaque élément.

Exemple :

```
\begin{itemize}
\item[$\rightarrow$] ligne 1
\item[$\leftarrow$] ligne 2
\end{itemize}
```

donnera

→ ligne 1

← ligne 2

Les énumérations personnalisées (2/3)

Le package `enumitem` permet de personnaliser les trois environnements élémentaires. Il autorise des arguments optionnels qui sont un ensemble d'objets de la forme `paramètre=valeur`. Par exemple,

`\setlist` définit le dénombrement. Il est intéressant de l'utiliser avec les valeurs prédéfinies `\alph*`, `\Alph*`, `\arabic*`, `\roman*` OU `\Roman*`.

`\resumeenum` permet de continuer la liste précédente. Ce paramètre n'a pas de valeur.

`\listfrom` indique par quel nombre commencera l'énumération.

Les énumérations personnalisées (2/3)

Le package `enumitem` permet de personnaliser les trois environnements élémentaires. Il autorise des arguments optionnels qui sont un ensemble d'objets de la forme `paramètre=valeur`. Par exemple,

`label` définit le dénombrement. Il est intéressant de l'utiliser avec les valeurs prédéfinies `\alph*`, `\Alph*`, `\arabic*`, `\roman*` ou `\Roman*`.

`resume` permet de continuer la liste précédente. Ce paramètre n'a pas de valeur.

`start` indique par quel nombre commencera l'énumération.

Les énumérations personnalisées (2/3)

Le package `enumitem` permet de personnaliser les trois environnements élémentaires. Il autorise des arguments optionnels qui sont un ensemble d'objets de la forme `paramètre=valeur`. Par exemple,

`label` définit le dénombrement. Il est intéressant de l'utiliser avec les valeurs prédéfinies `\alph*`, `\Alph*`, `\arabic*`, `\roman*` ou `\Roman*`.

`resume` permet de continuer la liste précédente. Ce paramètre n'a pas de valeur.

`start` indique par quel nombre commencera l'énumération.

Les énumérations personnalisées (2/3)

Le package `enumitem` permet de personnaliser les trois environnements élémentaires. Il autorise des arguments optionnels qui sont un ensemble d'objets de la forme `paramètre=valeur`. Par exemple,

`label` définit le dénombrement. Il est intéressant de l'utiliser avec les valeurs prédéfinies `\alph*`, `\Alph*`, `\arabic*`, `\roman*` ou `\Roman*`.

`resume` permet de continuer la liste précédente. Ce paramètre n'a pas de valeur.

`start` indique par quel nombre commencera l'énumération.

Les énumérations personnalisées (3/3)

Une autre alternative pour personnaliser l'environnement `enumerate` est d'indiquer en option de l'environnement le début de l'élément (lettre ou chiffre et mise en forme) qui sera incrémenté. Pour cela, il faut insérer le package `enumerate`. Il est impossible d'utiliser simultanément les packages `enumitem` et `enumerate`.

Exemple :

```
\begin{enumerate}[(a.)]  
  \item ligne 1  
  \item ligne 2  
\end{enumerate}
```

donnera

(a.) ligne 1
(a.) ligne 2

Les énumérations personnalisées (3/3)

Une autre alternative pour personnaliser l'environnement `enumerate` est d'indiquer en option de l'environnement le début de l'élément (lettre ou chiffre et mise en forme) qui sera incrémenté. Pour cela, il faut insérer le package `enumerate`. Il est impossible d'utiliser simultanément les packages `enumitem` et `enumerate`.

Exemple :

```
\begin{enumerate} [(a.)]  
\item ligne 1  
\item ligne 2  
\end{enumerate}
```

donnera

(a.) ligne 1

(a.) ligne 2

Les énumérations personnalisées (3/3)

Une autre alternative pour personnaliser l'environnement `enumerate` est d'indiquer en option de l'environnement le début de l'élément (lettre ou chiffre et mise en forme) qui sera incrémenté. Pour cela, il faut insérer le package `enumerate`. Il est impossible d'utiliser simultanément les packages `enumitem` et `enumerate`.

Exemple :

```
\begin{enumerate} [(a.)]  
\item ligne 1  
\item ligne 2  
\end{enumerate}
```

donnera

- (a.) ligne 1
- (b.) ligne 2

La taille des caractères

Avec la même logique que pour les polices, on peut définir la taille des caractères en utilisant les commandes suivantes qui définissent des tailles de la plus petite à la plus grande. Elles sont utiles pour les titres de la page de garde mais déconseillées dans le document.

<code>\tiny</code>	<small>tiny</small>
<code>\scriptsize</code>	<small>scriptsize</small>
<code>\small</code>	<small>small</small>
<code>\normalsize</code>	<small>normalsize</small>
<code>\large</code>	large
<code>\Large</code>	Large
<code>\LARGE</code>	LARGE
<code>\huge</code>	huge
<code>\Huge</code>	Huge

④ Structure d'un document mathématique

Introduction

Principe fondamental

La mise en page d'un document est décidée par \LaTeX .

Donc, il ne faut pas imposer la mise en page à \LaTeX .

Il faut juste lui indiquer ce qu'on veut :

- un nouveau paragraphe,
- un théorème,
- la bibliographie,
- etc.

Nous allons discuter quelques éléments de la mise en page d'un document mathématique.

Principe fondamental

La mise en page d'un document est décidée par \LaTeX .

Donc, il ne faut pas imposer la mise en page à \LaTeX .

Il faut juste lui indiquer ce qu'on veut :

- un nouveau paragraphe,
- un théorème,
- la bibliographie,
- etc.

Nous allons discuter quelques éléments de la mise en page d'un document mathématique.

Principe fondamental

La mise en page d'un document est décidée par \LaTeX .

Donc, il ne faut pas imposer la mise en page à \LaTeX .

Il faut juste lui indiquer ce qu'on veut :

- un nouveau paragraphe,
- un théorème,
- la bibliographie,
- etc.

