



UNIVERSITÉ D'AVIGNON  
ET DES PAYS DE VAUCLUSE

C E N T R E  
D'ENSEIGNEMENT  
ET DE RECHERCHE  
EN INFORMATIQUE



Nom de la formation  
Nom du parcours/spécialité  
UE Nom de l'unité d'enseignement

## Titre du document

1<sup>er</sup> décembre 2017  
Groupe de projet

Prénom1 Nom1  
Prénom2 Nom2  
Prénom3 Nom3

CERI - LIA  
339 chemin des Meinajariès  
BP 1228  
84911 AVIGNON Cedex 9  
France

Tél. +33 (0)4 90 84 35 00  
Fax +33 (0)4 90 84 35 01  
<http://ceri.univ-avignon.fr>

# Sommaire

Titre . . . . .	1
Sommaire . . . . .	2
1 Introduction . . . . .	3
2 Mise en place du système . . . . .	3
2.1 Installation . . . . .	3
2.2 Configuration . . . . .	4
2.3 Compilation . . . . .	5
2.4 Application Web . . . . .	5
3 Généralités . . . . .	6
3.1 Mise en forme de base . . . . .	6
3.2 Structure . . . . .	7
3.2.1 Section de niveau 3 . . . . .	7
3.3 Références croisées . . . . .	7
4 Objets . . . . .	7
4.1 Tables . . . . .	8
4.2 Équations . . . . .	9
4.3 Figures . . . . .	9
4.4 Algorithmes . . . . .	10
5 Références bibliographiques . . . . .	11
6 Conclusion . . . . .	13
Bibliographie . . . . .	14

## 1 Introduction

L'utilisation de logiciels tels que MS Word ou LibreOffice pour la rédaction de rapports aboutit généralement à des documents très hétérogènes, à la lisibilité douteuse, ce qui entraîne des difficultés lors de leur évaluation. Pour cette raison, c'est  $\LaTeX$  [6, 18] qui a été retenu pour les rapports de cette UE.

La mise en forme d'un document est aussi importante que son contenu. Il est clair qu'un rapport bien présenté mais dont le contenu est inadapté ou incorrect obtiendra une note faible. Mais de la même façon, un rapport dont le contenu est excellent et dont la présentation est mauvaise récoltera probablement une mauvaise note, car il sera difficile d'en comprendre les idées. Donc pour résumer, une bonne présentation est une condition nécessaire, mais pas suffisante pour produire un bon rapport. Tout ça pour dire que la mise en forme de vos rapports sera évaluée au même titre que leur contenu.

$\LaTeX$ , pour simplifier, est un langage permettant de programmer la mise en forme d'un document. Donc, au lieu de composer votre document dans un traitement de texte, vous allez écrire seulement sa structure et son contenu sous la forme d'un code source (extension `.tex`). Vous devrez ensuite compiler ce code source pour obtenir un fichier `.pdf`.

$\LaTeX$  et les outils associés sont libres (et gratuits). Il s'agit d'un standard dans le domaine de la recherche académique : la plupart des articles soumis à des conférences ou à des journaux scientifiques sont mis en forme grâce à ce système.

Bien sûr, le but de ce cours n'est pas d'apprendre à utiliser  $\LaTeX$ , c'est pourquoi le présent document tient lieu à la fois de tutoriel et de modèle pour la rédaction de rapports. Il ne s'agit pas d'un manuel sur  $\LaTeX$  : de nombreuses ressources disponibles sur le Web vous apprendront tout ce que vous devez savoir sur les bases de  $\LaTeX$ . Si vous n'avez jamais utilisé  $\LaTeX$ , il est recommandé de lire un texte d'introduction tel que ces WikiBooks en français [15] ou en anglais [16], en complément à ce document.

Le code source  $\LaTeX$  de ce document vous est fourni afin que vous l'utilisiez comme base pour vos propres rapports. Il est abondamment commenté afin d'en faciliter la compréhension. Il est recommandé de lire la version PDF et la version  $\LaTeX$  simultanément pour bien en comprendre le fonctionnement. Pour écrire votre propre rapport, il est conseillé de faire une copie du fichier `.tex` et de s'en servir de base après avoir effacé les parties inutiles et les commentaires.

