

Formation \LaTeX — Niveau intermédiaire

Jérôme Azé

Orsay

26 mars 2013

<http://www.lri.fr/~aze/FormationLaTeX>



Introduction

Table des matières



Introduction

Table des matières



Mes prédécesseurs

- Vincent Fevrier : <http://www.math.u-psud.fr/~fevrier/enseignement/2008/CIES/>
- Bernardo da Costa : <http://www.math.u-psud.fr/~bernardofpc/latex.php>



Introduction

Table des matières



Éditeur, ...

- TexMaker : installé sur toutes les machines
- <http://www.xmlmath.net/texmaker/>
- Multiplateformes, gratuit, éditeur, compilateur et visualisateur intégré



Introduction

Table des matières

Quelques exemples

Quelques formules

$$\sum_{i=1}^n \frac{1 + e^{-\alpha x}}{2x}$$

$$f(x) = \begin{cases} 4 \times \ln(x) & \text{if } x \leq \alpha \\ -\frac{2}{5} \ln(1+x) & \text{if } x \geq \theta \\ 0 & \text{else} \end{cases}$$

Quelques exemples

Quelques formules

$$\sum_{i=1}^n \frac{1 + e^{-\alpha x}}{2x}$$

$$f(x) = \begin{cases} 4 \times \ln(x) & \text{if } x \leq \alpha \\ -\frac{2}{5} \ln(1+x) & \text{if } x \geq \theta \\ 0 & \text{else} \end{cases}$$

Le code associé

```
\[
\underbrace{\overset{n}{\sum}} \frac{1+e^{-\alpha x}}{2 x}
\]

\[
f(x) = \left\{
\begin{array}{ll}
4 \times \ln(x) & \text{\mbox{if } x \le \alpha} \\
-\frac{2}{5} \ln(1+x) & \text{\mbox{if } x \ge \theta} \\
0 & \text{\mbox{else}}
\end{array}
\right.
\]
```

Quelques exemples

Quelques tableaux

	α	β
A	0.2	1.4
B	1	5
C	0.002	5.6

Quelques tableaux

	α	β
A	0.2	1.4
B	1	5
C	0.002	5.6

Le code associé

```
\begin{tabular}{|c|l|r|}  
  \cline{2-3}  
  \multicolumn{1}{c|}{} & $\alpha$ & $\beta$\\  
  \hline  
  $A$ & 0.2 & 1.4\\  
  $B$ & 1 & 5\\  
  $C$ & 0.002 & 5.6\\  
  \hline  
  \end{tabular}
```

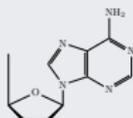
Un peu de chimie

The ribose template

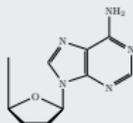


In deoxyribose, substituent r2 is replaced by nothing

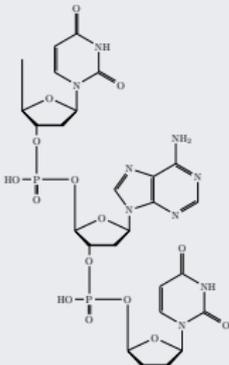




Deoxyadenosine is derived from deoxyribose by substituting r1 with adenine



Local redefinition of the r1 substituent in deoxyribose works, too



Using the previously derived dU and dA templates, we can now easily write a DNA molecule

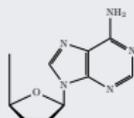
Un peu de chimie



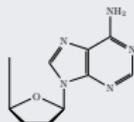
The ribose template



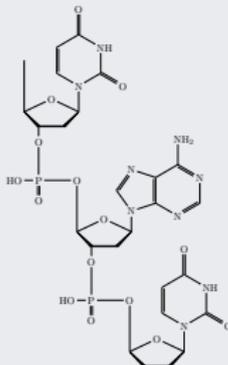
In deoxyribose, substituent rt2 is replaced by nothing



Deoxycytosine is derived from deoxyribose by substituting rt1 with adenine



Local redefinition of the rt1 substituent in deoxyribose works, too

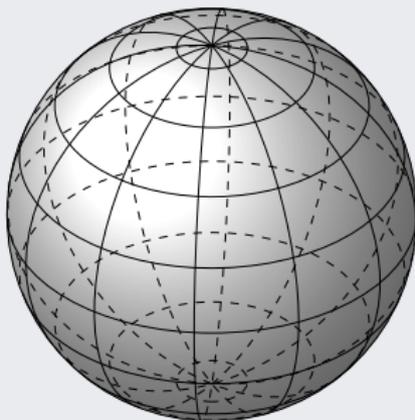


Using the previously derived dU and dA templates, we can now easily write a DNA molecule

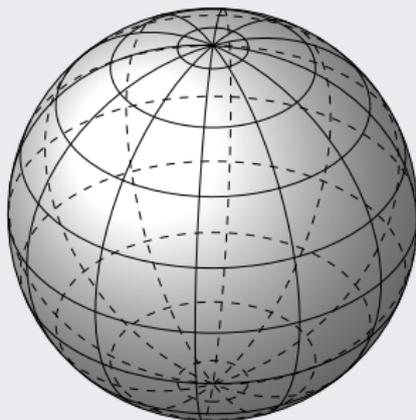
Le code associé

<http://tex.stackexchange.com/questions/40695/chemfig-and-lego-style-molecule-construction/41051#41051>

Un peu de 3D



Un peu de 3D



Le code associé

<http://www.texample.net/tikz/examples/map-projections/>



Quelques exemples

Et aussi

Ces slides



Quelques exemples

Et aussi

Ces slides

Le code associé

```
\begin{frame}
  \frametitle{Quelques exemples}
  \begin{block}{Et aussi}
    Ces slides
  \end{block}
  \pause

  \begin{block}{Le code associ\'e}
    \verbatiminput{exemple-slides.tex}
  \end{block}

\end{frame}
```



Table des matières

Exemple minimaliste



Fichier Intermediaire/code1.tex

```
\documentclass[francais]{letter}

\usepackage[francais]{babel}

\address{...\\...\\...}
\signature{...}

\begin{document}
  \begin{letter}{M...\\...\\...}
    \opening{Cher ...}
    ...
    \closing{Bien cordialement}
  \end{letter}
  \begin{letter}{M...\\...\\...}
    \opening{Cher ...}
    ...
    \closing{Bien cordialement}
  \end{letter}
\end{document}
```



Commandes supplémentaires à utiliser juste après

`\closing` :

- `\cc{...}` s'il y a des destinataires en copie ;
- `\encl{...}` s'il y a des pièces jointes ;
- `\ps{PS: ...}` s'il y a un post-scriptum (ou autre).





La classe `letter`

Elle fonctionne quasiment comme `letter`, avec quelques commandes supplémentaires, et respecte mieux les conventions françaises.



La classe `lettre`

Elle fonctionne quasiment comme `letter`, avec quelques commandes supplémentaires, et respecte mieux les conventions françaises.

On peut définir des paramètres par défaut pour toutes ses lettres.

À mettre dans un fichier `default.ins` ou juste avant `\opening` dans chaque lettre



Fichier Intermediaire/code2.tex

```
%Nom et signature de l'expéditeur
\name{...}
\signature{...}
%Coordonnées de l'expéditeur
\address{...}
\location{...}
\telephone{...}% ou \notelephone
\fax{...}% ou \nofax
\email{...}% ou \noemail
%lieu d'expédition des lettres
\lieu{...}
```



Commandes à utiliser juste avant `\opening` :

- `\conc{...}` pour définir l'objet de la lettre.



La classe `lettre`

Commandes à utiliser juste avant `\opening` :

- `\conc{...}` pour définir l'objet de la lettre.

Commandes à utiliser juste après `\closing` :

- `\cc{...}` s'il y a des destinataires en copie ;
- `\encl{...}` s'il y a des pièces jointes ;
- `\ps{PS:}{...}` s'il y a un post-scriptum.



La classe `letter`

Commandes à utiliser juste avant `\opening` :

- `\conc{...}` pour définir l'objet de la lettre.

Commandes à utiliser juste après `\closing` :

- `\cc{...}` s'il y a des destinataires en copie ;
- `\encl{...}` s'il y a des pièces jointes ;
- `\ps{PS:}{...}` s'il y a un post-scriptum.

Pour faire un télécopie il suffit de remplacer

`\begin{letter}{destinataire}`

par

`\begin{telex}{numéro}{destinataire}`





- écrire une (courte!) lettre adressée à la personne de votre choix ;



- écrire une (courte!) lettre adressée à la personne de votre choix ;
- la transformer en un téléfax.



Introduction

Table des matières



Un exemple de CV

Sur la page web <https://www.lri.fr/~aze/formationLaTeX/docs>, vous trouverez un package : `CV.tar.gz` permettant d'écrire un CV en \LaTeX



- écrire rapidement le début de votre CV





Quelques packages prêts à l'emploi

- http://olivier.commowick.org/thesis_template.php
- <http://www.loria.fr/~roegel/TeX/TL.html>



Compléments au niveau débutant

Table des matières



Compléments au niveau débutant

Table des matières

Règles de ponctuation automatique

Code



Fichier Intermediaire/code3.tex

L'espace entre un point terminant une phrase et le début de la suivante est plus longue que l'espace séparant deux mots.

Pour déterminer si un point termine une phrase ou non, `\TeX` utilise une règle très simple: un point (suivi d'un espace) termine une phrase sauf s'il suit une lettre majuscule. C'est le cas par exemple des initiales et sigles : L. Euler.

Sortie

L'espace entre un point terminant une phrase et le début de la suivante est plus longue que l'espace séparant deux mots. Pour déterminer si un point termine une phrase ou non, `TeX` utilise une règle très simple : un point (suivi d'un espace) termine une phrase sauf s'il suit une lettre majuscule. C'est le cas par exemple des initiales et sigles : L. Euler.

Points ne terminant pas une phrase

Code



Fichier Intermediaire/code4.tex

```
Pour indiquer qu'un point ne
termine pas une phrase, comme
pour etc.\ on utilise une espace
forcée juste après le point.
Si on ne le met pas, on obtient
etc. qui est un peu plus espacée.
```

Sortie

Pour indiquer qu'un point ne termine pas une phrase, comme pour etc. on utilise une espace forcée juste après le point. Si on ne le met pas, on obtient etc. qui est un peu plus espacée.

Points terminant une phrase

Code



Fichier Intermediaire/code5.tex

Pour indiquer qu'un point qui suit
une majuscule termine une phrase
, on met `\verb|\@|` JUSTE AVANT LE
POINT\@.

Sortie

Pour indiquer qu'un point qui
suit une majuscule termine
une phrase, on met `\@` JUSTE
AVANT LE POINT.

We see better in English

Les conventions d'espacement en anglais donnent beaucoup plus de différence : les espaces après des points de fin de phrase sont plus grandes, et peuvent s'« allonger » 3 fois plus par rapport aux espaces entre deux mots.

Code



Fichier Intermediaire/code6.tex

```
\selectlanguage{english}  
\ldots at the NATO. But also it is  
possible \ldots  
  
\ldots at the NATO\@. But also it  
is possible \ldots
```

Sortie

... at the NATO. But also it is possible ...

... at the NATO. But also it is possible ...

Cet exemple a été obtenu avec un peu de chance quand même !

Points de suspension

Code



Fichier Intermediaire/code7.tex

Dans le texte, au lieu d'utiliser ... (qui introduit des espaces incorrectes entre les points), on utilise plutôt la commande `\verb|\ldots|`, qui fonctionne aussi en mode mathématique : `\[a_1,a_2,\ldots,a_n]`

Sortie

Dans le texte, au lieu d'utiliser... (qui introduit des espaces incorrectes entre les points), on utilise plutôt la commande `\ldots`, qui fonctionne aussi en mode mathématique :

$$a_1, a_2, \dots, a_n$$

Comme toute commande qui fait du texte, il est parfois nécessaire d'utiliser une espace forcée "`\` " après.



Compléments au niveau débutant

Table des matières

Schéma de la compilation

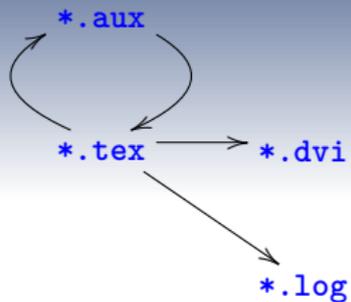
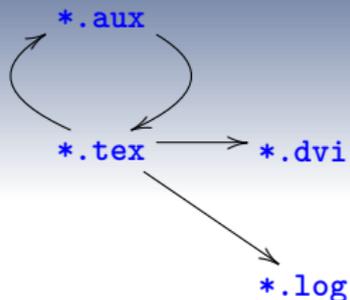
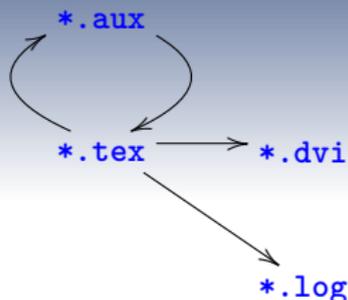


Schéma de la compilation



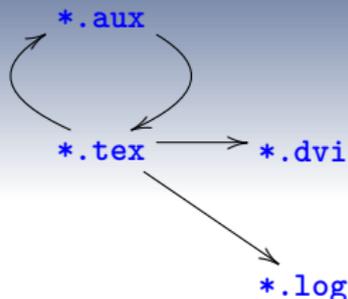
Le fichier `.aux` contient (entre autres) des informations pour les étiquettes.

Schéma de la compilation



Le fichier `.aux` contient (entre autres) des informations pour les étiquettes. Il faut parfois plusieurs compilations successives avant que les changements soient appliqués au fichier auxiliaire, puis transmis au document en entier. Dans plusieurs cas (surtout pour les références), \LaTeX avertit quand il est utile de compiler une fois de plus.