Nous allons discuter quelques éléments de la mise en page d'un document mathématique.

Principe fondamental

La mise en page d'un document est décidée par \LaTeX .

Donc, il ne faut pas imposer la mise en page à \LaTeX .

Il faut juste lui indiquer ce qu'on veut :

- un nouveau paragraphe,
- un théorème,
- la bibliographie,
- etc.

Nous allons discuter quelques éléments de la mise en page d'un document mathématique.

Principe fondamental

La mise en page d'un document est décidée par \LaTeX .

Donc, il ne faut pas imposer la mise en page à \LaTeX .

Il faut juste lui indiquer ce qu'on veut :

- un nouveau paragraphe,
- un théorème,
- la bibliographie,
- etc.

Nous allons discuter quelques éléments de la mise en page d'un document mathématique.

Le découpage du texte (1/3)

Le texte peut être découpé...

article : ... en sections, sous-sections, sous-sous-sections, etc.

book : ... (en parties), chapitres, sections, sous-sections, etc.

A la fin, on peut ajouter des annexes.

Les commandes sont les suivantes (un * supprime la numérotation) :

\chapter{C'est n'importe quoi}

\section{La première section}

\section{Une deuxième section}

\subsection{Une sous-section}

\subsection*{Une autre sous-section, sans numéro}

\appendix

\chapter{À ne pas oublier}

Dans tous les exemples qui suivent on a utilisé la classe book.

Voici le résultat.

Le découpage du texte (1/3)

Le texte peut être découpé...

article : ... en sections, sous-sections, sous-sous-sections, etc.

book : ... (en parties), chapitres, sections, sous-sections, etc.

A la fin, on peut ajouter des annexes.

Les commandes sont les suivantes (un * supprime la numérotation) :

\chapter* : C'est n'importe quoi !

\section : La première section !

\section* : Une deuxième section !

\subsection : Une sous-section !

\subsection* : Une autre sous-section, sans numéro !

\appendix :

\chapter* : À ne pas oublier !

Dans tous les exemples qui suivent on a utilisé la classe book.

Voici le résultat.

Le découpage du texte (1/3)

Le texte peut être découpé...

article : ... en sections, sous-sections, sous-sous-sections, etc.

book : ... (en parties), chapitres, sections, sous-sections, etc.

A la fin, on peut ajouter des annexes.

Les commandes sont les suivantes (un * supprime la numérotation) :

\chapter* : C'est n'importe quoi !

\section : La première section

\section* : Une deuxième section

\subsection : Une sous-section

\subsection* : Une autre sous-section, sans numéro

\part :

\chapter* : À ne pas oublier !

Dans tous les exemples qui suivent on a utilisé la classe book.

Voici le résultat.

Le découpage du texte (1/3)

Le texte peut être découpé...

`article` : ... en sections, sous-sections, sous-sous-sections, etc.

`book` : ... (en parties), chapitres, sections, sous-sections, etc.

A la fin, on peut ajouter des annexes.

Les commandes sont les suivantes (un * supprime la numérotation) :

`\section*` C'est n'importe quoi

`\section` La première section

`\section*` Une deuxième section

`\subsection` Une sous-section

`\subsection*` Une autre sous-section, sans numéro

`\chapter*` À ne pas oublier

Dans tous les exemples qui suivent on a utilisé la classe `book`.

Voici le résultat.

Le découpage du texte (1/3)

Le texte peut être découpé...

`article` : ... en sections, sous-sections, sous-sous-sections, etc.

`book` : ... (en parties), chapitres, sections, sous-sections, etc.

A la fin, on peut ajouter des annexes.

Les commandes sont les suivantes (un * supprime la numérotation) :

`\section{}` C'est n'importe quoi
`\section*` La première section
`\subsection{}` Une deuxième section
`\subsection*` Une sous-section
`\subsubsection{}` Une autre sous-section, sans numéro
`\subsubsubsection{}` À ne pas oublier

Dans tous les exemples qui suivent on a utilisé la classe `book`.

Voici le résultat.

Le découpage du texte (1/3)

Le texte peut être découpé...

`article` : ... en sections, sous-sections, sous-sous-sections, etc.

`book` : ... (en parties), chapitres, sections, sous-sections, etc.

A la fin, on peut ajouter des annexes.

Les commandes sont les suivantes (un * supprime la numérotation) :

`\chapter{C'est n'importe quoi}`

La première section

Une deuxième section

Une sous-section

Une autre sous-section, sans numéro

À ne pas oublier

Dans tous les exemples qui suivent on a utilisé la classe `book`.

Voici le résultat.

Le découpage du texte (1/3)

Le texte peut être découpé...

`article` : ... en sections, sous-sections, sous-sous-sections, etc.

`book` : ... (en parties), chapitres, sections, sous-sections, etc.

A la fin, on peut ajouter des annexes.

Les commandes sont les suivantes (un * supprime la numérotation) :

`\chapter{C'est n'importe quoi}`

`\section{La première section}`

`\section*{Une deuxième section}`

`\subsection{Une sous-section}`

`\subsection*{Une autre sous-section, sans numéro}`

`\chapter*{À ne pas oublier}`

Dans tous les exemples qui suivent on a utilisé la classe `book`.

Voici le résultat.

Le découpage du texte (1/3)

Le texte peut être découpé...

`article` : ... en sections, sous-sections, sous-sous-sections, etc.

`book` : ... (en parties), chapitres, sections, sous-sections, etc.

A la fin, on peut ajouter des annexes.

Les commandes sont les suivantes (un * supprime la numérotation) :

`\chapter{C'est n'importe quoi}`

`\section{La première section}`

`\section{Une deuxième section}`

`\section*{Une sous-section}`

`\section*{Une autre sous-section, sans numéro}`

`\chapter*{À ne pas oublier}`

Dans tous les exemples qui suivent on a utilisé la classe `book`.

Voici le résultat.