## 2 Mise en place du système

On peut distinguer deux façons de produire un document  $\LaTeX$  : soit via une application locale, soit grâce à une application Web. Les trois premières sous-sections suivantes sont consacrées à la première approche, alors que la dernière porte sur l'utilisation de la seconde.

### 2.1 Installation

Le système  $\LaTeX$  étant libre, il est disponible sur les principaux systèmes d'exploitation (Unix, Linux, Windows, MacOS, etc.). Il est extrêmement recommandé d'utiliser Unix ou Linux, qui intègrent généralement  $\LaTeX$  par défaut. La version de base d'Ubuntu contient le package `TexLive201x`, qui est une distribution  $\LaTeX$  suffisante pour générer vos rapports. Elle contient, entre autres, différentes extensions de  $\LaTeX$  utilisées dans ce document. Vérifiez dans votre gestionnaire de packages que `TexLive` est bien installé, et si ce n'est pas le cas : installez-le.

Si vous utilisez Windows, la distribution  $\LaTeX$  habituellement utilisée est `MiKTeX` [9]. Une alternative peut être d'utiliser une machine virtuelle telle que `VirtualBox` [10] pour installer Ubuntu sur votre ordinateur. La machine virtuelle va simuler le comportement d'un ordinateur, ce qui vous permettra d'exécuter Ubuntu depuis un autre OS. L'inconvénient est que votre machine doit être assez puissante pour pouvoir gérer les calculs supplémentaires dus à l'émulation. Autrement dit : ça sera peut-être un peu lent sur un PC portable.

Quel que soit le système d'exploitation, vous avez besoin d'un package  $\text{\LaTeX}$  permettant d'écrire des documents en français. Il s'agit du package `texlive-lang-french`, que vous pouvez installer depuis le gestionnaire de packages lui aussi (Ubuntu) ou à partir du gestionnaire de packages de MiKTeX (Windows). Sans ce package, vous ne pourrez pas compiler le document.

Vous avez aussi besoin d'un éditeur de texte. Vous pouvez utiliser n'importe lequel, peu importe. Par exemple, `gedit` qui est généralement installé avec Ubuntu convient tout à fait. Cependant, pour plus de confort, il est très fortement recommandé d'installer plutôt un éditeur adapté à  $\text{\LaTeX}$ . Le logiciel libre `Texmaker` [13] est suffisant, et c'est lui qui sera utilisé dans les descriptions données dans ce document. A noter que ce logiciel existe aussi pour Windows et d'autres OS. Il apporte différentes fonctionnalités bien pratiques, comme la possibilité de facilement compiler les documents. Sous Ubuntu, vous pouvez l'installer depuis le gestionnaire de packages. Notez que le logiciel que vous utiliserez pour éditer vos documents devra obligatoirement disposer d'un correcteur orthographique : c'est le cas de `Texmaker` (cf. la Section 2.2 pour savoir comment le configurer).

Enfin, il est possible (et recommandé) d'utiliser un logiciel spécifique pour gérer la bibliographie, tel que `JabRef` [4]. Ce logiciel libre peut être installé à partir du gestionnaire de packages (il existe aussi pour Windows ou Mac OS, car il est programmé en Java). Il permet d'ajouter/supprimer/éditer des références bibliographiques, et de générer un fichier au format BibTeX (voir la Section 5).

## 2.2 Configuration

L'essentiel du travail de configuration pour  $\text{\LaTeX}$  et `Texmaker` a été effectué lors de l'installation, donc il ne reste pas grand chose à faire. Vous devez d'abord modifier une option de `Texmaker`. Allez dans (*Options / Configure Texmaker*) et localisez l'option de BibTeX (3<sup>e</sup> ligne). Modifiez cette option de manière à avoir "`bibtex %`" à la place de "`bibtex %.aux`" (voir la Figure 1).

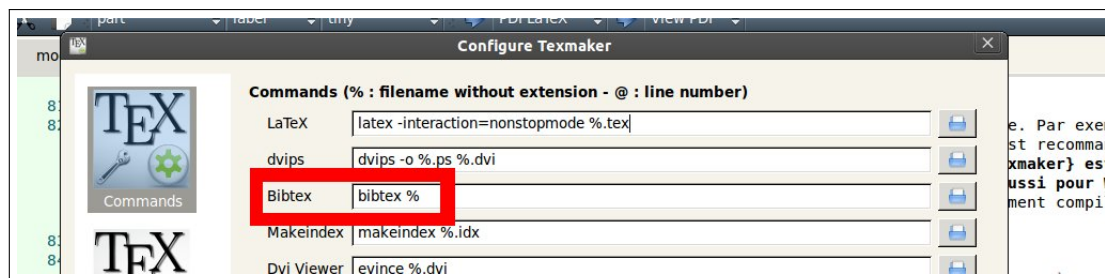


Figure 1. Il est nécessaire de modifier l'option de BibTeX dans `Texmaker`.