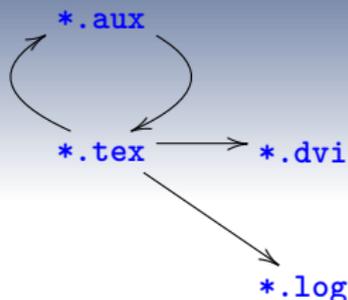
Schéma de la compilation



Le fichier `.aux` contient (entre autres) des informations pour les étiquettes. Il faut parfois plusieurs compilations successives avant que les changements soient appliqués au fichier auxiliaire, puis transmis au document en entier. Dans plusieurs cas (surtout pour les références), \LaTeX avertit quand il est utile de compiler une fois de plus.

Le fichier `.log` contient des informations sur le déroulement de la compilation (en particulier les *bad boxes*).

Schéma de la compilation



Le fichier `.aux` contient (entre autres) des informations pour les étiquettes. Il faut parfois plusieurs compilations successives avant que les changements soient appliqués au fichier auxiliaire, puis transmis au document en entier. Dans plusieurs cas (surtout pour les références), \LaTeX avertit quand il est utile de compiler une fois de plus.

Le fichier `.log` contient des informations sur le déroulement de la compilation (en particulier les *bad boxes*).

Certaines extensions (`minitoc` par exemple) utilisent d'autres fichiers auxiliaires encore.



Personnaliser la mise en page

Table des matières

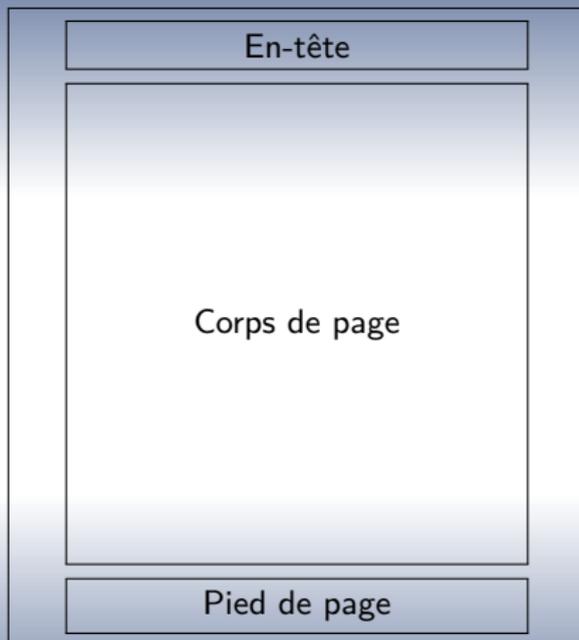


Personnaliser la mise en page

Table des matières



Structure d'une page





Foliation

La commande `\pagenumbering{...}` permet de spécifier cinq types de numérotation des pages :

- `arabic` pour la numérotation en chiffres arabes standard ;
- `roman` pour des chiffres romains minuscules ;
- `Roman` pour des chiffres romains majuscules ;
- `alph` pour des lettres minuscules ;
- `Alph` pour des lettres majuscules.



Dans le préambule :

- `\title{...}` définit le titre ;
- `\author{...}` définit le(s) auteur(s) ;
- `\date{...}` définit la date ;
- `\thanks{...}` produit une note pour le titre.

Certaines classes peuvent prendre l'option `titlepage` (avec `\documentclass`). Dans ce cas la commande `\maketitle` affichera le titre tout seul sur une page entière.



Styles de page

Il y a quatre styles en tout :

- `empty` : ni en-tête, ni pied de page ;
- `plain` : juste un numéro de page en bas (style par défaut) ;
- `headings` : affichage des en-têtes (leur contenu dépend de la classe, par exemple le titre de la section en cours à gauche et celui de la sous-section à droite) ;
- `myheadings` : affiche le contenu donné par `\markboth` et `\markright`.

On change de style pour tout le reste document avec `\pagestyle{...}` et on peut changer le style d'une page individuelle avec (par exemple)

`\thispagestyle{empty}`.

Important : si on veut mettre un style particulier à la page de titre, il faut mettre `\thispagestyle{...}` juste après `\maketitle`, **sans espace entre les deux**.



Styles de page

Il y a quatre styles en tout :

- `empty` : ni en-tête, ni pied de page ;
- `plain` : juste un numéro de page en bas (style par défaut) ;
- `headings` : affichage des en-têtes (leur contenu dépend de la classe, par exemple le titre de la section en cours à gauche et celui de la sous-section à droite) ;
- `myheadings` : affiche le contenu donné par `\markboth` et `\markright`.

On change de style pour tout le reste document avec `\pagestyle{...}` et on peut changer le style d'une page individuelle avec (par exemple)

`\thispagestyle{empty}`.

Important : si on veut mettre un style particulier à la page de titre, il faut mettre `\thispagestyle{...}` juste après `\maketitle`, **sans espace entre les deux**.

Certaines extensions (`fancyhdr`) définissent des styles supplémentaires.



Marques à gauche et à droite

Lorsque le style de page est `myheadings`, les commandes `\markboth{left head}{right head}` et `\markright{right head}` définissent les en-têtes en fonction de la parité de la page.

L'en-tête de gauche est définie par la dernière commande `\markboth` rencontrée, celle de droite par la première commande `\markboth` ou `\markright` rencontrée sur cette page s'il y en a une (ou sinon la dernière donnée avant la page).



Personnaliser la mise en page

Exercice



Solution

Code



Fichier Intermediaire/code8.tex

```
\newpage  
\thispagestyle{empty}  
\null %on peut aussi utiliser \strut ou ~  
\newpage
```



Personnaliser la mise en page

Table des matières

Unités de mesure fixes

L^AT_EX reconnaît le centimètre (`cm`), le millimètre (`mm`), le pouce (`in`), le point (`pt`), le pica (`pc`), le point réduit (`sp`), le gros point (`bp`), le point didot (`dd`) et le cicéro (`cc`).

Code



Fichier Intermediaire/code9.tex

```
\rule{2cm}{0.1pt}  
  
\rule{30mm}{1pt}  
  
\rule{1.5in}{3pt}  
  
\rule{130pt}{10pt}
```

Sortie





Correspondance des unités

Unité	Valeur en pt	Valeur en cm
cm	28.45	
mm	2.85	0.1
in	72.27	2.54
pt		0.0351
pc	12	0.421
sp	1/65536	
bp	1.004	0.0353
dd	1.07	0.0376
cc	12.84	0.451

Unités de mesure relatives

Ces unités dépendent de la police en cours. \LaTeX reconnaît la *largeur de la lettre « M »* (`em`, utilisée pour les espaces horizontales) et la *hauteur de la lettre « x »* (`ex`, utilisée pour les espaces verticales).

Code



Fichier Intermediaire/code10.tex

```

\tiny Voici un rectangle \rule{1em
}{1ex}

\normalsize Voici un rectangle \
rule{1em}{1ex}

\huge Voici un rectangle \rule{1em
}{1ex}

```

Sortie

 Voici un rectangle 

 Voici un rectangle 

Voici un rectangle



Saut de ligne de hauteur variable

Code



Fichier Intermediaire/code11.tex

```
Je vais sauter 1cm\\[1cm]
puis un saut de ligne normal\\[0pt
]
puis je saute une hauteur négative
\\[-2ex]
du coup le texte se chevauche\
ldots
```

Sortie

Je vais sauter 1cm

puis un saut de ligne normal
puis je saute une hauteur né-
gative
du coup le texte se che-
vauche...



Personnaliser la mise en page

Table des matières

Commandes de longueur

Certains noms de commandes correspondent à des longueurs. On peut leur affecter une valeur avec `\setlength`. Par exemple :

- `\baselineskip` contrôle l'interligne ;
- `\parindent` contrôle l'indentation des paragraphes ;
- `\parskip` contrôle l'espace vertical entre paragraphes.

Code



Fichier Intermediaire/code12.tex

```
\setlength\baselineskip{3ex}  
\setlength\parindent{1cm}  
\setlength\parskip{2ex}
```

Le premier paragraphe qui démontre
l'effet du changement des
longueurs sur la mise en page.

Voici un second paragraphe pour
voir l'effet du changement des
longueurs sur la mise en page.

Sortie

Le premier paragraphe
qui démontre l'effet du change-
ment des longueurs sur la mise
en page.

Voici un second para-
graphe pour voir l'effet du
changement des longueurs sur
la mise en page.

Arguments de longueur

Certaines commandes prennent un argument qui est une valeur de longueur (par exemple `\rule`, `\rule` ou encore `\setlength`). On peut lui donner comme valeur :

- une valeur explicite comme on l'a déjà vu ;
- la valeur d'une autre longueur ;
- la valeur d'une autre longueur multipliée par un facteur (par exemple `0.5\baselineskip`).

Code



Fichier Intermediaire/code13.tex

```
Un trait qui fait la moitié de la
  largeur disponible et une ligne
  de hauteur:\
\rule{0.5\textwidth}{\baselineskip
}
```

Sortie

Un trait qui fait la moitié de la
largeur disponible et une ligne
de hauteur :



Définition de nouvelles longueurs

- `\newlength` : définit une nouvelle commande de longueur ;
- `\setlength` : affecte une valeur à une longueur ;
- `\addtolength` : ajoute une valeur à une longueur ;

Code



Fichier Intermediaire/code14.tex

```
\newlength\testlength
\rule{\testlength}{1ex}

\setlength\testlength{2cm}
\rule{\testlength}{1ex}

\addtolength\testlength{1cm}
\rule{\testlength}{1ex}

\addtolength\testlength{-0.5\
testlength}
\rule{\testlength}{1ex}
```

Sortie



La valeur d'une longueur à sa création est zéro.



Longueurs de mise en page globale

Certaines longueurs définissent la mise en page en fonction du format de papier et de la classe. On peut citer :

- `\oddsidemargin` et `\evensidemargin` correspondent respectivement aux marges de gauche des pages paires ou impaires (entre le bord du papier et le texte) ;
- `\textwidth` détermine la largeur de texte sur la page (ou `\columnwidth` dans le cas de textes avec plusieurs colonnes) ;
- `\topmargin` règle la distance verticale entre le bord du papier et l'en-tête ;
- `\headheight` règle la hauteur de l'en-tête ;
- `\headsep` définit la distance verticale entre l'en-tête et le corps de la page ;
- `\textheight` détermine la hauteur de texte sur le corps de la page ;
- `\footskip` définit la distance verticale entre le corps et le pied de page ;
- `\footheight` règle la hauteur du pied de page.

Longueurs associées aux tableaux

- `\tabcolsep` ou `\arraycolsep` (en fonction de l'environnement) : moitié de l'espace horizontal placé entre les colonnes du tableau ;
- `\arrayrulewidth` : largeur des filets créés par `|`, `\hline`, `\cline` ou `\vline` ;
- `\doublerulesep` : espacement entre deux filets successifs créés par `||` ou `\hline\hline`.

Code



Fichier Intermediaire/code15.tex

```

\setlength\tabcolsep{0.1in}
\setlength\arrayrulewidth{3pt}
\begin{tabular}{r|c|l}
Un&tableau&avec\\
\hline
des&longueurs&spéciales
\end{tabular}
    
```

Sortie

Un	tableau	avec
des	longueurs	spéciales



Longueurs associées au texte



Longueurs associées au texte



La ligne bleue est la ligne de base

Longueurs associées au texte



La ligne bleue est la ligne de base
Chaque caractère possède sa propre boîte

Longueurs associées au texte



La ligne bleue est la ligne de base
Chaque caractère possède sa propre boîte
Chaque mot possède sa propre boîte
etc. . .

Changer des longueurs en fonction du contenu

- `\settowidth` : permet de stocker la largeur du texte (avec la police et la taille courante) dans une longueur ;
- `\settoheight` : permet de stocker la hauteur du texte (au-dessus de la ligne de base) dans une longueur ;
- `\settodepth` : permet de stocker la profondeur du texte (au-dessous de la ligne de base) dans une longueur ;

Code



Fichier Intermediaire/code16.tex

```
\newlength\longueur  
\settowidth\longueur{barrer}  
Exemple: on peut \makebox[0pt][1  
]{\rule[0.4ex]{\longueur}{1pt}}  
barrer du texte grâce à ça.
```

Sortie

Exemple : on peut ~~barrer~~ du
texte grâce à ça.



Personnaliser la mise en page

Table des matières

Mode LR

Le texte va occuper une seule ligne (donc aucun paragraphe ou de saut de ligne n'est autorisé). Il peut dépasser des limites de la boîte.

- `\makebox[length][pos]{texte}` : affiche **texte** dans une boîte LR de longueur **length**. L'argument optionnel **pos** contrôle la position du texte dans la boîte (**c** par défaut pour centré, **l** pour à gauche et **r** pour à droite). Commande utile pour faire croire à L^AT_EX qu'un mot est d'une longueur différente.
- `\mbox{texte}` : affiche **texte** dans une boîte LR dont la longueur s'adapte automatiquement au contenu. Utilisé pour éviter un passage à la ligne par exemple.

Code



Fichier Intermediaire/code17.tex

```
\makebox[3cm][r]{Ce texte va être
décalé à gauche}

\mbox{Cette ligne ne sera pas
coupée même si elle est trop
longue}
% Mais elle fera une "Overfull
box" à la compilation
```

Sortie

Ce texte va être décalé à gauche

Cette ligne ne sera pas coupée même

Mode paragraphe

Ces boîtes permettent d'écrire du texte d'une taille fixée avec des sauts de ligne. l'indentation des paragraphes (`\parindent`) est remise à zéro automatiquement, on peut toutefois la changer.

- `\parbox[pos]{width}{texte}` : affiche `texte` dans une boîte de largeur `width`, l'argument optionnel `pos` permet de préciser l'alignement vertical par rapport à la ligne de texte courante. Par défaut, c'est `m` (centré) on peut utiliser `t` (pour aligner le haut avec la ligne courante) et `b` (pour aligner le bas avec la ligne courante).
- `\begin{minipage}[pos]{width} ...texte... \end{minipage}` : même usage, il est en outre possible d'utiliser des notes de bas de paragraphe.

Code



Fichier Intermediaire/code18.tex

```
\begin{minipage}{0.6\textwidth}\  
  paperwidth=1.1764705882\textwidth  
\setlength\parindent{1cm}  
Voici un paragraphe\footnote{qui  
  possède une indentation de 1\,cm  
  .} dans une minipage.  
\end{minipage}
```

Sortie

Voici un
paragraphe^a dans
une minipage.

a. qui possède une
indentation de 1 cm.