Le découpage du texte (1/3)

Le texte peut être découpé...

article : ... en sections, sous-sections, sous-sous-sections, etc.

book : ... (en parties), chapitres, sections, sous-sections, etc.

A la fin, on peut ajouter des annexes.

Les commandes sont les suivantes (un * supprime la numérotation) :

`\chapter{C'est n'importe quoi}`

`\section{La première section}`

`\section{Une deuxième section}`

`\subsection{Une sous-section}`

`\subsection*{Une autre sous-section, sans numéro}`

`\chapter*{À ne pas oublier}`

Dans tous les exemples qui suivent on a utilisé la classe book.

Voici le résultat.

Le découpage du texte (1/3)

Le texte peut être découpé...

article : ... en sections, sous-sections, sous-sous-sections, etc.

book : ... (en parties), chapitres, sections, sous-sections, etc.

A la fin, on peut ajouter des annexes.

Les commandes sont les suivantes (un * supprime la numérotation) :

```
\chapter{C'est n'importe quoi}
```

```
\section{La première section}
```

```
\section{Une deuxième section}
```

```
\subsection{Une sous-section}
```

```
\subsection*{Une autre sous-section, sans numéro}
```

À ne pas oublier

Dans tous les exemples qui suivent on a utilisé la classe book.

Voici le résultat.

Le découpage du texte (1/3)

Le texte peut être découpé...

article : ... en sections, sous-sections, sous-sous-sections, etc.

book : ... (en parties), chapitres, sections, sous-sections, etc.

A la fin, on peut ajouter des annexes.

Les commandes sont les suivantes (un * supprime la numérotation) :

`\chapter{C'est n'importe quoi}`

`\section{La première section}`

`\section{Une deuxième section}`

`\subsection{Une sous-section}`

`\subsection*{Une autre sous-section, sans numéro}`

`\appendix`

À ne pas oublier

Dans tous les exemples qui suivent on a utilisé la classe book.

Voici le résultat.

Le découpage du texte (1/3)

Le texte peut être découpé...

article : ... en sections, sous-sections, sous-sous-sections, etc.

book : ... (en parties), chapitres, sections, sous-sections, etc.

A la fin, on peut ajouter des annexes.

Les commandes sont les suivantes (un * supprime la numérotation) :

```
\chapter{C'est n'importe quoi}
```

```
\section{La première section}
```

```
\section{Une deuxième section}
```

```
\subsection{Une sous-section}
```

```
\subsection*{Une autre sous-section, sans numéro}
```

```
\appendix
```

```
\chapter{À ne pas oublier}
```

Dans tous les exemples qui suivent on a utilisé la classe book.

Voici le résultat.

Le découpage du texte (1/3)

Le texte peut être découpé...

article : ... en sections, sous-sections, sous-sous-sections, etc.

book : ... (en parties), chapitres, sections, sous-sections, etc.

A la fin, on peut ajouter des annexes.

Les commandes sont les suivantes (un * supprime la numérotation) :

`\chapter{C'est n'importe quoi}`

`\section{La première section}`

`\section{Une deuxième section}`

`\subsection{Une sous-section}`

`\subsection*{Une autre sous-section, sans numéro}`

`\appendix`

`\chapter{À ne pas oublier}`

Dans tous les exemples qui suivent on a utilisé la classe `book`.

Voici le résultat.

Le découpage du texte (1/3)

Le texte peut être découpé...

article : ... en sections, sous-sections, sous-sous-sections, etc.

book : ... (en parties), chapitres, sections, sous-sections, etc.

A la fin, on peut ajouter des annexes.

Les commandes sont les suivantes (un * supprime la numérotation) :

```
\chapter{C'est n'importe quoi}
```

```
\section{La première section}
```

```
\section{Une deuxième section}
```

```
\subsection{Une sous-section}
```

```
\subsection*{Une autre sous-section, sans numéro}
```

```
\appendix
```

```
\chapter{À ne pas oublier}
```

Dans tous les exemples qui suivent on a utilisé la classe book.

Voici le résultat.

Chapitre 1

C'est n'importe quoi

...

1.1 La première section

...

1.2 Une deuxième section

...

1.2.1 Une sous-section

...

Une autre sous-section, sans numéro

...

Annexe A

A ne pas oublier

Les environnements « théorème » et « définition » (1/8)

Pour les environnements comme les théorèmes, les propositions, les définitions, les exemples, les remarques, etc. il faut les définir d'abord dans le préambule.

```
\newtheorem{theoreme}{Théorème}  
\newtheorem{definition}{Définition}  
\newtheorem{exemple}{Exemple}
```

Dans le document on peut alors les utiliser :

```
\begin{theoreme}  
L'énoncé du théorème apparaît ici.  
\end{theoreme}
```

Voici le résultat.

Les environnements « théorème » et « définition » (1/8)

Pour les environnements comme les théorèmes, les propositions, les définitions, les exemples, les remarques, etc. il faut les définir d'abord dans le préambule.

```
\newtheorem{theoreme} {Théorème}
```

```
\newtheorem{definition} {Définition}
```

```
\newtheorem{exemple} {Exemple}
```

Dans le document on peut alors les utiliser :

```
\begin{theoreme}
```

L'énoncé du théorème apparaît ici.

```
\end{theoreme}
```

Voici le résultat.

Les environnements « théorème » et « définition » (1/8)

Pour les environnements comme les théorèmes, les propositions, les définitions, les exemples, les remarques, etc. il faut les définir d'abord dans le préambule.

```
\newtheorem{theoreme} {Théorème}  
\newtheorem{definition} {Définition}  
\newtheorem{exemple} {Exemple}
```

Dans le document on peut alors les utiliser :

```
\begin{theoreme}  
L'énoncé du théorème apparaît ici.  
\end{theoreme}
```

Voici le résultat.