Vous devez ensuite configurer `Texmaker` pour qu'il utilise le dictionnaire français lors de la vérification orthographique. Allez dans *Options / Configure Texmaker / Editor / Spelling Dictionnary* et cliquez sur le bouton pour sélectionner le fichier `fr_FR.dic` (c.f. Figure 2). Si ce fichier n'est pas disponible, vous devez installer depuis le gestionnaire de packages ou MiKTeX un package dont le nom est du type `myspell-fr`. Une fois le dictionnaire sélectionné dans `Texmaker`, les fautes d'orthographe seront soulignées en rouge. Attention, le logiciel ne détecte pas les fautes de grammaire ou d'accord (genre, pluriel, conjugaison...). Soyez donc attentifs à ce type d'erreurs!

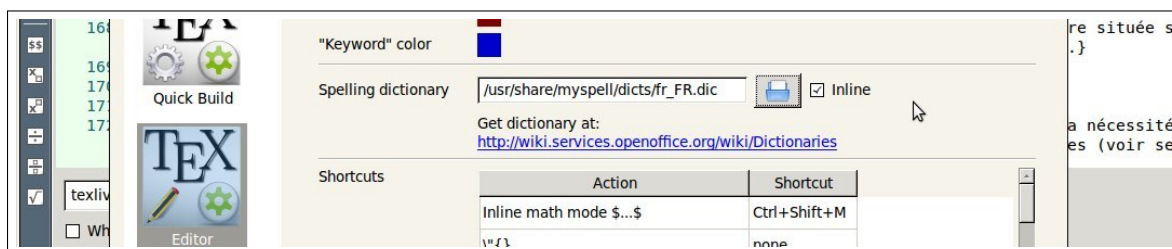


Figure 2. Configuration du dictionnaire utilisé par le vérificateur d'orthographe.

Pour pouvoir générer directement un fichier PDF, vous devez sélectionner le compilateur approprié. Dans la barre d'outil de Texmaker, cliquez sur *LaTeX* et sélectionnez à la place *PDFLaTeX*. Quand vous aurez besoin de compiler un document avec BibTeX (cf. Section 5), procédez de la même manière avec l'option BibTeX.

## 2.3 Compilation

Texmaker facilite grandement la création de documents  $\LaTeX$ . Il faut d'abord créer un nouveau fichier, y écrire le code source  $\LaTeX$  et l'enregistrer. Un exemple de code minimal est donné ci-dessous, que vous pouvez utiliser pour tester votre installation :

```
\documentclass{article}
\begin{document}
  Ceci est un document.
\end{document}
```

Il suffit ensuite de cliquer sur la flèche située à gauche de *PDFLaTeX* dans la barre d'outils de Texmaker, comme indiqué dans la Figure 3, pour lancer la compilation du document. Vous pouvez également tester votre installation sur le fichier `.tex` fourni avec ce document : vous devez obtenir exactement le même PDF, puisqu'il a été généré de cette façon.

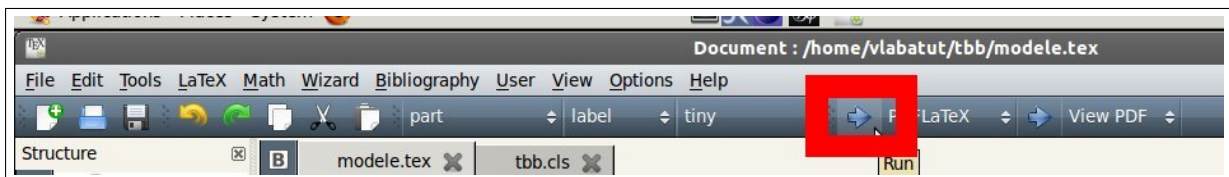


Figure 3. Le fait de cliquer sur la flèche indiquée déclenche la compilation du document  $\LaTeX$ .