Boîtes dans les tableaux

Le spécificateur de colonne **p** prend en argument une longueur : il crée dans chaque cellule une boîte en mode paragraphe de la largeur souhaitée, alignée sur la ligne supérieure. Les spécificateurs **m** et **b** (définis par l'extension **array**) fonctionnent de manière identique mais imposent un alignement vertical respectivement centré ou sur la ligne inférieure.

Code



Fichier Intermediaire/code19.tex

```
\begin{tabular}{p{1cm}|p{3cm}}
Dans ce&tableau\\
\hline
toutes les colonnes&sont de
  largeur fixée, ce qui occasionne
  des passages à la ligne\ldots
\end{tabular}
```

Sortie

Dans ce	tableau
toutes les co- lonnes	sont de largeur fixée, ce qui occasionne des passages à la ligne...

Boîtes décoratives

- `\framebox[length][pos]{texte}` : affiche `texte` dans un cadre, les arguments `length` et `pos` ont la même fonction que pour `\makebox`. Les longueurs `\fboxsep` et `\fboxrule` contrôlent respectivement la distance du contenu au cadre et la largeur du trait du cadre ;
- `\fbox{texte}` : version de `framebox` analogue à `mbox` : la longueur est calculée automatiquement.
- `\rule[raise]{width}{height}` : affiche un rectangle noir de taille `width`×`height`. L'option `raise` permet de donner un décalage vertical par rapport à la ligne de base ;
- l'extension `fancybox` introduit des boîtes décoratives supplémentaires s'utilisant comme `\fbox` (`\shadowbox`, `\doublebox`, `\ovalbox`, etc).

Code



Fichier Intermediaire/code20.tex

```
Le texte \fbox{encadré} c'est \
Ovalbox{joli} mais pas très \
shadowbox{lisible}.
```

Sortie

Le texte encadré c'est joli
mais pas très lisible.

Boîtes décalées

`\raisebox{raise}[above][below]{texte}` : affiche `texte` en le décalant de `raise` par rapport à la ligne de base. Les arguments optionnels `above` et `below` permettent en plus de faire croire à `LaTeX` que le texte s'étend en haut et en bas.

Code



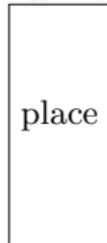
Fichier Intermediaire/code21.tex

```
On peut mettre du texte \raisebox  
{1ex}{plus haut} ou \raisebox{-1  
ex}{plus bas} et on peut faire  
croire à LaTeX qu'il occupe  
beaucoup de \fbox{\raisebox{0pt  
}[1cm][1cm]{place}}.
```

Sortie

On peut mettre du texte
plus haut ou plus bas et on
peut faire croire à `LaTeX` qu'il

occupe beaucoup de place .



Boîtes déformantes

Ne fonctionne pas avec les fichiers `dvi` (mais `ps` et `pdf` devraient s'afficher correctement).

- `\reflectbox{texte}` : affiche `texte` avec une symétrie d'axe vertical ;
- `\scalebox{hscale}[vscale]{texte}` : dilate le texte d'un facteur `hscale` (et optionnellement d'un facteur différent `vscale` verticalement) ;
- `\rotatebox{angle}{texte}` : applique une rotation d'`angle` degrés à `texte`.

Code



Fichier Intermediaire/code22.tex

```
\reflectbox{À l'envers}  
  
\scalebox{3}{Dilaté}  
  
\scalebox{0.5}[2]{Contracté}  
  
\rotatebox{45}{Pivoté}
```

Sortie

À l'envers
Dilaté
Contracté
Pivoté

Nécessite l'extension `color`.

Code



Fichier Intermediaire/code23.tex

```
\colorbox{red}{Seule la couleur de  
fond change}
```

Sortie

Seule la couleur de fond change

Longueurs associées aux boîtes

À chaque boîte en cours de traitement sont associées quatre longueurs.

- `\width` : largeur de la boîte courante ;
- `\height` : hauteur de la boîte courante ;
- `\depth` : profondeur de la boîte courante ;
- `\totalheight` : hauteur totale de la boîte courante (somme de sa hauteur et de sa profondeur).

Code



Fichier Intermediaire/code24.tex

```
\framebox[1.5\width]{Boîte trop  
large}  
  
\framebox[0.5\width]{Boîte pas  
assez large}
```

Sortie

Boîte trop large

Boîte pas assez large



Personnaliser la mise en page

Table des matières

Espaces verticales standard

Les commandes `\smallskip`, `\medskip` et `\bigskip` permettent d'insérer un espacement vertical de hauteur proportionnelle à la police utilisée (respectivement `\smallskipamount`, `\medskipamount` et `\bigskipamount`). On peut les utiliser avant un saut de paragraphe ou de ligne.

Code



Fichier Intermediaire/code25.tex

```
Entre chaque paragraphe  
je fais un petit saut\smallskip  
puis un moyen\medskip  
puis un grand\bigskip  
qui m'amène ici.
```

Sortie

Entre chaque paragraphe
je fais un petit saut
puis un moyen
puis un grand
qui m'amène ici.



Sortie

Comment obtenir des espaces d'un pouce en haut et en bas du texte ?

Code



Fichier Intermediaire/code27.tex

```
\vspace{1in}  
Comment obtenir des espaces d'un pouce en haut et en bas du texte?  
\vspace{1in}
```

Espaces explicites forcées

- `\vspace*{size}` : ajoute une espace verticale de hauteur `size` qui ne sera pas coupée si elle dépasse de la page ;
- `\hspace*{size}` : ajoute une espace horizontale de longueur `size` qui ne sera pas coupée si elle dépasse de la ligne ;

Code



Fichier Intermediaire/code29.tex

```
Ceci est une espace qui a été \  
\hspace{3cm} automatiquement  
supprimée.
```

```
Voilà une espace qui est toujours  
\hspace*{3cm} conservée.
```

Sortie

Ceci est une espace qui a été automatiquement supprimée.

Voilà une espace
qui est toujours
 conservée.

Les longueurs peuvent être données avec une marge (on parle de colle). Par exemple : $1\text{ cm} \pm 5\text{ mm}$. En fonction du mode (vertical ou horizontal), \LaTeX adaptera les valeurs pour avoir le meilleur rendu.

Code



Fichier Intermediaire/code30.tex

```
Voici une ligne\hspace{1cm plus 5  
mm minus 5mm} avec une espace  
élastique.
```

```
Une autre ligne\hspace{1cm plus 5  
mm minus 5mm} pour montrer  
comment ça peut changer.
```

Sortie

Voici une ligne avec une
espace élastique.

Une autre ligne pour mon-
trer comment ça peut changer.

Espaces infinis

Les commandes `\hfill` et `\vfill` (ou `\hspace{\fill}` et `\vspace{\fill}`) insèrent un espace élastique spécial qui remplit toute la hauteur ou largeur disponible.

Attention : \LaTeX supprime les espaces en début et fin de ligne ou de page. On peut insérer un caractère « fantôme » (`\null`) pour l'éviter, ou alors utiliser des espaces forcées.

Code



Fichier Intermediaire/code31.tex

```
Du texte centré
```

```
\vfill verticalement\vfill  
et\hfill horizontalement\hfill\  
null  
\hspace*{\fill}Autre façon\hspace  
*{\fill}
```

Sortie

Du texte centré

verticalement

et horizontalement
 Autre façon



Exercice

Sortie

Comment mettre un mot

au bout de la ligne ?



Solution

Code



Fichier Intermediaire/code33.tex

```
Comment mettre un mot \hfill au bout de la ligne?
```

Sortie

Jean DUPOND

18 décembre 2013

Exercice de L^AT_EX

Exemples et applications

Code



Fichier Intermediaire/code34.tex

```
Jean \textsc{Dupond}\hfill\today\[\fill]  
\null\hfill\Ovalbox{Exercice de \LaTeX}\hfill\null\[\bigskipamount]  
\null\hfill{\footnotesize Exemples et applications}\hfill\null\[\fill]  
\null\hfill\textsc{Paris Sud}
```



Sortie

N° d'ordre : 1234

Faculté des sciences d'Orsay

THÈSE

Présentée pour obtenir

LE GRADE DE DOCTEUR EN SCIENCES
DE L'UNIVERSITÉ PARIS SUD

Spécialité : L^AT_EX

par

Jean DUPOND

Comment faire une page de titre

Soutenue le 18 décembre 2013

Code



Fichier Intermediaire/code35.tex

```
\scriptsize N\textdegree\ d'ordre: 1234\hfill Faculté des sciences d
'Orsay

\vfill

\begin{center}
\normalsize\textbf{THÈSE}\medskip\\
\scriptsize Présentée pour obtenir\medskip\\
\small LE GRADE DE DOCTEUR EN SCIENCES\\
DE L'UNIVERSITÉ PARIS SUD\medskip\\
\scriptsize Spécialité: \LaTeX\smallskip\\
par\smallskip\\
Jean \textsc{Dupond}\[\fill]
\normalsize\fbx{\bfseries\scshape Comment faire une page de titre}
\end{center}

\vfill

\scriptsize Soutenue le \today
```



Personnaliser L^AT_EX

Table des matières



Personnaliser L^AT_EX

Table des matières

Commentaires de fin de ligne

L'une des particularités de L^AT_EX est d'ignorer les espaces en début de ligne situés après une ligne terminée par un commentaire. Cela permet d'écrire du code indenté sans introduire d'espaces supplémentaires non désirées.

Code



Fichier Intermediaire/code36.tex

```

Un mot qui change tout le long:
\color{red}anti%
\textbf{%
  consti%
  \color{green}%
  tu%
  \textit{%
    tion%
  }%
  nelle%
  \color{blue}%
  ment%
}%
.

```

Sortie

Un mot qui change
 tout le long :
anticonstitutionnellement.

Tout d'abord on passe comme options à l'extension `babel` tous les langages désirés séparés par des virgules, le dernier étant le langage principal. Ensuite, utiliser `\selectlanguage{lang}` à l'endroit désiré (et ne pas oublier de revenir de la même façon au langage principal ensuite).

Code



Fichier Intermediaire/code37.tex

```
%\usepackage[english,français]{
  babel} dans le préambule
\begin{abstract}C'est le résumé en
français.\end{abstract}
\selectlanguage{english}
\begin{abstract}This is the
English abstract.\end{abstract}
\selectlanguage{français}
```

Sortie

Résumé

C'est le résumé en français.

Abstract

This is the English abstract.

Disposition des flottants

Les environnements `figure` et `table` prennent un argument optionnel qui peut prendre une combinaison des valeurs suivantes :

- `h` pour inciter à disposer le flottant ici ;
- `t` pour inciter à disposer le flottant en haut de la page ;
- `b` pour le bas de la page ;
- `p` pour le mettre sur une page supplémentaire ;
- `!` pour formuler une demande de placement avec insistance.

Code



Fichier Intermediaire/code38.tex

```
\begin{figure}[h!]  
  \includegraphics [width=0.9\textwidth]{image.eps}  
  \caption{Ma figure}  
  \label{myfigure}  
\end{figure}
```



Disposition forcée

Les flottants peuvent, dans certains cas, être disposés par L^AT_EX très loin de l'emplacement où ils sont définis, parfois même avant.

L'extension `flafter` empêche les flottants d'être disposés avant leur définition.

La commande `\clearpage` termine la page en cours et affiche tous les flottants en suspens. Si le saut de page est indésirable on peut utiliser `\afterpage{\clearpage}` qui nécessite l'extension `afterpage`.

On peut aussi utiliser la commande `\FloatBarrier` de l'extension `placeins`. La commande `\suppressfloats` empêche tout flottant d'être disposé sur la page en cours. Elle peut prendre un argument optionnel (`t` ou `b`) pour interdire le haut ou le bas de la page uniquement.

Notes de bas de page dans les titres et flottants

Sans entrer dans les détails, `\footnote` est une commande **fragile**, c'est-à-dire que son argument n'est pas placé au même endroit que son utilisation. Il faut donc la **protéger** lorsqu'on l'utilise dans les titres. Dans le cas des flottants c'est beaucoup plus compliqué ; on peut s'en tirer en utilisant un environnement **minipage**.

Code



Fichier Intermediaire/code39.tex

```
\section{Une section\protect\footnote{Avec une note protégée}}

\begin{figure}
  \includegraphics[width=2cm]{lenna.png}
  \caption{%
    \begin{minipage}{\textwidth}%
      Cette figure a une note\footnote{De bas de paragraphe}.%
    \end{minipage}%
  }
\end{figure}
```



Personnaliser L^AT_EX

Table des matières

Définition d'un compteur

- `\newcounter{XXX}` : définit le compteur `XXX`. Attention, il s'agit d'un nom formé de lettres uniquement, sans antislash ;
- `\setcounter{XXX}{valeur}` : stocke le nombre entier `valeur` dans le compteur `XXX` ;
- `\stepcounter{XXX}` : incrémente le compteur `XXX` ;
- `\addtocounter{XXX}{valeur}` : ajoute le nombre `valeur` au compteur `XXX` ;
- `\theXXX` : cette commande est automatiquement définie par `\newcounter{XXX}`, et affiche les caractères numériques (chiffres arabes) formant la valeur du compteur `XXX`.

Code



Fichier Intermediaire/code40.tex

```

\newcounter{XXX}
Mon compteur vaut \theXXX, puis \
stepcounter{XXX}\theXXX, puis \
addtocounter{XXX}{\theXXX}\theXXX
.
    
```

Sortie

Mon compteur vaut 0, puis 1,
puis 2.

Quelques compteurs prédéfinis

- **page** : le numéro de la page en cours ;
- **part** et **chapter** : le numéro de la partie ou du chapitre en cours ;
- **section**, **subsection** et **subsubsection** : le numéro de la section, sous-section ou sous-sous-section en cours ;
- **equation**, **figure**, **table** : le numéro de la dernière équation, figure ou table ;
- **footnote** : le numéro de la dernière note de bas de page ;
- **enumi**, **enumii**, **enumiii**, **enumiv** : le numéro de l'élément en cours dans une énumération (avec plusieurs niveaux d'imbrication) ;

Code



Fichier Intermediaire/code41.tex

```
Nous sommes page \thepage, section  
\thesection.
```

Sortie

Nous sommes page 104, section
4.