Les environnements « théorème » et « définition » (1/8)

Pour les environnements comme les théorèmes, les propositions, les définitions, les exemples, les remarques, etc. il faut les définir d'abord dans le préambule.

```
\newtheorem{theoreme} {Théorème}  
\newtheorem{definition} {Définition}  
\newtheorem{exemple} {Exemple}
```

Dans le document on peut alors les utiliser :

```
\begin{theoreme}  
L'énoncé du théorème apparaît ici.  
\end{theoreme}
```

Voici le résultat.

Les environnements « théorème » et « définition » (1/8)

Pour les environnements comme les théorèmes, les propositions, les définitions, les exemples, les remarques, etc. il faut les définir d'abord dans le préambule.

```
\newtheorem{theoreme} { Théorème }  
\newtheorem{definition} { Définition }  
\newtheorem{exemple} { Exemple }
```

Dans le document on peut alors les utiliser :

```
\begin{theoreme}  
L'énoncé du théorème apparaît ici.  
\end{theoreme}
```

Voici le résultat.

Les environnements « théorème » et « définition » (2/8)

Théorème 1 *L'énoncé du théorème apparaît ici.*

Définition 1 *Ici on définit quelque chose.*

Théorème 2 *Un deuxième résultat.*

Exemple 1 *Ici un exemple.*

Remarque 1 *Ici une remarque.*

Les environnements « théorème » et « définition » (3/8)

On voit que chaque environnement est numéroté indépendamment.

Pour que la numérotation de la définition suive la numérotation des théorèmes il faut le dire :

```
\newtheorem{definition}{théorème} : Définition
```

De plus, les numéros peuvent aussi indiquer, par exemple, le chapitre dans lequel on se trouve :

```
\newtheorem{theoreme}{Théorème} (chapter)
```

On peut aussi ajouter des spécifications aux théorèmes, etc., par exemple :

```
\begin{theoreme}[Théorème de M. X]  
L'énoncé du théorème apparaît ici.  
\end{theoreme}
```

Voici le résultat.

Les environnements « théorème » et « définition » (3/8)

On voit que chaque environnement est numéroté indépendamment. Pour que la numérotation de la définition suive la numérotation des théorèmes il faut le dire :

```
\newtheorem{definition}{Définition}
```

De plus, les numéros peuvent aussi indiquer, par exemple, le chapitre dans lequel on se trouve :

```
\newtheorem{theoreme}{Théorème}[chapter]
```

On peut aussi ajouter des spécifications aux théorèmes, etc., par exemple :

```
\begin{theoreme}[Théorème de M. X]  
L'énoncé du théorème apparaît ici.  
\end{theoreme}
```

Voici le résultat.

Les environnements « théorème » et « définition » (3/8)

On voit que chaque environnement est numéroté indépendamment. Pour que la numérotation de la définition suive la numérotation des théorèmes il faut le dire :

```
\newtheorem{definition}[theoreme]{Définition}
```

De plus, les numéros peuvent aussi indiquer, par exemple, le chapitre dans lequel on se trouve :

```
\newtheorem{theoreme}{Théorème}[chapter]
```

On peut aussi ajouter des spécifications aux théorèmes, etc., par exemple :

```
\begin{theoreme}[Théorème de M. X]  
L'énoncé du théorème apparaît ici.  
\end{theoreme}
```

Voici le résultat.

Les environnements « théorème » et « définition » (3/8)

On voit que chaque environnement est numéroté indépendamment. Pour que la numérotation de la définition suive la numérotation des théorèmes il faut le dire :

```
\newtheorem{definition}[theoreme]{Définition}
```

De plus, les numéros peuvent aussi indiquer, par exemple, le chapitre dans lequel on se trouve :

```
theorem Théorème [chapter]
```

On peut aussi ajouter des spécifications aux théorèmes, etc., par exemple :

```
\begin{theorem}[Théorème de M. X]  
L'énoncé du théorème apparaît ici.  
\end{theorem}
```

Voici le résultat.

Les environnements « théorème » et « définition » (3/8)

On voit que chaque environnement est numéroté indépendamment. Pour que la numérotation de la définition suive la numérotation des théorèmes il faut le dire :

```
\newtheorem{definition} [theoreme] { Définition }
```

De plus, les numéros peuvent aussi indiquer, par exemple, le chapitre dans lequel on se trouve :

```
\newtheorem{theoreme} { Théorème } [chapter]
```

On peut aussi ajouter des spécifications aux théorèmes, etc., par exemple :

```
\begin{theoreme} [Théorème de M. X]  
L'énoncé du théorème apparaît ici.  
\end{theoreme}
```

Voici le résultat.

Les environnements « théorème » et « définition » (3/8)

On voit que chaque environnement est numéroté indépendamment. Pour que la numérotation de la définition suive la numérotation des théorèmes il faut le dire :

```
\newtheorem{definition} [theoreme] { Définition }
```

De plus, les numéros peuvent aussi indiquer, par exemple, le chapitre dans lequel on se trouve :

```
\newtheorem{theoreme} { Théorème } [chapter]
```

On peut aussi ajouter des spécifications aux théorèmes, etc., par exemple :

```
\begin{theoreme}[Théorème de M. X]  
L'énoncé du théorème apparaît ici.  
\end{theoreme}
```

Voici le résultat.

Les environnements « théorème » et « définition » (3/8)

On voit que chaque environnement est numéroté indépendamment. Pour que la numérotation de la définition suive la numérotation des théorèmes il faut le dire :

```
\newtheorem{definition}[theoreme]{Définition}
```

De plus, les numéros peuvent aussi indiquer, par exemple, le chapitre dans lequel on se trouve :

```
\newtheorem{theoreme}[Théorème][chapter]
```

On peut aussi ajouter des spécifications aux théorèmes, etc., par exemple :

```
\begin{theoreme}[Théorème de M. X]  
L'énoncé du théorème apparaît ici.  
\end{theoreme}
```

Voici le résultat :

Les environnements « théorème » et « définition » (3/8)

On voit que chaque environnement est numéroté indépendamment. Pour que la numérotation de la définition suive la numérotation des théorèmes il faut le dire :

```
\newtheorem{definition}[theoreme]{Définition}
```

De plus, les numéros peuvent aussi indiquer, par exemple, le chapitre dans lequel on se trouve :

```
\newtheorem{theoreme}[Théorème][chapter]
```

On peut aussi ajouter des spécifications aux théorèmes, etc., par exemple :

```
\begin{theoreme}[Théorème de M. X]  
L'énoncé du théorème apparaît ici.  
\end{theoreme}
```

Voici le résultat.