Si la compilation rencontre des erreurs, celles-ci seront affichées dans la fenêtre *Messages / Log File* située en bas de Texmaker (Figure 4). Sinon, vous pourrez visualiser le document PDF en cliquant sur la flèche située à gauche de *View PDF* (dans la barre d'outils). Si, lors du premier test, le compilateur ne trouve pas le fichier de classe `ceri.cls`, vérifiez que vous l'avez bien placé dans le même dossier que votre fichier `.tex`.

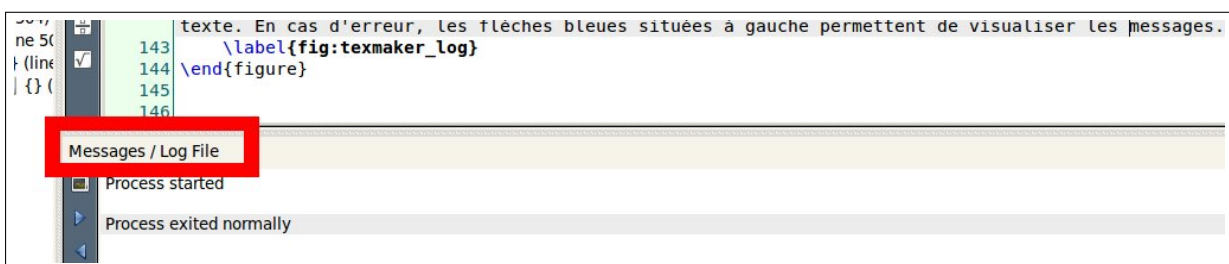


Figure 4. Texmaker affiche le résultat de la compilation dans la fenêtre située sous l'éditeur de texte. En cas d'erreur, les flèches bleues situées à gauche permettent de visualiser les messages.

A noter qu'il est possible que le compilateur indique à l'aide d'un avertissement (warning) la nécessité de compiler une autre fois le document. C'est notamment le cas si des références bibliographiques ont été modifiées (voir Section 5) ou si la structure du document a changé (cf. Section 3.2).

## 2.4 Application Web

Si vous ne voulez pas installer et configurer  $\LaTeX$  (ainsi que l'éditeur qui va avec) sur votre système, vous pouvez à la place vous tourner vers une application Web du type *OverLeaf*. Vous

devez d'abord vous créer un compte (c'est gratuit), puis vous pourrez définir et compiler vos propres documents en ligne.

Notez que le source L<sup>A</sup>T<sub>E</sub>X du présent document est disponible sur OverLeaf à l'adresse : <https://www.overleaf.com/latex/templates/modele-rapport-uapv/pdbgdpszsgwrt>. Il s'agit d'un modèle à partir duquel vous pouvez instancier votre propre document.

L'inconvénient principal de l'application en ligne (par rapport à l'application locale) est que la compilation prend sensiblement plus de temps. Par contre, les avantages sont multiples : pas d'installation requise, simplicité d'utilisation, utilisable sur la plupart des navigateurs, etc. De plus, OverLeaf permet le travail collaboratif : partage de documents, suivi des modifications, gestion des versions, système de commentaires, etc. Différents tutoriels sont disponibles sur le site officiel.

## 3 Généralités

Vous devez indiquer le titre du document grâce à la commande `\title` définie en début de fichier. Les noms des auteurs sont à spécifier dans la commande `\author`.

### 3.1 Mise en forme de base

La mise en forme de base est décrite clairement dans les références déjà citées [15, 16]. Voici rapidement quelques éléments utiles (référez-vous au code source pour comprendre le fonctionnement de ces commandes).

Le texte peut être mis en **gras** avec `\textbf` ou en *italique* avec `\textit`. Tous les identificateurs (i.e. les noms de classes, méthodes, variables, constantes...) que vous citez dans le texte doivent être mis en forme avec la commande `\texttt`, de manière à obtenir un résultat du type `MyClass`. Ceci permet de facilement repérer les identificateurs à la lecture de votre document.

Il est possible d'inclure des listes non-numérotées avec `\begin{itemize}` et `\item`, telles que :

- premier élément ;
- 2<sup>e</sup> élément ;
- troisième élément.
  - on peut aussi placer des sous-listes ;
    - et rajouter encore d'autres niveaux ;
    - etc.
  - on continue au niveau supérieur ;
- et au niveau encore supérieur.