Comment numéroter la première page à partir de zéro ?



Utiliser `\setcounter{page}{0}` au début du document.

Affichage de la valeur d'un compteur

- `\arabic{XXX}` : affiche le compteur en chiffres arabes (équivalente à `\theXXX`);
- `\roman{XXX}` : chiffres romains en minuscules;
- `\Roman{XXX}` : chiffres romains en majuscules;
- `\alph{XXX}` : numérotations par des lettres minuscules (entre 1 et 26);
- `\Alph{XXX}` : numérotations par des lettres majuscules (entre 1 et 26);
- `\fnsymbol{XXX}` : utilisation de neuf symboles spéciaux (*, †, ‡, §, ¶, ||, **, †† et ‡‡).

Code



Fichier Intermediaire/code42.tex

```

\newcounter{Siecle}
\setcounter{Siecle}{21}
Au \Roman{Siecle}\ieme\ siècle,
  les ordinateurs dominèrent la
  planète.
  
```

Sortie

Au XXI^e siècle, les ordinateurs dominèrent la planète.



Personnaliser L^AT_EX

Table des matières

Permet de spécifier les marges et les différentes tailles de mise en page en une seule ligne.

Code



Fichier Intermediaire/code43.tex

```
%Mode paysage
\usepackage[landscape]{geometry}

%Des marges horizontales et verticales de 2 cm tout autour du texte
\usepackage[margin=2cm]{geometry}

%Du texte de 7 pouces par 10 centré dans la page
\usepackage[text={7in,10in},centering]{geometry}

%Des marges différentes selon le côté de la page
\usepackage[hmargin={4cm,2cm},vmargin={2cm,3cm}]{geometry}
```



Personnaliser \LaTeX

Exercice



Code



Fichier Intermediaire/code44.tex

```
\usepackage[margin=0pt]{geometry}
```

L'extension `multicol`

Cette extension introduit l'environnement `multicols`, qui permet de changer localement le nombre de colonnes. On peut sauter à la colonne suivante avec `\columnbreak`.

La longueur `\columnseprule` contrôle l'épaisseur du trait de séparation (par défaut zéro) et `\columnsep` l'espace entre les colonnes.

Code



Fichier Intermediaire/code45.tex

```
\setlength\columnseprule{1pt}
\begin{multicols}{3}
Du texte sur plusieurs colonnes c'
est bien, mais attention à ce que
les colonnes soient assez larges
sinon ça peut devenir très laid.
\end{multicols}
```

Sortie

Du	mais at-	larges
texte	tention	sinon
sur plu-	à ce que	ça peut
sieurs	les co-	deve-
colonnes	lonnes	nir très
c'est	soient	laid.
bien,	assez	

Sortie

1 Quatre colonnes

Voici une première section sur quatre colonnes. Voici une première section	sur quatre colonnes. Voici une première section sur quatre colonnes. Voici une	première section sur quatre colonnes. Voici une première section sur quatre	colonnes.
--	--	---	-----------

2 Trois colonnes

Voilà une seconde section sur trois colonnes.	Voilà une seconde section sur trois colonnes. Voilà une seconde section sur trois colonnes. Voilà une seconde section sur trois colonnes.	Voilà une seconde section sur trois colonnes. Voilà une seconde section sur trois colonnes. Voilà une seconde section sur trois colonnes.	Voilà une seconde section sur trois colonnes. Voilà une seconde section sur trois colonnes. Voilà une seconde section sur trois colonnes.
---	---	---	---

Code



Fichier Intermediaire/code46.tex

```
\setlength\columnseprule{1pt}
\section{Quatre colonnes}
\begin{multicols}{4}
Voici une première section sur quatre colonnes. Voici une première
section sur quatre colonnes. Voici une première section sur quatre
colonnes. Voici une première section sur quatre colonnes. Voici une
première section sur quatre colonnes.
\end{multicols}
\section{Trois colonnes}
\begin{multicols}{3}
Voilà une seconde section sur trois colonnes.\columnbreak

Voilà une seconde section sur trois colonnes. Voilà une seconde
section sur trois colonnes. Voilà une seconde section sur trois
colonnes. Voilà une seconde section sur trois colonnes. Voilà une
seconde section sur trois colonnes. Voilà une seconde section sur
trois colonnes.
\end{multicols}
```



L'extension `verbatim`

Cette extension permet d'ajouter du texte façon machine à écrire, sans y changer quoi que ce soit. En particulier les sauts de ligne et espaces sont conservés tels quels. À noter que le `verbatim` ne peut pas être utilisé comme argument d'une commande.

- `\verb|texte|` : ajoute tout le texte compris entre les deux barres. Il est aussi possible d'utiliser n'importe quel autre symbole pour délimiter le texte (au cas où il contient `|`), et il n'est pas possible de sauter de ligne ;
- `\begin{verbatim}...\end{verbatim}` : même principe, mais on peut sauter des lignes dans le texte ;
- `\verbatiminput{fichier}` : ajoute le contenu d'un fichier.

Code



Fichier Intermediaire/code47.tex

```
La commande \verb|\LaTeX| affiche  
\LaTeX.  
\begin{verbatim}  
Si la ligne est trop longue, elle  
va dépasser de la page.  
Les sauts de ligne sont respectés,  
et les espaces      sont  
conservées  
\end{verbatim}
```

Sortie

La commande `\LaTeX` affiche
L^AT_EX.

Si la ligne est trop longue, elle
Les sauts de ligne sont respectés
et les espaces sont conser



L'extension `fancyhdr`

Elle introduit le style de page supplémentaire `fancy` (utilisable avec `\pagestyle`). On peut modifier l'en-tête avec :

- `\lhead[pages paires]{page impaires}` pour afficher du texte à gauche de l'en-tête ;
- `\chead[pages paires]{page impaires}` pour afficher du texte au centre de l'en-tête ;
- `\rhead[pages paires]{page impaires}` pour afficher du texte à droite de l'en-tête.

Les commandes `\lfoot`, `\cfoot` et `\rfoot` s'utilisent de même pour le pied de page.

On peut aussi utiliser `\fancyhead[position]{texte}` et `\fancyfoot[position]{texte}` où `position` est composé des lettres **L** (left), **C** (centré), **R** (right), **O** (odd) et **E** (even).

On peut encore utiliser `\fancyhf[position]{texte}` avec **H** pour indiquer l'en-tête et **F** pour le pied de page.

Les commandes `\headrulewidth` et `\footrulewidth` contrôlent l'épaisseur du trait vertical en-dessous de l'en-tête et au-dessus du pied de page (attention, ce ne sont pas des longueurs!).

Code



Fichier Intermediaire/code48.tex

```

%Supprimer le trait d'en-tête
\renewcommand\headrulewidth{0pt}

%Désactiver toutes les décorations
\fancyhf{}

%Style standard
\fancyfoot [C]{\thepage}

%Ou encore
\fancyhf [CF]{\thepage}

%Exemple classique
\lhead [\textbf{\thepage}]{\textsl{\rightmark}}
\rhead [\textsl{\leftmark}]{\textbf{\thepage}}

%Formulation équivalente
\fancyhead [LE,RO]{\textbf{\thepage}}
\fancyhead [LO]{\textsl{\rightmark}}
\fancyhead [RE]{\textsl{\leftmark}}
  
```

L'extension `lastpage`

Cette extension crée automatiquement une étiquette appelée `LastPage` placée sur la dernière page du document. Comme toutes les étiquettes, elle peut nécessiter une double compilation pour être mise à jour.

Code

Fichier Intermediaire/code49.tex

```
Ce document contient \pageref{
  LastPage}~pages au total.
```

Sortie

Ce document contient ?? pages
au total.



Personnaliser \LaTeX

Exercice



Code



Fichier Intermediaire/code50.tex

```
\pagestyle{fancy}  
\cfoot{Page \thepage/\pageref{LastPage}}
```



Personnaliser L^AT_EX

Table des matières

Nouvelles commandes

On utilise `\newcommand\xxx{...}`. La commande `\xxx` ne doit pas être déjà définie. On peut utiliser `\providecommand` avec la même syntaxe, qui définit la commande seulement si celle-ci ne l'est pas déjà.

Code



Fichier Intermediaire/code51.tex

```
\newcommand\acidlysergic{acide 6-  
Mé\ -thyl-9,10-di\ -de\ -hy\ -dro\ -er  
\ -go\ -li\ -ne-8-car\ -bo\ -xy\ -li\  
que}  
L'\acidlysergic\ est long à taper!  
  
\newcommand\bi{\begin{itemize}}  
\newcommand\ei{\end{itemize}}  
\bi  
\item les commandes \verb|\bi|  
\item et \verb|\ei|  
\item sont plus courtes à taper!  
\ei
```

Sortie

L'acide 6-Méthyl-9,10-didehydroergoline-8-carboxylique est long à taper !

- les commandes `\bi`
- et `\ei`
- sont plus courtes à taper !

Nouvelles commandes avec paramètres

La commande `\newcommand` peut prendre un paramètre optionnel compris entre 1 et 9 qui définit le nombre d'arguments de la nouvelle commande. Ceux-ci seront identifiés par `#1`, `#2`, ... `#9` dans la définition.

Code



Fichier Intermediaire/code52.tex

```
\newcommand\evidence [1]{%
  \fbox{\color{red}\textbf{#1}}}%
}
Ce \evidence{mot} est mis en \
evidence{évidence}.

\newcommand\norm [2]{$\sqrt{(#1)^2+(#2)^2}$}
Distance: \norm{a-b}{c-d}.
```

Sortie

Ce mot est mis en évidence.

Distance $\sqrt{(a-b)^2 + (c-d)^2}$:

Nouvelles commandes avec paramètre optionnel

`\newcommand` peut prendre un second paramètre optionnel qui rend le premier paramètre de la nouvelle commande optionnel et permet de lui associer une valeur par défaut.

Code



Fichier Intermediaire/code53.tex

```
\newlength\longueur
\newcommand\strikeout [2] [1pt]{%
  \settowidth\longueur{#2}%
  \makebox [0pt] [1]{\rule [0.4ex]{\
    longueur}{#1}}%
  #2%
}
Ce \strikeout{texte} est \
strikeout [3pt]{barré}.
```

Sortie

Ce ~~texte~~ est barré.

`\newenvironment{nom}{debut}{fin}` définit un nouvel environnement. Lorsque celui-ci sera utilisé, le contenu de `debut` et `fin` sera inséré de chaque côté du texte dans l'environnement.

Code



Fichier Intermediaire/code54.tex

```
\newenvironment{redquotation}{\
  begin{quotation}\color{red
} <<-}{->\end{quotation}}

\begin{redquotation}
C'est un peu flashy, non?
\end{redquotation}
```

Sortie

*« C'est un peu
flashy, non ? »*



Personnaliser L^AT_EX

Table des matières

Redéfinir les commandes

La commande `\renewcommand` s'utilise comme `\newcommand` pour redéfinir des commandes déjà existantes. Exemples d'utilisation :

- `\renewcommand\headrulewidth{0pt}` (avec l'extension `fancyhdr`);
- `\renewcommand\baselinestretch{2}` pour passer en interligne double;
- `\renewcommand\arraystretch{1.5}` change l'interligne des tableaux;
- `\renewcommand\figurename{Dessin}` pour changer le nom des figures;
- `\renewcommand\tablename{Listing}` pour changer le nom des tables;
- `\renewcommand\contentsname{Sommaire}` pour changer le nom de la table des matières.

Code



Fichier Intermediaire/code55.tex

```
%Pour faire apparaitre le chapitre et la section en cours dans les
en-têtes
\renewcommand\chaptermark [1]{\markboth{#1}{}}
\renewcommand\sectionmark [1]{\markright{#1}{}}
```



Personnaliser \LaTeX

Exercice



Code



Fichier Intermediaire/code56.tex

```
\renewcommand\thefootnote{\fnsymbol{footnote}}
```

Sortie

Comment changer la numérotation des éléments d'une énumération ?

- A. Les
 - a. commandes
 - b. à
- B. redéfinir
- C. sont
 - a. `\labelenumi`
 - b. `\labelenumii`
 - c. `\labelenumiii`
 - d. `\labelenumiv`

Code



Fichier Intermediaire/code57.tex

```

\renewcommand\labelenumi{\Alph{enumi}}
\renewcommand\labelenumii{\alph{enumii}}
Comment changer la numérotation des éléments d'une énumération?
\begin{enumerate}
  \item Les
  \begin{enumerate}
    \item commandes
    \item à
  \end{enumerate}
  \item redéfinir
  \item sont
  \begin{enumerate}
    \item\verb|\labelenumi|
    \item\verb|\labelenumii|
    \item\verb|\labelenumiii|
    \item\verb|\labelenumiv|
  \end{enumerate}
\end{enumerate}

```

Utiliser l'ancienne définition

On peut copier une commande existante sous un autre nom avec `\def\nouvelle=\ancienne`. On peut alors utiliser l'ancienne définition dans `\renewcommand`.

Code



Fichier Intermediaire/code59.tex

```
\let\oldsection=\section
\renewcommand\section[1]{\oldsection{\underline{#1}}}
\section{Section soulignée}
```

En théorie ça n'est pas la meilleure façon de redéfinir la commande `\verb|\section|`.

```
\section{Section soulignée}
```

Mais ça fonctionne.

Sortie

1 Section soulignée

En théorie ça n'est pas la meilleure façon de redéfinir la commande `\section`.

2 Section soulignée

Mais ça fonctionne.

Redéfinir les environnements

On utilise `\renewenvironment`. Pour redéfinir l'environnement `XXX` à partir de son ancienne définition, il faut préalablement dupliquer les commandes `\XXX` et `\endXXX` avec `\let`.