Chapitre 2

This is the second chapter

Théorème 2.1 (**Théorème de M. X**) *L'énoncé du théorème apparaît ici.*

Définition 2.2 *Ici on définit quelque chose.*

Théorème 2.3 *Un deuxième résultat.*

Exemple 1 *Ici un exemple.*

Remarque 1 *Ici une remarque.*

Les environnements « théorème » et « définition » (5/8)

Problème : le style de chaque environnement est le même.

Mais normalement, on distingue

Les environnements « théorème » et « définition » (5/8)

Problème : le style de chaque environnement est le même.

Mais normalement, on distingue

- les environnements avec le texte en italique : théorèmes, propositions, etc.,

• les environnements avec le texte en style droit : définitions, remarques, etc.

Avec le package `amsthm` on peut associer un de trois styles (`plain`, `definition`, `remark`) à chaque environnement (un `*` supprime la numérotation) :

`\newtheorem{theorem}` C'est le style des théorèmes...

`\newtheorem{theorem}{Théorème}[chapter]`

`\newtheorem{corollaire}[theorem]{Corollaire}`

Les environnements « théorème » et « définition » (5/8)

Problème : le style de chaque environnement est le même.

Mais normalement, on distingue

- les environnements avec le texte en italique : théorèmes, propositions, etc.,
- les environnements avec le texte en style droit : définitions, remarques, etc.

Avec le package `amsthm` on peut associer un de trois styles (`plain`, `definition`, `remark`) à chaque environnement (un * supprime la numérotation) :

```
\newtheorem{theorem}{Théorème}[chapter]
\newtheorem{corollaire}[theorem]{Corollaire}
```

C'est le style des théorèmes...

Les environnements « théorème » et « définition » (5/8)

Problème : le style de chaque environnement est le même.

Mais normalement, on distingue

- les environnements avec le texte en italique : théorèmes, propositions, etc.,
- les environnements avec le texte en style droit : définitions, remarques, etc.

Avec le package `amsthm` on peut associer un de trois styles (`plain`, `definition`, `remark`) à chaque environnement (un `*` supprime la numérotation) :

```
\newtheorem{theoreme}{Théorème}[chapter]
\newtheorem{corollaire}[theoreme]{Corollaire}
```

Les environnements « théorème » et « définition » (5/8)

Problème : le style de chaque environnement est le même.

Mais normalement, on distingue

- les environnements avec le texte en italique : théorèmes, propositions, etc.,
- les environnements avec le texte en style droit : définitions, remarques, etc.

Avec le package `amsthm` on peut associer un de trois styles (`plain`, `definition`, `remark`) à chaque environnement (un `*` supprime la numérotation) :

`\theoremstyle{plain}` C'est le style des théorèmes...

`\newtheorem{theoreme}{Théorème}[chapter]`

`\newtheorem{corollaire}[theoreme]{Corollaire}`

Les environnements « théorème » et « définition » (6/8)

`\theoremstyle{definition}` C'est le style des définitions...

`\newtheorem{definition}[theoreme]{Définition}`

`\newtheorem{exemple}{Exemple}`

`\theoremstyle{remark}` C'est le style des remarques...

`\newtheorem{remarque}{Remarque}`

`\newtheorem*{notation}{Notation}`

Les environnements « théorème » et « définition » (6/8)

`\theoremstyle{definition}` C'est le style des définitions...

`\newtheorem{definition}[theoreme]{Définition}`

`\newtheorem{exemple}{Exemple}`

`\theoremstyle{remark}` C'est le style des remarques...

`\newtheorem{remarque}{Remarque}`

`\newtheorem*{notation}{Notation}`

Les environnements « théorème » et « définition » (7/8)

De plus, il existe un environnement pour les preuves.

```
\begin{proof}
```

```
preuve
```

```
\end{proof}
```

Voici le résultat.

Les environnements « théorème » et « définition » (7/8)

De plus, il existe un environnement pour les preuves.

```
\begin{proof}  
preuve  
\end{proof}
```

Voici le résultat.

Les environnements « théorème » et « définition » (7/8)

De plus, il existe un environnement pour les preuves.

```
\begin{proof}  
preuve  
\end{proof}
```

Voici le résultat.

Chapitre 2

This is the second chapter

Définition 2.1. Ici on définit quelque chose.

Théorème 2.2 (Théorème de M. X). *L'énoncé du théorème apparaît ici.*

Démonstration. Voici on fait la preuve. . .



Corollaire 2.3. *Comme conséquence on obtient quelque chose.*

Nous allons remarquer que ce n'est pas difficile à voir que . . .

Notation. Dans tout ce qui suit nous allons noter. . .

Théorème 2.4. *Un deuxième résultat.*

Exemple 1. Ici un exemple.

Remarque 1. Ici une remarque.

La page de garde (1/2)

On peut créer une page de garde.

On y spécifie

`\title{C'est mon titre}`

`\author{C'est moi}`

`\date{28.10.2009}` ou bien `\date{\today}`

Avec la commande suivante on crée la page de garde :

`\maketitle`

Voici le résultat.

La page de garde (1/2)

On peut créer une page de garde.

On y spécifie

`\title{C'est mon titre}`

`\author{C'est moi}`

`\date{28.10.2009}` ou bien `\date{\today}`

Avec la commande suivante on crée la page de garde :

`\maketitle`

Voici le résultat.