Ou bien des listes numérotées avec `\begin{enumerate}`, comme :

1. premier élément ;
2. 2<sup>e</sup> élément ;
3. troisième élément.
  - (a) là encore, on peut placer des sous-listes ;
  - (b) ça marche exactement pareil que pour les listes non-numérotées ;
    - on peut même mélanger les deux types de listes comme ici ;
    - et ici.

Il est aussi possible de placer un fond coloré derrière le texte avec la commande `\colorbox{couleur}`. Pour les couleurs, vous pouvez soit utiliser le nom de la couleur en anglais, par exemple `red`, soit définir une couleur précise en utilisant des valeurs numériques.

**Remarque** : évitez au maximum les notes de bas de page (pas de commande `\footnote`) : placez vos remarques directement dans le texte, entre parenthèses.

## 3.2 Structure

Pour créer des paragraphes séparés, vous devez sauter une ligne (voir le code source). La structure du document est définie en utilisant des commandes spécifiques. Par ordre de niveau décroissant, on a :

1. `\section{Titre de la section de niveau 1}`
2. `\subsection{Titre de la section de niveau 2}`
3. `\subsubsection{Titre de la section de niveau 3}`
4. `\paragraph{Titre de la section de niveau 4}`

Les deux derniers niveaux ne sont pas réellement utilisés dans ce document, car ils n'y sont pas nécessaires. Pour l'exemple, les voici :

### 3.2.1 Section de niveau 3

(Il est nécessaire de mettre une section de niveau 3 entre la section de niveau 2 et celle de niveau 4, sinon une section factice de niveau 3 numérotée 0 sera automatiquement créée).

**Section de niveau 4.** Le niveau 4 n'est pas vraiment une section mais plutôt un titre de paragraphe. Pour cette raison, le titre n'est pas sur une ligne séparée, mais il est directement placé dans le paragraphe concerné.

**Remarques :** ne définissez pas une section seulement pour y placer 1 ou 2 lignes de texte. Une section doit avoir un contenu assez long, sinon ce n'est pas la peine d'en créer une.

## 3.3 Références croisées

$\LaTeX$  permet de définir un label pour marquer un endroit particulier du document. Il est ensuite possible de faire référence à ce label. La référence apparaîtra dans le texte sous la forme d'un lien hypertexte, ce qui est pratique pour naviguer dans le document.

Un label se définit avec la commande `\label{nomlabel}` et une référence à un label avec `\ref{nomlabel}`. Si le label correspond à un objet numéroté (figure, section, table...), alors c'est son numéro qui sera affiché.

La convention veut qu'on utilise des préfixes spécifiques dans les noms des labels en fonction du type d'objet référencé. Vous devez utiliser cette convention dans vos rapports (voir la Table 1).

Type d'objet	Préfixe
section	sec
figure	fig
table	tab
equation	eq

**Table 1.** Préfixes utilisés conventionnellement pour définir les noms des labels.

Il est absolument obligatoire que tout objet (équation, table, figure...) numéroté soit cité dans votre texte. Donc chacun doit avoir son label, et ce label doit apparaître au moins une fois dans le texte. Bien sûr vous ne devez pas vous contenter de citer chaque objet : ils doivent être suffisamment décrits, et de façon précise. Un objet qui n'est pas mentionné dans le texte ne sert à rien.

## 4 Objets

$\LaTeX$  permet d'insérer un certain nombre d'objets numérotés : tables, équations, figures et algorithmes. Ces objets doivent toujours posséder une légende (sauf les équations) et être numérotés.

De plus, ils doivent être mentionnés, décrits et commentés dans le texte. Toute référence à un objet doit être faite en utilisant la commande `\ref{nomlabel}` (cf. Section 3.3).

$\LaTeX$  se charge de placer les objets dans le document. Il est possible de restreindre le positionnement en précisant si on veut que l'objet en question soit placé à l'endroit où il se trouve dans le document original (`h` pour *here*), en haut de page (`t` pour *top*) ou en bas de page (`b` pour *bottom*). Il est possible de combiner ces valeurs si on veut donner plus de liberté au compilateur (ce qui est recommandé). Il est aussi possible que le compilateur place les objets ailleurs s'il est impossible de les placer à l'endroit spécifié.

En plus du positionnement, les légendes sont également gérées de façon automatique. Elles sont déclarées dans les objets, à l'aide de la commande `\caption{légende}`.