Code



Fichier Intermediaire/code60.tex

```
\let\oldquotation=\quotation
\let\oldendquotation=\endquotation
\renewenvironment{quotation}{%
  \oldquotation%
  \color{blue}%
}{%
  \oldendquotation%
}
\begin{quotation}
Une citation bleue.
\end{quotation}
```

Sortie

*Une citation
bleue.*



Gérer de gros documents

Table des matières



Gérer de gros documents

Table des matières

Utiliser les commentaires !

Code



Fichier Intermediaire/code61.tex

%Ça peut être une bonne idée de souligner les sections, sous-sections etc... de façon à les mettre en valeur et les repérer du premier coup d'oeil.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
\section{Gérer de gros documents}  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
  
%-----%  
\subsection{S'organiser}
```



Répartir le contenu dans des sous-fichiers

Il y a deux commandes qui permettent de faire ça : `\input{fichier}` ou `\include{fichier}` où `fichier` est le nom d'un fichier `.tex` sans extension. À la différence de la première, `\include` génère un fichier `.log` supplémentaire et commence toujours une nouvelle page. Par contre, `include` ne peut pas être utilisée dans le préambule, ni "s'emboîter" (un fichier appelé avec `\include` ne peut pas contenir cette commande).

Dans le préambule, la commande `\includeonly{fichier1,fichier2,...}` désactive l'inclusion des fichiers qui ne sont pas dans la liste, en ignorant les commandes `\include{...}` supplémentaires. Cela permet de compiler une partie sans changer les changements de pages : très utile quand les extensions commencent à prendre beaucoup de temps !



Une manière d'organiser sa thèse

Code



Fichier Intermediaire/code62.tex

```
\documentclass{report}

%Packages, format, etc...
\input{format}

\begin{document}

\maketitle
%Résumé(s)
\input{abstract}

\input{partie1}
\input{partie2}
\input{partie3}

\end{document}
```



Mettre les figures dans un sous-répertoire

L'extension `graphicx` permet d'insérer des images avec la commande `\includegraphics`. On peut spécifier un répertoire où les figures seront recherchées avec `\graphicspath{dossier}`. Par exemple le sous-dossier `figures/` à l'intérieur du répertoire du document est un choix possible...



Gérer de gros documents

Table des matières

Listes personnalisées

L'environnement `list` s'utilise avec deux arguments :

Code



Fichier Intermediaire/code63.tex

```
\begin{list}{label}{spacing}  
  \item Premier élément  
  \item Deuxième élément  
  ....  
\end{list}
```

`label` permet de définir l'étiquette utilisée et `spacing` permet de changer les longueurs définissant les espaces de la liste.

Listes personnalisées

Les longueurs modifiables sont les suivantes :

- `\topsep` : espace verticale entre la liste et ce qui la précède ;
- `\partopsep` : espace verticale supplémentaire lorsque la liste est précédée d'une ligne vide (on utilise généralement une longueur élastique) ;
- `\itemsep` : séparation verticale entre les éléments ;
- `\parsep` : séparation verticale entre les paragraphes d'un même élément ;
- `\leftmargin` et `\rightmargin` : espaces à gauche et à droite entre la liste et son contenant ;
- `\itemindent` : indentation du premier paragraphe d'un élément (éventuellement négatif) ;
- `\listparindent` : indentation des paragraphes suivants ;
- `\labelsep` : séparation entre l'étiquette et le début d'un élément ;
- `\labelwidth` : largeur minimale de l'étiquette ;
- `\usecounter{XXX}` : indique d'incrémenter le compteur `XXX` à chaque `\item`.

Listes personnalisées

Code



Fichier Intermediaire/code64.tex

```
\newcounter{maliste}
\begin{list}{\ 'Elément ~\Roman{
maliste} --- }{\usecounter{
maliste}\setlength\leftmargin{1cm
}\setlength\itemsep{1cm}}
\item C'est le premier élément ;
\item et le second.
\end{list}
```

Sortie

Élément I — C'est le premier élément ;

Élément II — et le second.



Exercice

Comment faire un environnement de liste avec des questions numérotées dans la marge, et un espace pour répondre en-dessous ?

Du texte avant la question pour constater que l'indentation des paragraphes est conservée.

Question 1.

Voici la première question, voici la première question, voici la première question.

Noter le changement de paragraphe avant l'item suivant, et l'indentation correspondante.

Question 2.

Voici la deuxième question, voici la deuxième question, voici la deuxième question.

Du texte après la dernière question (avec une espace verticale avant).

Code



Fichier Intermediaire/code65.tex

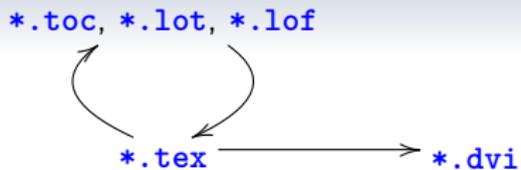
```

\newcounter{question}
\newenvironment{question}{%
  \begin{list}{\hfill Question-\arabic{question}.}{
    \usecounter{question}
    \setlength\labelwidth{6cm}
    \setlength\labelsep{.3\parindent}
    \setlength\itemsep{5cm plus 1cm minus 1cm}
    \setlength\itemindent{\parindent}
    \setlength\listparindent{\parindent}
    \setlength\leftmargin{0cm}
  }%
}{%
  \end{list}\vspace*{1cm}%
}%
\begin{question}
  \item Voici la première question, voici la première question,
  voici la première question.
  \item Voici la deuxième question, voici la deuxième question,
  voici la deuxième question.
\end{question}

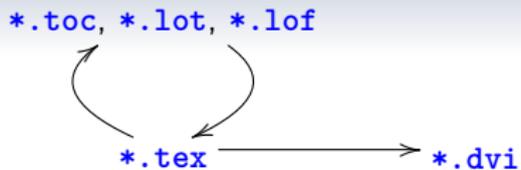
```



Compilation des tables

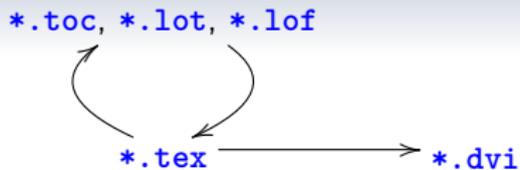


Compilation des tables



Le fichier `.toc` contient la table des matières.

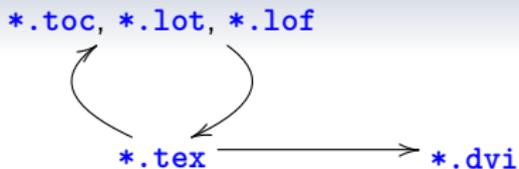
Compilation des tables



Le fichier `.toc` contient la table des matières.

Le fichier `.lot` contient la liste des tables.

Compilation des tables



Le fichier `.toc` contient la table des matières.

Le fichier `.lot` contient la liste des tables.

Le fichier `.lof` contient la liste des figures.



Génération des tables

- `\tableofcontents` : affiche la table des matières ;
- `\listoftables` : affiche la liste des tables ;
- `\listoffigures` : affiche la liste des figures ;
- `\addcontentsline{type}{level}{texte}` : ajoute une entrée dans une liste. L'argument `type` (`toc`, `lot` ou `lof`) spécifie quelle liste, `level` à quel niveau de la liste (par exemple `section`, `subsection` pour `toc`, `table` pour `lot` ou encore `figure` pour `lof`). Les version étoilées des commandes et environnements correspondants (`\section*`, `\begin{figure*}`) n'ajoutent rien dans la liste, et suppriment la numérotation.



Agir sur la table des matières

Les commandes de sectionnement (`\section`, etc.) prennent un argument optionnel (par défaut le même que l'argument obligatoire) qui spécifie le titre à apparaître dans la table des matières. La commande `\caption` fait de même pour les flottants.

Le compteur `tocdepth` permet de régler la profondeur maximale de la table des matières.

Niveau	<code>tocdepth</code>
part	-1
chapter	0
section	1
subsection	2
subsubsection	3
paragraph	4
subparagraph	5

Remarque : seules les classes `report` et `book` définissent `\part` et `\chapter`, mais les indices sont les mêmes pour `article`.



Agir sur la table des matières

Les commandes de sectionnement ne sont plus numérotées à partir d'une certaine profondeur (`\paragraph` par défaut).

Le compteur `secnumdepth` indique la dernière profondeur qui est numérotée.

Niveau	<code>secnumdepth</code>
part	-1
chapter	0
section	1
subsection	2
subsubsection	3
paragraph	4
subparagraph	5

Remarque : seules les classes `report` et `book` définissent `\part` et `\chapter`, mais les indices sont les mêmes pour `article`.

Changer le format des sections : l'extension `titlesec`

On peut utiliser directement les options de l'extension :

- `rm`, `sf`, `tt`, `md`, `bf`, `up`, `it`, `sl`, `sc`
- `big`, `medium`, `small`, `tiny`
- `raggedleft`, `center`, `raggedright`
- `compact`

On peut changer le format de numérotation avec `\titlelabel{format}`, par exemple :

Code



Fichier Intermediaire/code66.tex

```
\titlelabel{\thetitle.\hspace{2em}}
```

On peut aussi changer le style d'un niveau en particulier avec `\titleformat*{command}{format}`, par exemple :

Code



Fichier Intermediaire/code67.tex

```
\titleformat*{\section}{\itshape}
```



Personnaliser la table des matières : l'extension `titletoc`

On utilise

```
\titlecontents{section}[left]{above}
    {before with label}{before without label}
    {filler and page}[after]
```

Par exemple :

Code



Fichier Intermediaire/code68.tex

```
\titlecontents{section}
    [1.5em]
    {}
    {\contentslabel{2.3em}}
    {\hspace*{-2.3em}}
    {\titlerule*[1pc]{.}\contentspage}
```

Tables des matières partielles

L'extension `minitoc` permet de générer des sommaires partiels, par exemple pour chaque chapitre.

Code



Fichier Intermediaire/code69.tex

```
\documentclass{report}
\usepackage{minitoc}

\begin{document}
%Activation
\dominitoc
\tableofcontents

\chapter{Introduction}
\minitoc
\section{Quelques rappels}
\section{Quelques définitions}

\end{document}
```



Gérer de gros documents

Table des matières



Les flèches pleines indiquent la compilation \LaTeX , les flèches en pointillés avec `makeindex`.



Entrées

Avant toute chose on utilise `\usepackage{makeidx}`, puis `\makeindex` dans le préambule. L'index sera affiché grâce par la commande `\printindex` dans le document.

Là où l'on souhaite indiquer que le mot **XXX** apparait, on utilise `\index{XXX}` juste à côté. Lorsqu'un mot à indexer contient des caractères accentués on utilise par exemple `\index{ete@été}` (sinon il ne sera pas classé correctement dans la liste). Même remarque si on souhaite utiliser des commandes.

Code



Fichier Intermediaire/code70.tex

```
Bias\index{Bias@\textsc{Bias}},  
dont la sagesse est plus durable  
sur la terre que ne fut Priène\  
index{Priene@Priène} sa patrie,  
[...] disait que [...]. Ce  
principe\index{principe}, à  
première vue, pourrait sembler  
peut-être trop prudent.
```

Sortie

Bias, dont la sagesse est plus durable sur la terre que ne fut Priène sa patrie, [...] disait que [...]. Ce principe, à première vue, pourrait sembler peut-être trop prudent.

On utilise des points d'exclamation : `\index{Principale!sous-entree}`. Il est bien sûr possible d'avoir des réécritures, dans quel cas on posera l'arobase après l'entrée.

Code



Fichier Intermediaire/code71.tex

```
Bias\index{Bias@\textsc{Bias}}
[...] disait que, dans la
pratique de l'amitié\index{
Relations humaines!amitie@amitie
}, il faut se comporter de
manière à ne pas perdre de vue qu
'elle peut se changer un jour en
une haine\index{Relations
humaines!haine} implacable.
```

Sortie

Bias [...] disait que, dans la pratique de l'amitié, il faut se comporter de manière à ne pas perdre de vue qu'elle peut se changer un jour en une haine implacable.

Références croisées

Les références croisées sont introduites par une barre verticale. Donc on peut faire `\index{Principale!sous-entree@sous-entrée|see{Autre}}`
Cela a l'effet particulier d'empêcher les références possédant une telle barre (par exemple, si vous compilez une liste de notations usuelles en maths... voilà encore une bonne raison de préférer `$$\vert$` ou `$$\Vert$` à les entrer directement).

Code



Fichier Intermediaire/code72.tex

```
Ce principe\index{principe}, à
première vue, pourrait sembler
peut-être trop prudent et
contraire à la franchise qui est
le principal charme des relations
amicales\index{amicales|see{
Relations humaines}}; mais, quand
on y aura réfléchi\index{
principe!reflechir@réflechir|see{
Bias}} plus profondément, on le
trouvera fort utile.
```

Sortie

Ce principe, à première vue, pourrait sembler peut-être trop prudent et contraire à la franchise qui est le principal charme des relations amicales; mais, quand on y aura réfléchi plus profondément, on le trouvera fort utile.



Gérer de gros documents

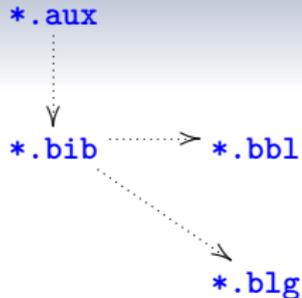
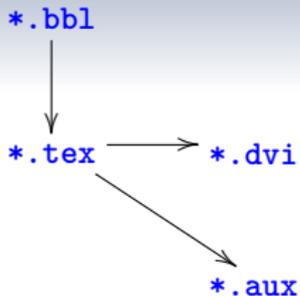
Résultat



Gérer de gros documents

Table des matières

Compilation



Les flèches pleines indiquent la compilation \LaTeX , les flèches en pointillés avec \BibTeX .



Utilisation

La commande `\bibliographystyle{style}` permet de définir le style des références du document. Puis `\bibliography{fichier}` – où `fichier` est le nom du fichier `.bib` sans extension – affiche la liste des références. BibTeX n'affichera que celles qui sont effectivement cités (on peut donc utiliser le même fichier `.bib` pour tous ses documents).

On peut faire une citation invisible avec `\nocite{ref}` ou encore `\nocite{*}` pour afficher la totalité du fichier `.bib`.

Quelques styles courants :

- `plain`, classement par ordre alphabétique, référence par des numéros ;
- `unsrt`, classement par ordre d'apparition, référence par des numéros ;
- `abbrv`, semblable à `plain` mais plus succinct ;
- `alpha`, semblable à `plain` mais référence par les initiales des auteurs et la date de publication ;

Exemple de fichier `.bib`

Code



Fichier Intermediaire/code73.tex