La page de garde (1/2)

On peut créer une page de garde.

On y spécifie

```
\title{C'est mon titre}
```

```
\author{C'est moi}
```

```
\date{28.10.2009} ou bien \date{\today}
```

Avec la commande suivante on crée la page de garde :

```
\maketitle
```

Voici le résultat.

La page de garde (1/2)

On peut créer une page de garde.

On y spécifie

```
\title{C'est mon titre}
```

```
\author{C'est moi}
```

```
\date{28.10.2009} ou bien \date{\today}
```

Avec la commande suivante on crée la page de garde :

```
\maketitle
```

Voici le résultat.

La page de garde (1/2)

On peut créer une page de garde.

On y spécifie

```
\title{C'est mon titre}
```

```
\author{C'est moi}
```

```
\date{28.10.2009} ou bien \date{\today}
```

Avec la commande suivante on crée la page de garde :

```
\maketitle
```

Voici le résultat.

La page de garde (1/2)

On peut créer une page de garde.

On y spécifie

```
\title{C'est mon titre}
```

```
\author{C'est moi}
```

```
\date{28.10.2009} ou bien \date{\today}
```

Avec la commande suivante on crée la page de garde :

```
\maketitle
```

Voici le résultat.

La page de garde (1/2)

On peut créer une page de garde.

On y spécifie

```
\title{C'est mon titre}
```

```
\author{C'est moi}
```

```
\date{28.10.2009} ou bien \date{\today}
```

Avec la commande suivante on crée la page de garde :

```
\maketitle
```

Voici le résultat.

La page de garde (1/2)

On peut créer une page de garde.

On y spécifie

```
\title{C'est mon titre}
```

```
\author{C'est moi}
```

```
\date{28.10.2009} ou bien \date{\today}
```

Avec la commande suivante on crée la page de garde :

```
\maketitle
```

Voici le résultat.

C'est mon titre

C'est moi

28 octobre 2009

La table de matières (1/2)

On peut également inclure une table de matières.

Elle est automatiquement créée avec la commande

`\tableofcontents`

Attention : il faut compiler deux fois.

Voici un exemple.

La table de matières (1/2)

On peut également inclure une table de matières.

Elle est automatiquement créée avec la commande

`\tableofcontents`

Attention: il faut compiler deux fois.

Voici un exemple.

La table de matières (1/2)

On peut également inclure une table de matières.

Elle est automatiquement créée avec la commande

```
\tableofcontents
```

Attention : il faut compiler deux fois.

Voici un exemple.

La table de matières (1/2)

On peut également inclure une table de matières.

Elle est automatiquement créée avec la commande

```
\tableofcontents
```

Attention : Il faut compiler deux fois.

Voici un exemple.

La table de matières (1/2)

On peut également inclure une table de matières.

Elle est automatiquement créée avec la commande

```
\tableofcontents
```

Attention : Il faut compiler deux fois.

Voici un exemple.

Table des matières

1	C'est n'importe quoi	3
1.1	La première section	3
1.2	Une deuxième section	4
1.2.1	Une sous-section	4
A	A ne pas oublier	5

La bibliographie (1/2)

A la fin du texte on peut inclure une bibliographie.

Voici les commandes à utiliser :

```
\begin{thebibliography}{9}
\bibitem{Gra00} George Grätzer, \textit{Math into
\LaTeX}, Birkhäuser, Boston 2000
\bibitem{Mar02} Nicolas Markey, \textit{Introduction
\LaTeX}, ENS Cachan 2002
\end{thebibliography}
```

Ici, le 9 donne le nombre maximal de références.

Pour faire référence à une entrée dans la bibliographie on écrit

```
\cite{Gra00} ou \cite[pp. 42–43]{Gra00}
```

Attention : Il faut compiler deux fois.

Voici un exemple (avec la classe article).

La bibliographie (1/2)

A la fin du texte on peut inclure une bibliographie.

Voici les commandes à utiliser :

```
\begin{thebibliography}{9}  
\bibitem{Gra00} George Grätzer, \textit{Math into  
\LaTeX}, Birkhäuser, Boston 2000  
\bibitem{Mar02} Nicolas Markey, \textit{Introduction  
\LaTeX}, ENS Cachan 2002  
\end{thebibliography}
```

Ici, le 9 donne le nombre maximal de références.

Pour faire référence à une entrée dans la bibliographie on écrit

```
\cite{Gra00} ou \cite[pp. 42–43]{Gra00}
```

Attention : Il faut compiler deux fois.

Voici un exemple (avec la classe article).

La bibliographie (1/2)

A la fin du texte on peut inclure une bibliographie.

Voici les commandes à utiliser :

```
\begin{thebibliography}{9}  
\bibitem{Gra00} George Grätzer, \textit{Math into  
\LaTeX}, Birkhäuser, Boston 2000  
\bibitem{Mar02} Nicolas Markey, \textit{Introduction  
\LaTeX}, ENS Cachan 2002  
\end{thebibliography}
```

Ici, le 9 donne le nombre maximal de références.

Pour faire référence à une entrée dans la bibliographie on écrit

```
\cite{Gra00} ou \cite[pp. 42–43]{Gra00}
```

Attention : il faut compiler deux fois.

Voici un exemple (avec la classe article).

La bibliographie (1/2)

A la fin du texte on peut inclure une bibliographie.

Voici les commandes à utiliser :

```
\begin{thebibliography}{9}  
\bibitem{Gra00} George Grätzer, \textit{Math into  
\LaTeX}, Birkhäuser, Boston 2000  
\bibitem{Mar02} Nicolas Markey, \textit{Introduction  
\LaTeX}, ENS Cachan 2002  
\end{thebibliography}
```

Ici, le 9 donne le nombre maximal de références.

Pour faire référence à une entrée dans la bibliographie on écrit

```
\cite{Gra00} ou \cite[pp. 42–43]{Gra00}
```

Attention : il faut compiler deux fois.