## 4.1 Tables

La syntaxe permettant de définir des tables en  $\LaTeX$  est relativement compliquée. Vous vous limiterez à des tables de la forme décrite dans cette sous-section : cette mise en forme doit être respectée, afin de garder des tables lisibles. La commande `\begin{table}[position]` permet d'insérer une table dans le document. Le paramètre `position` correspond à une combinaison de `h`, `t` et `b` qui permettra au compilateur de placer la table dans le document. Comme indiqué auparavant, le mieux est d'utiliser les trois, c'est à dire `htb`.

Chaque table doit être centrée, ce qui est réalisé avec la commande `\centering`. La première ligne de la table contient les titres des colonnes. Le texte contenu dans cette ligne de titre doit être en **gras**. Les traits de séparation sont dessinées avec `\hline`.

Le contenu proprement dit de la table est défini grâce à la commande `\begin{tabular}{colonnes}`. Le paramètre `colonnes` permet de préciser à la fois le nombre de colonnes contenues dans la table, et leur alignement :

- `l` (*left*) pour une colonne alignée à gauche ;
- `r` (*right*) pour qu'elle soit alignée à droite ;
- `c` (*center*) pour qu'elle soit centrée.

Les données de la table sont ensuite décrites ligne par ligne : la valeur de chaque colonne est séparée par un caractère `&` et la fin de la ligne est marquée par un `\\` (obligatoire).

Vos tables doivent ressembler exactement à celles données en exemple dans ce document : couleurs, séparations, légende située en-dessous, etc. Les valeurs numériques doivent être alignées à droite. Les réels doivent être définis de manière à avoir le même nombre de décimales. Les valeurs textuelles peuvent être alignées à droite ou à gauche, ou bien centrées. Il est possible d'insérer des équations en-ligne (i.e. non-numérotées) dans une table (voir la Section 4.2). La Table 2 constitue un autre exemple à prendre en compte.

<b>Texte</b>	<b>Texte</b>	<b>Entiers</b>	<b>Réels</b>	<b>Formules</b>
Blabla bla bla bla bla	Blabla bbla bla	21	12,62	$a + b = c$
Blabla bla bla bla	Blabla bla bla bla bla	12356	3456,21	$x \cos x$
Blabla bla bla	Blabla bla bbla bla	1	45,87	$\frac{\alpha^2 + \beta}{\gamma}$
Blabla blabla bla bla	Blabla bla bla bla	456	45,16	$\sqrt{x + y}$
Blabla bla blaa bla	Blabla bla bla bla	78	89,28	$x_1 - x_2$
Blabla bla bla bla bla	Blabla bla bla bla	416	10,10	$\max_x x$
Blabla b bla bla	Blabla blbla bla bla	9635	99,99	$\frac{x}{y}$
Blabla bla bbla bla	Blabla bla b	965801	46465,45	$\sum_{x=1}^{10} xy$
Blabla bla	Blabla bla	7	48,78	$\varphi^2$

**Table 2.** Exemple de table contenant du texte, des valeurs entières, réelles et des formules mathématiques.

Si votre table est haute et étroite, alors vous devez la décomposer afin de mieux utiliser l'espace.



Ceci est illustré avec la Table 3, qui contient 50 lignes pour seulement 2 colonnes. Ceci est réalisé en utilisant l'environnement `minipage`, cf. le code source de ce document.

Col1	Col2	Col1	Col2	Col1	Col2	Col1	Col2	Col1	Col2
1	hdc	11	ytr	21	ioe	31	zyc	41	oze
2	orf	12	xcv	22	snu	32	opv	42	anb
3	seh	13	fsd	23	dgp	33	azv	43	kir
4	lig	14	pui	24	zbu	34	xdf	44	sfk
5	jhg	15	gsf	25	zvv	35	yzd	45	bed
6	azr	16	aze	26	orv	36	sgg	46	pyj
7	opi	17	bcv	27	czr	37	sss	47	zdz
8	hgj	18	wxc	28	azh	38	zdv	48	pyj
9	xvc	19	uyr	29	cyh	39	sgu	49	zen
10	ree	20	ndz	30	ytj	40	vzd	50	oef

**Table 3.** Exemple de table haute et étroite, décomposée en 5 parties pour occuper plus efficacement l'espace disponible.