```
@book{lampport,
  title={\LaTeX: a document preparation system},
  author={Lampport, L.},
  year={1989},
  publisher={Addison-Wesley Longman Publishing Co., Inc. Boston, MA,
    USA}
}

@book{knuth,
  title={The \TeX book},
  author={Knuth, D.E. and Bibby, D.},
  year={1986},
  publisher={Addison-Wesley Reading, Mass}
}
```



Gérer de gros documents

Exercice

Sortie



Solution

Code



Fichier Intermediaire/code74.tex

```
\bibliographystyle{alpha}  
\nocite{lamport,knuth}  
\bibliography{example}
```



Table des matières



Table des matières

Lignes horizontales tronquées

On utilise `\cline{x-y}` pour tracer un trait horizontal entre les colonnes de numéro `x` et `y`.

Code



Fichier Intermediaire/code76.tex

```
\begin{tabular}{|c|c|c|c|}  
\hline  
A&B&C\\  
\cline{1-2}  
D&E&F\\  
\cline{2-3}  
G&H&I\\  
\cline{1-1}\cline{3-3}  
J&K&L\\  
\hline  
\end{tabular}
```

Sortie

A	B	C
D	E	F
G	H	I
J	K	L

Cellules sur plusieurs colonnes

On utilise `\multicolumn{cols}{pos}{text}` pour disposer `text` sur `cols` colonnes avec l'alignement `pos` (`r`, `c` ou `l`) avec éventuellement des traits verticaux. On peut en particulier l'utiliser pour supprimer un trait vertical sur une ligne.

Code



Fichier Intermediaire/code77.tex

```
\begin{tabular}{|c|c|c|c|}  
\hline  
A&\multicolumn{2}{c|}{B}\\  
\hline  
\multicolumn{2}{|c|}{C}&D\\  
\hline  
\multicolumn{1}{c}{E}&\multicolumn  
{1}{c}{F}&\multicolumn{1}{c}{G}\\  
\hline  
\end{tabular}
```

Sortie

A	B	
C		D
E	F	G

Cellules sur plusieurs lignes

On peut utiliser la commande `\multirow{rows}{*}{text}` de l'extension `multirow` pour disposer `text` sur `rows` lignes. Noter qu'il faudra s'assurer manuellement dans ce cas que les cellules recouvertes sont vides (sinon il y a des risques de chevauchement).

Code



Fichier Intermediaire/code78.tex

```
\begin{tabular}{|c|c|c|c|}  
\hline  
\multirow{2}{*}{A}&B\\  
\cline{2-2}  
&C\\  
\hline  
D&\multirow{2}{*}{E}\\  
\cline{1-1}  
F&\\  
\hline  
\end{tabular}
```

Sortie

A	B
	C
D	E
F	



Exercice

Sortie

A	B	C
D	E	
F		



Solution

Code



Fichier Intermediaire/code79.tex

```
\begin{tabular}{|c|c|c|}  
\hline  
A&B&C\\  
\hline  
D&\multicolumn{2}{c|}{\multirow{2}{*}{E}}\\  
\cline{1-1}  
F&\multicolumn{1}{c}{}&\\  
\hline  
\end{tabular}
```

L'extension `array`

En plus de définir les types de colonne `m` et `b` (qui prennent en argument une longueur fixée) cette extension permet de donner une commande avant et après une colonne avec `>{}` et `<{}`.

Code



Fichier Intermediaire/code80.tex

```
%Un tableau avec une colonne en
  gras et une autre en mode
  mathématique
\begin{tabular}{>{\bfseries}r@{ :
  }>{$}l<{$}}
Coordonnée 1&x\\
Coordonnée 2&y\\
Norme&\sqrt{x^2+y^2}
\end{tabular}
```

Sortie

Coordonnée 1 : x

Coordonnée 2 : y

Norme : $\sqrt{x^2 + y^2}$

L'extension `tabularx`

Elle permet entre autres de créer des tableaux de largeur fixée, avec un type de colonne (**X**) qui adapte automatiquement sa largeur.

Code



Fichier Intermediaire/code81.tex

```

\begin{tabularx}{4cm}{|X|c|X|}
\hline
A&B&C\\
\hline
D&E&F\\
\hline
\end{tabularx}
    
```

Sortie

A	B	C
D	E	F

L'extension `xy`

Permet de dessiner des diagrammes commutatifs, des automates, etc... On l'inclut avec `\usepackage[all]{xy}`.

Code



Fichier Intermediaire/code82.tex

```

\[\xymatrix{
  A\ar[d]&B\ar[d]&C\\
  D\ar@{->}[r]\ar@{<-}[d]&E\ar@
  {<->}[ru]\ar@{.}>[d]\ar@{<=>}[
  rd]&F\ar@{_{(-)}->}[lu]\\
  G\ar@/^/@{->}[r]&H\ar@/_/@{->}[r
  ]&I
}
\]
    
```

Sortie

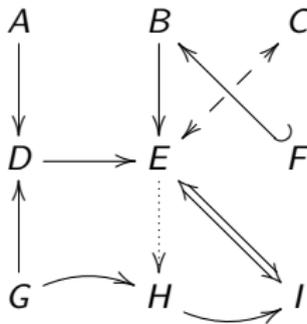




Table des matières

Superposition de symboles

- `\stackrel{top}{bottom}` permet de placer une boîte (en police plus petite) au-dessus d'une autre ;
- `\mathop{...}` permet de grouper des symboles comme un symbole unique avec des indices et exposants verticaux ;
- `\overset{up}{bottom}` et `\underset{below}{bottom}` permettent de placer un symbole au-dessus d'un autre.

Code

Fichier Intermediaire/code83.tex

```

 $\text{H}_2\text{CO}_3 \stackrel{\text{chaleur}}{\longrightarrow} \text{H}_2\text{O} + \text{CO}_2$ 
% Ça demande l'extension chemsym

```

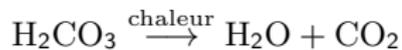
```


$$\mathop{\sum \sum}_{i,j=1}^N a_i a_j$$


```

L'intérieur de A est noté $\overset{\circ}{A}$.

Sortie



$$\sum_{i,j=1}^N a_i a_j$$

L'intérieur de A est noté $\overset{\circ}{A}$.

Les espaces dans les formules

Code



Fichier Intermediaire/code84.tex

```

Espace négatif : $a!b$\\
Pas d'espace : $ab$\\
Espace fine : $a\,b$\\
Espace moyenne : $a\;b$\\
Grande espace : $a\ b$ ou $a~b$\\
Cadratin : $a\quad b$\\
Double cadratin : $a\qquad b$
    
```

Sortie

Espace négatif : $a!b$
 Pas d'espace : ab
 Espace fine : $a\,b$
 Espace moyenne : $a\;b$
 Grande espace : $a\ b$ ou $a~b$
 Cadratin : $a\quad b$
 Double cadratin : $a\qquad b$

Changer la taille de la police en mode mathématique

- `\displaystyle` active le mode `\display` (comme avec `\[...\]`);
- `\textstyle` active le mode en ligne (comme avec `$$...$$`);
- `\scriptstyle` active le mode indice;
- `\scriptscriptstyle` active le mode double indice.

Code



Fichier Intermediaire/code85.tex

Grâce à ça les fractions dans les lignes peuvent être agrandies: `$$\displaystyle\frac{a}{b}$$` au lieu de `$$\frac{a}{b}$$`.

Les opérateurs aussi sont affectés : `$$\displaystyle\sum_{i=1}^n$$` au lieu de `$$\sum_{i=1}^n$$`.

Les formules centrées peuvent aussi être réduites.
`\[\frac{a+b}{c+d}\]` `\qqquad` `\text{devient}` `\qqquad` `\textstyle\frac{a+b}{c+d}` `\]`

Sortie

Grâce à ça les fractions dans les lignes peuvent être agrandies : $\frac{a}{b}$ au lieu de $\frac{a}{b}$.

Les opérateurs aussi sont affectés : $\sum_{i=1}^n$ au lieu de $\sum_{i=1}^n$.

Les formules centrées peuvent aussi être réduites.

$$\frac{a+b}{c+d} \quad \text{devient} \quad \frac{a+b}{c+d}$$



Changer la taille des indices mathématiques

Code



Fichier Intermediaire/code86.tex

```
\DeclareMathSizes{10}{10}{6}{5}
```

Il faut remplacer **10** par la taille de la police de base du document.



Table des matières



C'est un ensemble d'extensions contenant entre autres :

- **amsmath** pour inclure directement les trois suivants, les environnements d'équations et en plus quelques symboles (comme `\AmS`) ;
- **amstext** pour écrire du texte en mode mathématique ;
- **amsbsy** pour les symboles gras ;
- **amsopn** pour la déclaration d'opérateurs ;
- **amsthm** pour les environnements **proof** et **theorem** ;
- **amssymb** pour des symboles supplémentaires.

Sous-équations

Avec les environnements `subequations` et `gather` on peut numérotter les sous-équations.

Code



Fichier Intermediaire/code87.tex

```
\begin{subequations}
  \begin{gather}
    \alpha+\beta=2\label{eq:1}\\
    \alpha\cdot\beta=1\label{eq:2}
  \end{gather}
  \alpha=\beta=1\label{eq:3}
\end{subequations}
```

D'après `\eqref{eq:1}` et `\eqref{eq:2}` on obtient `\eqref{eq:3}`.

Sortie

$$\alpha + \beta = 2 \quad (1a)$$

$$\alpha \cdot \beta = 1 \quad (1b)$$

$$\alpha = \beta = 1 \quad (1c)$$

D'après (1a) et (1b) on obtient (1c).

Groupement d'équations

- l'environnement `align` permet d'aligner les équations ;
- l'environnement `multline` gère les lignes trop longues ;
- l'environnement `split` aligne les suites d'équations.

Code



Fichier Intermediaire/code88.tex

```

\begin{align}
I_n&=\int_0^{\pi/2}\sin^2x\cos^{n-2}x\,dx \\
&\text{nonnumber} \\
J_n&=\int_0^{\pi/2}(1-\cos^2x)\cos^{n-2}x\,dx \\
\end{align}

\begin{multline}
\sum_{k=1}^{\infty}k^2=1+4+9+16+25+36+ \\
49+64+\ldots \\
\end{multline}
    
```

Sortie

$$I_n = \int_0^{\pi/2} \sin^2 x \cos^{n-2} x \, dx$$

$$J_n = \int_0^{\pi/2} (1 - \cos^2 x) \cos^{n-2} x \, dx \quad (2)$$

$$\sum_{k=1}^{\infty} k^2 = 1+4+9+16+25+36+$$

$$49 + 64 + \dots \quad (3)$$

Théorèmes

- `\newtheorem{name}{caption}` : définit l'environnement `name` ;
- `\newtheorem{name}{caption}[within]` : idem avec la numérotation héritée du compteur `within` ;
- `\newtheorem{name}[like]{caption}` : idem mais numérotation liée à celle de `like`.

Code



Fichier Intermediaire/code89.tex

```
\newtheorem{Def}{Définition}
\newtheorem{Thm}{Théorème}[subsection]
\newtheorem{Lem}[Thm]{Lemme}
\begin{Def}
Un triangle rectangle a un angle droit.
\end{Def}
\begin{Thm}[Pythagore]
Dans un triangle  $ABC$  rectangle en  $A$  on a  $AB^2+AC^2=BC^2$ .
\end{Thm}
\begin{Lem}
Si dans un triangle  $ABC$  on a  $AB^2+AC^2 \neq BC^2$  alors  $ABC$  n'
est pas rectangle en  $A$ .
\end{Lem}
```

On utilise l'environnement `proof`. Le symbole de fin de preuve (`\qed`) est inséré automatiquement à la fin.

Code

Fichier Intermediaire/code90.tex

```
\begin{proof}  
C'est évident!  
\end{proof}
```

Sortie**Démonstration.**

C'est évident!



On peut utiliser les environnements `matrix`, `bmatrix`, `pmatrix` ou `vmatrix` en fonction des délimiteurs désirés.

Code

Fichier Intermediaire/code91.tex

```
\[\begin{bmatrix}
0&\hdotsfor{2}&0\\
\vdots&&\vdots\\
0&\hdotsfor{2}&0\\
0&\dots&0&1
\end{bmatrix}\]
```

Sortie

$$\begin{bmatrix} 0 & \dots & 0 \\ \vdots & & \vdots \\ 0 & \dots & 0 \\ 0 & \dots & 0 & 1 \end{bmatrix}$$



Un peu de graphisme

Table des matières



Un peu de graphisme

Table des matières



Le paquet PGF

PGF est un paquet qui permet de créer des graphiques directement dans le code \TeX . Pour cela, il définit plusieurs commandes qui dessinent directement sur le document, au moyen de commandes **PostScript** ou **pdf**, selon le cas. Ensuite, c'est à l'utilisateur de composer ces commandes (à différents niveaux) et obtenir le résultat final, dans la même logique des **pstricks**.

Code



Fichier Intermediaire/code92.tex