Voici un exemple (avec la classe article).

La bibliographie (1/2)

A la fin du texte on peut inclure une bibliographie.

Voici les commandes à utiliser :

```
\begin{thebibliography}{9}  
\bibitem{Gra00} George Grätzer, \textit{Math into  
\LaTeX}, Birkhäuser, Boston 2000  
\bibitem{Mar02} Nicolas Markey, \textit{Introduction  
\LaTeX}, ENS Cachan 2002  
\end{thebibliography}
```

Ici, le 9 donne le nombre maximal de références.

Pour faire référence à une entrée dans la bibliographie on écrit

```
\cite{Gra00} ou \cite[pp. 42–43]{Gra00}
```

Attention : il faut compiler deux fois.

Voici un exemple (avec la classe article).

La bibliographie (1/2)

A la fin du texte on peut inclure une bibliographie.

Voici les commandes à utiliser :

```
\begin{thebibliography}{9}  
\bibitem{Gra00} George Grätzer, \textit{Math into  
\LaTeX}, Birkhäuser, Boston 2000  
\bibitem{Mar02} Nicolas Markey, \textit{Introduction  
\LaTeX}, ENS Cachan 2002  
\end{thebibliography}
```

Ici, le 9 donne le nombre maximal de références.

Pour faire référence à une entrée dans la bibliographie on écrit

```
\cite{Gra00} ou \cite[pp. 42-43]{Gra00}
```

Attention : il faut compiler deux fois.

Voici un exemple (avec la classe article).

La bibliographie (1/2)

A la fin du texte on peut inclure une bibliographie.

Voici les commandes à utiliser :

```
\begin{thebibliography}{9}  
\bibitem{Gra00} George Grätzer, \textit{Math into  
\LaTeX}, Birkhäuser, Boston 2000  
\bibitem{Mar02} Nicolas Markey, \textit{Introduction  
\LaTeX}, ENS Cachan 2002  
\end{thebibliography}
```

Ici, le 9 donne le nombre maximal de références.

Pour faire référence à une entrée dans la bibliographie on écrit

```
\cite{Gra00} ou \cite[pp. 42-43]{Gra00}
```

Attention : Il faut compiler deux fois.

Voici un exemple (avec la classe article).

La bibliographie (1/2)

A la fin du texte on peut inclure une bibliographie.

Voici les commandes à utiliser :

```
\begin{thebibliography}{9}  
\bibitem{Gra00} George Grätzer, \textit{Math into  
\LaTeX}, Birkhäuser, Boston 2000  
\bibitem{Mar02} Nicolas Markey, \textit{Introduction  
\LaTeX}, ENS Cachan 2002  
\end{thebibliography}
```

Ici, le 9 donne le nombre maximal de références.

Pour faire référence à une entrée dans la bibliographie on écrit

```
\cite{Gra00} ou \cite[pp. 42-43]{Gra00}
```

Attention : Il faut compiler deux fois.

Voici un exemple (avec la classe `article`).

La bibliographie (2/2)

Pour plus d'informations sur les bibliographies on peut regarder [1, pp. 42–43] ou bien les pages 19–20 de Markey [2].

Références

- [1] George Grätzer, *Math into L^AT_EX*, Birkhäuser, Boston 2000
- [2] Nicolas Markey, *Introduction à L^AT_EX*, ENS Cachan 2002

Diverses choses

Avec la commande

```
\footnote{Je veux signaler quelque chose}
```

on peut faire des notes de bas de page.

Mais il faut les éviter dans un mémoire en mathématiques.

On peut aussi faire un index. Mais c'est rarement utilisé dans un mémoire.

Dans la classe `article`, il y a la possibilité d'inclure un résumé, par exemple avant le texte, en mettant

```
\begin{abstract}  
...  
\end{abstract}
```

Diverses choses

Avec la commande

```
\footnote{Je veux signaler quelque chose}
```

on peut faire des notes de bas de page.

Mais il faut les *éviter* dans un mémoire en mathématiques.

On peut aussi faire un index. Mais c'est rarement utilisé dans un mémoire.

Dans la classe `article`, il y a la possibilité d'inclure un résumé, par exemple avant le texte, en mettant

```
\begin{abstract}  
...  
\end{abstract}
```

Diverses choses

Avec la commande

```
\footnote{Je veux signaler quelque chose}
```

on peut faire des notes de bas de page.

Mais il faut les *éviter* dans un mémoire en mathématiques.

On peut aussi faire un index. Mais c'est rarement utilisé dans un mémoire.

Dans la classe `article`, il y a la possibilité d'inclure un résumé, par exemple avant le texte, en mettant

```
\begin{abstract}  
...  
\end{abstract}
```

Diverses choses

Avec la commande

```
\footnote{Je veux signaler quelque chose}
```

on peut faire des notes de bas de page.

Mais il faut les *éviter* dans un mémoire en mathématiques.

On peut aussi faire un index. Mais c'est rarement utilisé dans un mémoire.

Dans la classe `article`, il y a la possibilité d'inclure un résumé, par exemple avant le texte, en mettant

```
\begin{abstract}
```

```
...
```

```
\end{abstract}
```

L'espacement

Parfois il est quand même nécessaire d'intervenir, par exemple pour augmenter un espace. Voici quelques commandes :

<code>\quad</code>	un espace normal
<code>\qquad</code>	un espace un peu plus petit
<code>\: </code>	un espace encore un peu plus petit
<code>\, </code>	un espace encore un peu plus petit
<code>\! </code>	un espace négatif
<code>\quad</code>	un espace, largeur de 'm'
<code>\qquad</code>	un espace, largeur de 'mm'
<code>\hspace*{ 30mm}</code>	un espace de longueur 30mm
<code>\vspace*{ 30mm}</code>	un espace vertical de hauteur 30mm
<code>\hfill</code>	remplit l'espace restant sur la ligne courante
<code></code>	un espace de longueur du mot 'abcd'