```
On peut obtenir une ligne avec \
  tikz \draw (0pt,0pt) -- (20pt,6pt
);
et un cercle colori\`e avec \tikz
\fill[orange] (1ex,1ex) circle (1
ex);.
```

Sortie

On peut obtenir une ligne avec
— et un cercle colorié avec





Le paquet PGF

Comme le format `dvi` ne supporte pas les dessins de type `PostScript` ou `pdf`, ces commandes vont en fait utiliser des recours dits spéciaux pour placer du code directement dans le `ps` ou `pdf` final. Ce qui implique que la prévisualisation avec `xdvi` ne marche pas.



Les 3 parties de PGF

L'organisation de la programmation se fait en trois niveaux :

- La couche **système**, qui contient les commandes les plus basiques, et qui dépend du type de document à produire, et comment (**dvips**, **dvipdfm**, **pdftex**);
- La couche **basique**, qui fournit des commandes pour faire des dessins directement sans avoir à utiliser les commandes de la couche système;
- La couche **d'interface**, qui est celle de plus haut niveau et qui permet de faire des utilisations avancées utilisant (relativement) peu de commandes. Elle est appelée **TikZ**.

La plupart des utilisations normales de **PGF** se font majoritairement à l'aide de **TikZ**, plus parfois quelques commandes de la couche de base.

En \LaTeX , la façon d'inclure des parties graphiques se fait, bien sûr, à l'aide d'un environnement :

Code

Fichier Intermediaire/code93.tex

```
Je fais une croix
\begin{tikzpicture}
\draw (-0.5,0) -- (0.5,0);
\draw (0,-0.5) -- (0,0.5);
\end{tikzpicture}
compos\`ee de deux segments.
```

Sortie

Je fais une croix com-
posée de deux segments.

Cet environnement crée uniquement la plus petite boîte contenant le graphique, ce qui permet de l'inclure directement dans du texte, comme on a fait.

Commandes fondamentales

On a vu la commande **draw**, qui fait une ligne d'un point à un autre. Les formats **-**, **|-** et **-|** relient selon trois styles différents : la droite qui contient les deux points, ou sinon selon les verticales et horizontales, dans l'ordre indiqué. Si on rajoute **controls**, on peut contrôler les tangentes.

Code

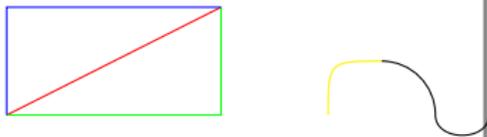


Fichier Intermediaire/code94.tex

```

\begin{tikzpicture}
\draw[red] (-1.5,0) -- (0.5,1);
\draw[green] (-1.5,0) -| (0.5,1);
\draw[blue] (-1.5,0) |- (0.5,1);
\draw[yellow] (1.5,0) .. controls
(1.5,0.5) .. (2,0.5);
\draw (2, 0.5) .. controls
(2.277,0.5) and (2.5,0.277)..
(2.5,0) .. controls (2.5,-.25)
and (3,-.25) .. (3,0);
\end{tikzpicture}
    
```

Sortie



Les points de contrôle règlent aussi le module du vecteur tangent.

Formes fondamentales

On peut dessiner des cercles, ellipses, rectangles et arcs de cercle :

Code



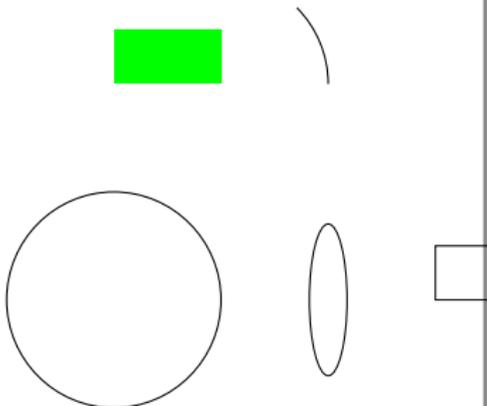
Fichier Intermediaire/code95.tex

```

\begin{tikzpicture}
\draw (0,0) circle (1cm);
\draw (2,0) ellipse (5pt and 20pt)
;
\draw (3,0) rectangle (3.5,0.5);
\fill[green] (0,2) rectangle
+(1,0.5);
\draw (2,2) arc (0:45:1);
\end{tikzpicture}

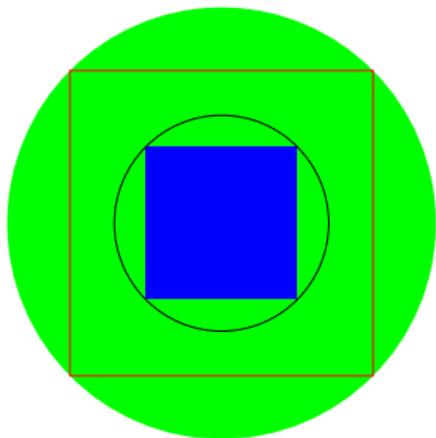
```

Sortie



qui peuvent être simplement « dessinés » ou remplis avec **fill**.

Sortie



Code



Fichier Intermediaire/code96.tex

```
\begin{tikzpicture}
  \fill[green] (0,0) circle (2);
  \fill[blue] (-0.707, -0.707) rectangle (0.707, 0.707);
  \draw      (0,0) circle (1);
  \draw[red] (-1.4142, -1.4142) rectangle (1.4142, 1.4142);
\end{tikzpicture}
```

On peut relier des dessins et des points :

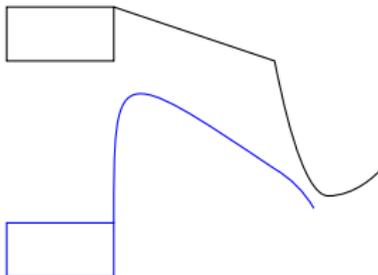
Code



Fichier Intermediaire/code97.tex

```
\begin{tikzpicture}
\draw (0,2) rectangle +(1,0.5) --
(2.5,2) parabola bend (3,0.75)
(3.5,1);
\draw[blue] (0,0) rectangle
+(1,0.5) .. controls (1,2) ..
(2.5,1) -- (2.5,1) arc (60:30:1);
\end{tikzpicture}
```

Sortie

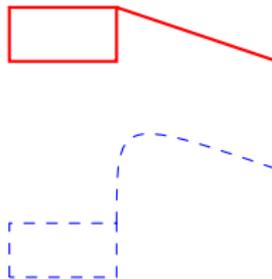


Tout sous **TikZ** peut avoir un style particulier ; c'est-à-dire, la couleur, le trait... Cela donne une grande flexibilité d'utilisation :

Code

Fichier Intermediaire/code98.tex

```
\begin{tikzpicture}
\draw[red,thick] (0,2) rectangle
+(1,0.5) -- (2.5,2) ;
\draw[blue,dashed] (0,0) rectangle
+(1,0.5) .. controls (1,1.5) ..
(2.5,1) ;
\end{tikzpicture}
```

Sortie

Déplacements et coordonnées

On peut effectuer des déplacements relatifs à l'aide de `+` ou `++`. La deuxième forme a pour effet de changer le **point courant** de **PGF**. Les unités sont des centimètres par défaut, mais on peut les remplacer par toute autre mesure comprise par **T_EX**.

On a beaucoup utilisé des coordonnées cartésiennes, mais on peut donner aussi des coordonnées polaires sous la forme **(angle:rayon)**.

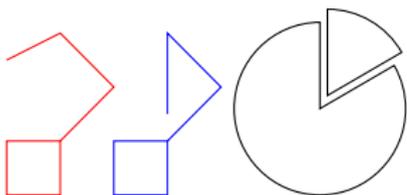
Code



Fichier Intermediaire/code99.tex

```
\begin{tikzpicture}[scale=0.5]
\draw[red] (0,0) rectangle +(1,1)
  -- (2,2) -- ++ (-1,1) --
  ++(-1,-0.5);
\draw[blue] (2,0) rectangle +(1,1)
  -- (4,2) -- +(-1,1) --
  +(-1,-0.5) ;
\end{tikzpicture}
\begin{tikzpicture}[scale=0.8]
\draw (0,0) -- (90:1) arc
  (90:360:1) arc (0:30:1) -- cycle;
\draw (60:5pt) -- +(30:1) arc
  (30:90:1) -- cycle;
\end{tikzpicture}
```

Sortie



Déplacements et coordonnées

Finalement, on peut donner des coordonnées comme intersections, de la forme $(p \mid - q)$ pour être l'intersection d'une droite verticale passant par le point p et une droite horizontale passant par le point q .

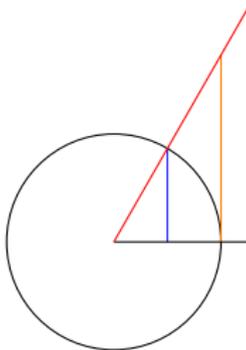
Code



Fichier Intermediaire/code100.tex

```
\tikzpicture
\draw (0,0) circle (1);
\draw[orange] (1,0) -- (
  intersection of 1,0--1,1 and
  0,0--60:1);
\draw[blue] (60:1) -- (60:1 \mid
  0,0);
\draw (0,0) -- (1.25,0);
\draw[red] (0,0) -- (intersection
  of 0,0--60:1 and 1.25,0--1.25,2);
\endtikzpicture
```

Sortie

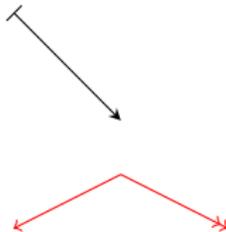


On peut rajouter des options à `draw` pour avoir des pointes dans les chemins :

Code

Fichier Intermediaire/code101.tex

```
\begin{tikzpicture}  
\draw[<->,color=red] (0,0) --  
  (1,0.5) -- (2,0);  
\draw[|->,>=stealth] (0,2) --  
  (1,1) ;  
\end{tikzpicture}
```

Sortie

On a accès à plusieurs effets depuis **TikZ** :

- Les dégradés avec **shade** ;
- Les sélections avec **clip**.

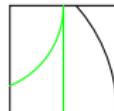
Code



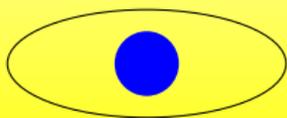
Fichier Intermediaire/code102.tex

```
\begin{tikzpicture}
\shade[left color=red, right color
=blue] (0,1) rectangle +(1,0.5) ;
\shade[top color=green, bottom
color=yellow] (0,0) rectangle
+(1,0.5) ;
\draw (2.5,0) rectangle +(1,1);
\clip (2.5,0) rectangle +(1,1);
\draw (2,0) circle (1.5) ;
\draw[green] (3,0) -- (3,1) arc
(360:200:0.8) ;
\end{tikzpicture}
```

Sortie



Sortie



Code



Fichier Intermediaire/code103.tex

```
\begin{tikzpicture}
  \shade[top color = white, bottom color = yellow] (0,0) -- (6,0) --
    (3,5.19615242);
  \draw (3,1.5) ellipse (1.3 and 0.5);
  \fill[blue] (3,1.5) circle (0.3);
\end{tikzpicture}
```

TikZ, de même que `pstricks` a une commande pour faire des choses répétées.

Code

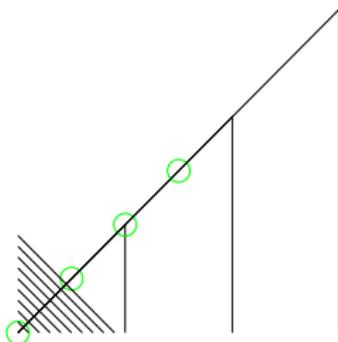


Fichier Intermediaire/code104.tex

```
\begin{tikzpicture}
\foreach \x in {0,1,2,3} {
\draw[green] (0.5*\x,0.5*\x)
circle (3pt) ;
\draw (0,0) -- (\x,\x) -- (\x,0)
;}

\foreach \x in {0, 0.1, ..., 1} \
draw (\x,0) -- (0, \x) ;
\end{tikzpicture}
```

Sortie



Attention à ne pas en abuser, on peut très vite épuiser la mémoire de TEX , qui n'a pas été conçu pour faire ce type de programmation !

Nœuds et texte

Enfin, on peut, à l'aide de la commande `node`, écrire du texte et faire des diagrammes.

Un nœud est simplement un objet qui peut contenir du texte, ou une forme géométrique.

Code

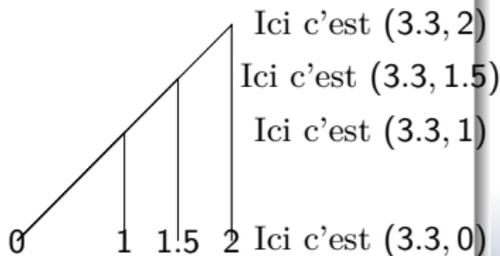


Fichier Intermediaire/code105.tex

```

\begin{tikzpicture}
\foreach \x in {0,1,1.5,2} {
\draw (0,0) -- (\x,\x) -- (\x,0)
  node { $\x$ };
\draw (3.3,\x) node [shape=
  rectangle] {Ici c'est  $(3.3,\x)$ };
}
\end{tikzpicture}
  
```

Sortie



Nœuds et texte

Il est possible de placer un nœud relativement à un point. Cela est obtenu simplement en passant une option de plus.

De même, on peut placer un nœud dans une courbe.

Code

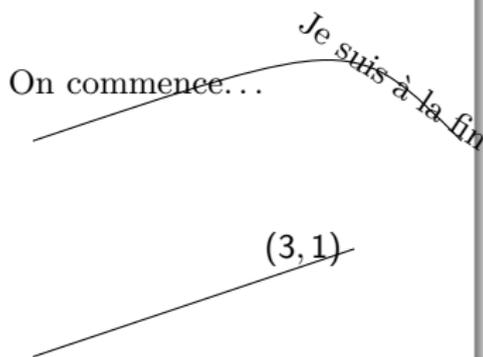


Fichier Intermediaire/code106.tex

```

\begin{tikzpicture}
\draw (0,0) -- (3,1) node [anchor=
east] {$(3,1)$} ;
\draw (0,2) .. controls (3,3) ..
node [near end,sloped] {Je suis
\'a la fin}
node [very near start,above] {On
commence\ldots} (4,2) ;
\end{tikzpicture}
  
```

Sortie



Nœuds et texte

Les diagrammes se font avec des nœuds nommés (entre parenthèses), avec un style optionnel — mais qui aide beaucoup à la conception — et un placement relatif avec espacement minimal.

Code



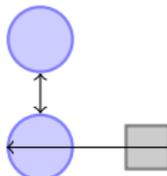
Fichier Intermediaire/code107.tex