Surtout les commandes `\hspace*` et `\vspace*` sont à éviter

L'espacement

Parfois il est quand même nécessaire d'intervenir, par exemple pour augmenter un espace. Voici quelques commandes :

<code>\quad</code>	un espace normal
<code>\; </code>	un espace un peu plus petit
<code>\: </code>	un espace encore un peu plus petit
<code>\, </code>	un espace encore un peu plus petit
<code>\!</code>	un espace négatif
<code>\quad</code>	un espace, largeur de 'm'
<code>\qquad</code>	un espace, largeur de 'mm'
<code>\hspace*{30mm}</code>	un espace de longueur 30mm
<code>\vspace*{30mm}</code>	un espace vertical de hauteur 30mm
<code>\hfill</code>	remplit l'espace restant sur la ligne courante
<code></code>	un espace de longueur du mot 'abcd'

Surtout les commandes `\hspace*` et `\vspace*` sont à éviter

L'espacement

Parfois il est quand même nécessaire d'intervenir, par exemple pour augmenter un espace. Voici quelques commandes :

<code>\quad</code>	un espace normal
<code>\; </code>	un espace un peu plus petit
<code>\: </code>	un espace encore un peu plus petit
<code>\, </code>	un espace encore un peu plus petit
<code>\!</code>	un espace négatif
<code>\quad</code>	un espace, largeur de 'm'
<code>\qqquad</code>	un espace, largeur de 'mm'
<code>\hspace*{30mm}</code>	un espace de longueur 30mm
<code>\vspace*{30mm}</code>	un espace vertical de hauteur 30mm
<code>\hfill</code>	remplit l'espace restant sur la ligne courante
<code></code>	un espace de longueur du mot 'abcd'

Surtout les commandes `\hspace*` et `\vspace*` sont à éviter.  Université de Mons

La mise en page

De même, on veut parfois changer les dimensions du texte, la longueur de l'alinéa, etc.

Il faut placer ces commandes dans le préambule. Elles se trouvent normalement au début du préambule.

Voici quelques commandes :

<code>\textwidth470pt</code>	largeur du texte (1 pt= 0,188 mm)
<code>\textheight650pt</code>	hauteur du texte
<code>\topmargin0pt</code>	largeur de la marge en haut de page
<code>\oddsidemargin0pt</code>	largeur de la marge, pages impaires
<code>\evensidemargin0pt</code>	largeur de la marge, pages paires
<code>\parindent5em</code>	longueur de l'alinéa d'un paragraphe
<code>\parskip2ex</code>	espace vertical entre deux paragraphes

Le package `geometry` permet également la définition de paramètres de mise en page.

La mise en page

De même, on veut parfois changer les dimensions du texte, la longueur de l'alinéa, etc.

Il faut placer ces commandes dans le préambule. Elles se trouvent normalement au début du préambule.

Voici quelques commandes :

<code>\textwidth470pt</code>	largeur du texte (1 pt= 0,188 mm)
<code>\textheight650pt</code>	hauteur du texte
<code>\topmargin0pt</code>	largeur de la marge en haut de page
<code>\oddsidemargin0pt</code>	largeur de la marge, pages impaires
<code>\evensidemargin0pt</code>	largeur de la marge, pages paires
<code>\parindent5em</code>	longueur de l'alinéa d'un paragraphe
<code>\parskip2ex</code>	espace vertical entre deux paragraphes

Le package `geometry` permet également la définition de paramètres de mise en page.

La mise en page

De même, on veut parfois changer les dimensions du texte, la longueur de l'alinéa, etc.

Il faut placer ces commandes dans le préambule. Elles se trouvent normalement au début du préambule.

Voici quelques commandes :

<code>\textwidth470pt</code>	largeur du texte (1 pt= 0,188 mm)
<code>\textheight650pt</code>	hauteur du texte
<code>\topmargin0pt</code>	largeur de la marge en haut de page
<code>\oddsidemargin0pt</code>	largeur de la marge, pages impaires
<code>\evensidemargin0pt</code>	largeur de la marge, pages paires
<code>\parindent5em</code>	longueur de l'alinéa d'un paragraphe
<code>\parskip2ex</code>	espace vertical entre deux paragraphes

Le package `geometry` permet également la définition de paramètres de mise en page.

La mise en page

De même, on veut parfois changer les dimensions du texte, la longueur de l'alinéa, etc.

Il faut placer ces commandes dans le préambule. Elles se trouvent normalement au début du préambule.

Voici quelques commandes :

<code>\textwidth470pt</code>	largeur du texte (1 pt= 0,188 mm)
<code>\textheight650pt</code>	hauteur du texte
<code>\topmargin0pt</code>	largeur de la marge en haut de page
<code>\oddsidemargin0pt</code>	largeur de la marge, pages impaires
<code>\evensidemargin0pt</code>	largeur de la marge, pages paires
<code>\parindent5em</code>	longueur de l'alinéa d'un paragraphe
<code>\parskip2ex</code>	espace vertical entre deux paragraphes

Le package `geometry` permet également la définition de paramètres de mise en page.

Conclusion

Finalement, en \LaTeX on peut tout modifier (ou presque) :

- en utilisant les options des commandes et des environnements
- en utilisant des packages convenables
- en utilisant (ou en créant !) d'autres classes.

Conclusion

Finalement, en \LaTeX on peut tout modifier (ou presque) :

- en utilisant les options des commandes et des environnements
- en utilisant des packages convenables
- en utilisant (ou en créant !) d'autres classes.

Conclusion

Finalement, en \LaTeX on peut tout modifier (ou presque) :

- en utilisant les options des commandes et des environnements
- en utilisant des packages convenables
- en utilisant (ou en créant !) d'autres classes.

Conclusion

Finalement, en \LaTeX on peut tout modifier (ou presque) :

- en utilisant les options des commandes et des environnements
- en utilisant des packages convenables
- en utilisant (ou en créant !) d'autres classes.