```

\begin{tikzpicture}
[place/.style={circle,draw=blue
!50,fill=blue!20,thick,inner sep
=0pt,minimum size=6mm},
transition/.style={rectangle,draw
=black!50,fill=black!20,thick,
inner sep=0pt,minimum size=4mm}]
\node[place]      (waiting)
{};
\node[place]      (critical)
[below of=waiting] {};
\node[transition] (leave
critical) [right of=critical]
{};
\draw [->] (leave critical.east)
-- (critical.west);
\draw [<->] (critical) -- (
waiting);
\end{tikzpicture}

```

Sortie



Graphiques avec gnuplot

TikZ offre plusieurs façons de dessiner des graphes ; l'utilisation de **gnuplot** donne toute la puissance de cet outil, directement conçu pour cela, facilement accessible en **TikZ**.

Code

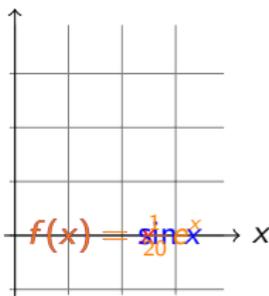


Fichier Intermediaire/code108.tex

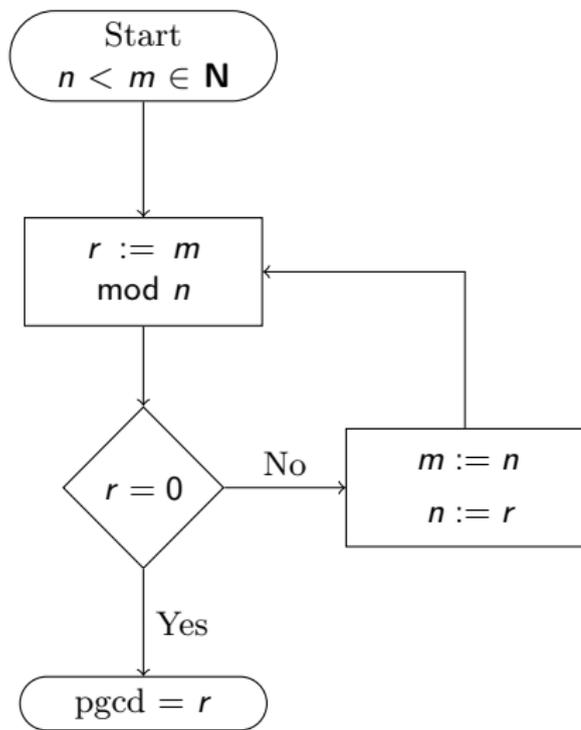
```

\begin{tikzpicture}[domain=0:4,
  scale=0.5]
\draw[very thin,color=gray]
  (-0.1,-1.1) grid (3.9,3.9);
\draw[->] (-0.2,0) -- (4.2,0) node
  [right] {$x$};
\draw[->] (0,-1.2) -- (0,4.2) node
  [above] {$f(x)$};
\draw[color=red] plot[id=x]
  function{x}
  node[right] {$f(x) = x$};
\draw[color=blue] plot[id=sin]
  function{sin(x)}
  node[right] {$f(x) = \sin x$};
\draw[color=orange] plot[id=exp]
  function{0.05*exp(x)}
  node[right] {$f(x) = \frac{
  1}{20} e^x$};
\end{tikzpicture}
    
```

Sortie

 $f(x)$


Sortie



Code



Fichier Intermediaire/code109.tex

```

\begin{tikzpicture}
%Styles
[node distance=2cm,
every text node part/.style={text centered},
start/.style={rounded rectangle, draw, text width=2cm},
end/.style={rounded rectangle, draw, text width=2cm},
operation/.style={rectangle, draw, text width=2cm, minimum height =
1cm},
decision/.style={diamond, draw, minimum size=1.5cm},
scale=0.4]
% The graphic
\node[start]      (start)                {Start \\ $n < m
  \in \mathbf{N}$};
\node[operation] (remainder) [below of=start]  {$r := m \bmod n
  $};
\node[decision]  (testRem)   [below of=remainder] {$r = 0$};
\node[operation] (update)    [right of=testRem, node distance=3cm]
  {$\begin{aligned} m &:= n \\ n &:= r \end{aligned}$};
\node[operation] (invisible) [right of=remainder, node distance=3
  cm, draw=white] {};
\node[end]      (result)    [below of=testRem] {pgcd = $r$};
\draw [->] (start.south) -- (remainder.north);
\draw [->] (remainder) -- (testRem);
\draw [->] (testRem) -- (update) node [above, midway] {No};
\draw [->] (update.north) |- (remainder.east) ;
\draw [->] (testRem) -- (result) node [right, midway] {Yes};
    
```



Un peu de graphisme

Table des matières



Utilisation du package chemfig

- Une aide très bien faite :
http://www.tex.ac.uk/ctan/macros/generic/chemfig/chemfig_doc_en.pdf
- Quelques exemples et un peu de syntaxe

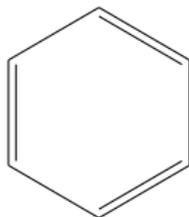
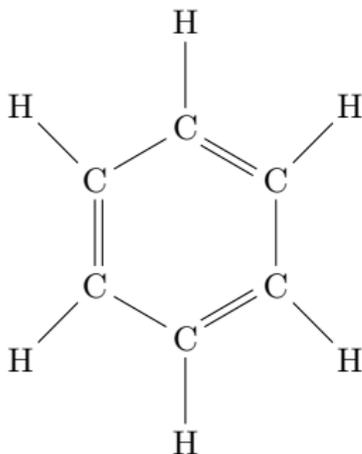
Code



Fichier Intermediaire/codel10.tex

```
\chemfig{C*6(-C(-[6]H)=C(-[7]H  
)-C(-[1]H)=C(-[2]H)-C(-[3]H)  
=) (-[5]H)}\quad  
\chemfig{*6(-----)}
```

Sortie



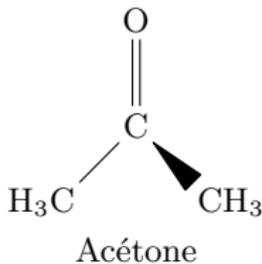
Code



Fichier Intermediaire/code111.tex

```
\definesubmol{CH3}[H_3C]{CH_3}  
\chemname{\chemfig{C(=[2]O)  
(<[7]!{CH3})(-[5]!{CH3})}}{Ac  
\'etone}
```

Sortie



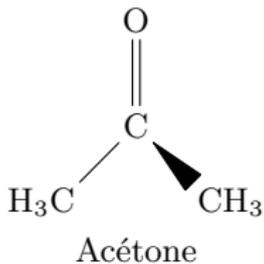
Code



Fichier Intermediaire/code112.tex

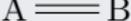
```
\definesubmol{CH3}[H_3C]{CH_3}  
\chemname{\chemfig{C(=[2]O)  
(<[7]!{CH3})(-[5]!{CH3})}}{Ac  
\'etone}
```

Sortie



Principales commandes de chemfig

Les principales commandes

Bond #	Code	Result	Bond type
1	<code>\chemfig{AB}</code>	AB	Single
2	<code>\chemfig{A=B}</code>	A  B	Double
3	<code>\chemfig{A~B}</code>	A  B	Triple
4	<code>\chemfig{A>B}</code>	A  B	right Cram, plain
5	<code>\chemfig{A<B}</code>	A  B	left Cram, plain
6	<code>\chemfig{A>:B}</code>	A  B	right Cram, dashed
7	<code>\chemfig{A<:B}</code>	A  B	left Cram, dashed
8	<code>\chemfig{A> B}</code>	A  B	right Cram, hollow
9	<code>\chemfig{A< B}</code>	A  B	left Cram, hollow

Quelques indices et angles prédéfinis

Code



Fichier Intermediaire/code113.tex

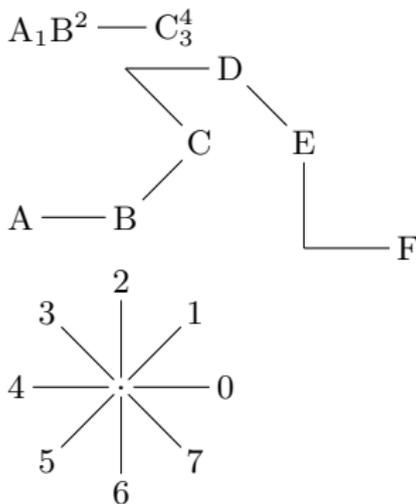
```

\chemfig{A_1B^2 - C_3 ^ 4}

\chemfig{A-B-[1]C-[3]-D-[7]E
-[6]-F}

\chemfig{.(-0)(-[1]1)(-[2]2)
(-[3]3)(-[4]4)(-[5]5)
(-[6]6)(-[7]7)}
    
```

Sortie



Principales commandes de chemfig

Angles absolus [:<angle>]

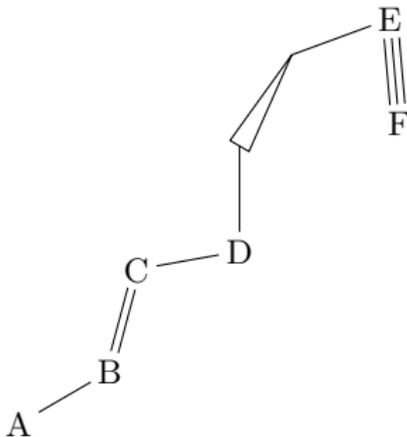
Code



Fichier Intermediaire/code114.tex

```
\chemfig{A-[:30]B=[:75]C-[:10]
D-[:90]>|[:60]-[:20]E~[:275]F
}
```

Sortie



Principales commandes de chemfig

Angles relatifs [::<angle>]

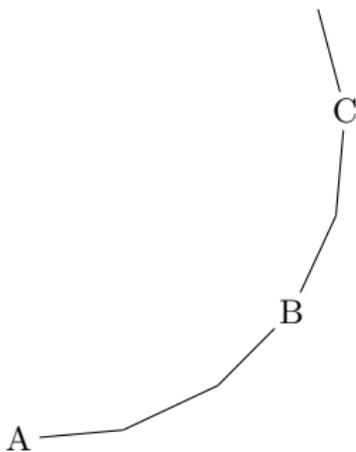
Code



Fichier Intermediaire/codel15.tex

```
\chemfig{A-[:5]-[::+20]-[::20]  
B-[::+20]-[::20]C-[::20]}
```

Sortie



Principales commandes de chemfig

Modifier les points de connection [*i*, <i><j>

Code

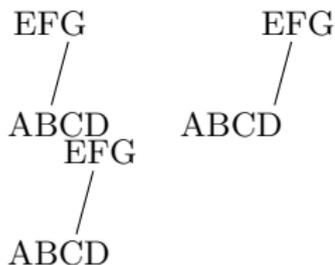


Fichier Intermediaire/code116.tex

```

\chemfig{ABCD-[:75,,2,3]EFG}\
  qquad
\chemfig{ABCD-[:75,,2]EFG}\
  qquad
\chemfig{ABCD-[:75,,3,2]EFG}
  
```

Sortie



Principales commandes de chemfig

Personnaliser les liens [,,, <tikzcode>]

Code



Fichier Intermediaire/code117.tex

```
\chemfig{A-[,,,red]B}\par  
\chemfig{A-[,,,dash pattern=  
on 2pt off 2pt]B}\par  
\chemfig{A-[,,,line width=2pt  
]B}\par  
\chemfig{A-[,,,red,line width  
=2pt]B}
```

Sortie

A — B

A - - - - B

A ——— B

A ——— B

Principales commandes de chemfig

Sous-molécule (<submolecule>)

Code



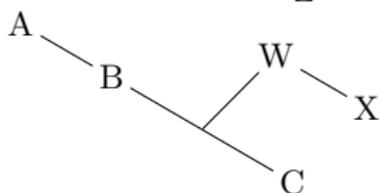
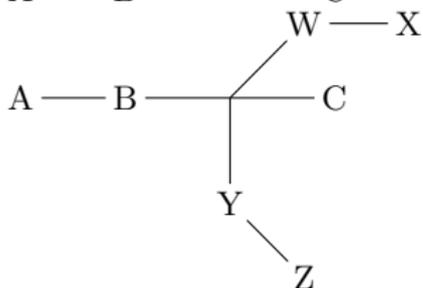
Fichier Intermediaire/code118.tex

```
\chemfig{A-B-(-[1]W-X)-C}
```

```
\chemfig{A-B-(-[1]W-X)(-[6]Y  
-[7]Z)-C}
```

```
\chemfig{[: -30]A-B-(-[1]W-X)-C  
}
```

Sortie



Principales commandes de chemfig

Connexion entre atomes lointains ? [<name>, <bond>, >tikz>]

Code

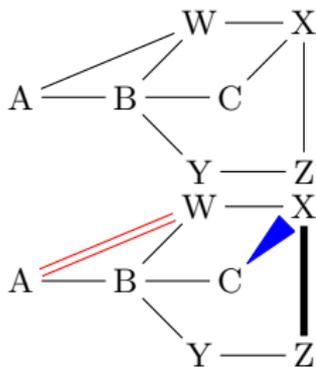


Fichier Intermediaire/code119.tex

```
\chemfig{A?[a]-B(-[1]W?[a]-X?[b])
(-[7]Y-Z?[b])-C?[b]}

\chemfig{A?[a]-B(-[1]W?[a,2,red]-X
?[b])(-[7]Y-Z?[b,1,{line width=2
pt}))-C?[b,>],blue]}
```

Sortie



Principales commandes de chemfig

Formes régulières `<atom>*<n>(<code>)`

Code



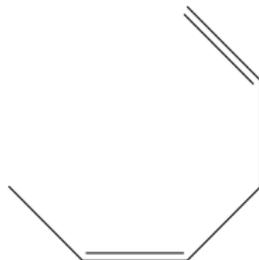
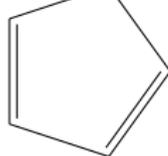
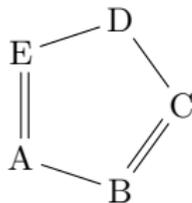
Fichier Intermediaire/code120.tex

```
\chemfig{A*5(-B=C-D-E=)}
```

```
\chemfig{*5(-----)}
```

```
\chemfig{*8(-----)}
```

Sortie



Exercice

- Écrire le code permettant d'obtenir ceci