## 4.2 Équations

Les équations peuvent être introduites sous deux formes : *en-ligne* et *hors-ligne*. Une équation en-ligne est intégrée au texte, par exemple  $y = ax + b$ . La syntaxe  $\LaTeX$  consiste simplement à l'entourer avec des caractères `$` : le 1<sup>er</sup> `$` active ce qu'on appelle le *mode mathématique* et le 2<sup>e</sup> le désactive. Le texte situé entre les deux est interprété de façon différente du texte normal. Cette méthode permet aussi d'insérer des formules dans les tables, comme par exemple dans la Table 2. Voyez [15, 16] pour plus de détails sur les nombreuses possibilités de  $\LaTeX$  en ce qui concerne les formules mathématiques, et le code source de la Table 2 pour quelques exemples.

$$f(s) = h(s) + \max(g_1(s), g_2(s)) \quad (1)$$

Les équations hors-ligne sont séparées du texte et numérotées. On les définit en utilisant la commande `\begin{equation}`. En y associant un label, il est donc possible d'y faire référence dans le texte. Par exemple, pour faire référence à celle située ci-dessus, on parlera de l'équation (1), ou pour faire court : `eq.(1)`. Une équation n'a pas de légende, et elle doit être obligatoirement décrite dans le texte qui y fait référence.

Les matrices ne doivent pas être représentées par des tables : le mode mathématique est prévu à cet effet, comme montré par l'`eq.(2)`.

$$M = \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix} \quad (2)$$

### Remarques :

- Vos variables et fonctions doivent avoir des noms courts : une lettre et un indice, maximum. Pas de nom du type *ma fonction*, mais plutôt  $f$ ,  $f'$ ,  $f_1$ , etc.
- Ne mélangez pas les notations informatiques et mathématiques. Par exemple n'utilisez pas la notation tableau (`array`), comme par exemple `tab[12]`, dans une équation.

## 4.3 Figures

Une figure s'insère de manière très similaire à une table, excepté qu'on utilise la commande `\begin{figure}` au lieu de `\begin{table}`. A part ça, la commande `\centering` doit toujours



**Figure 5.** La façade de l'Université d'Avignon.

être présente pour centrer la figure, la légende doit être définie en-dessous de la figure avec `\caption`, et un label peut être créé avec `\label` pour référencer la figure, comme par exemple la Figure 1.

L'utilisation du compilateur *PDFLaTeX* permet de traiter seulement des fichiers aux formats PNG, JPEG, GIF et PDF. L'image est spécifiée au moyen de la commande `\includegraphics[scale=s]{fichier}`, où `fichier` est le chemin du fichier contenant l'image (les chemins relatifs sont exprimés par rapport à l'emplacement du fichier `.tex` en cours de compilation) et `s` est une valeur réelle indiquant l'échelle de l'image (optionnel). A la place du paramètre `scale`, vous pouvez spécifier la largeur ou la hauteur de l'image, par exemple avec `width=12cm` ou `height=5cm`.

#### 4.4 Algorithmes

Tous les algorithmes doivent être donnés sous forme de diagrammes du genre *flowchart* [17], qui apparaîtront dans le rapport en tant que *figures*. Ne donnez vos algorithmes sous forme de pseudo-code que si cela vous a été explicitement demandé. Dans ce cas là, on utilisera le package `algorithm2e` (déjà inclus dans la classe  $\LaTeX$  de ce document) manière à obtenir un résultat du type de l'Algorithme 1. Pour plus de détails, voyez le code source situé juste après ce paragraphe.

Notez qu'il est possible de faire référence dans le texte à la fois à un algorithme entier, par exemple l'Algorithme 1, mais aussi à une ligne dans un algorithme donné, par exemple la ligne 6 (pour laquelle on a, au préalable, inséré un `label` dans l'algorithme, cf. le code source).

En plus de cette description formelle (le pseudo-code ou le flowchart), il est nécessaire de décrire l'algorithme informellement, c'est-à-dire avec du texte, afin d'expliquer son fonctionnement en détail de manière à bien le comprendre. En particulier, il faut lister et décrire chaque variable utilisée dans l'algorithme. Il est également recommandé de donner un exemple d'application sur des données simples.

Vous devez utiliser au maximum les notations mathématiques dans votre pseudo-code, car cela permet de le rendre plus compact et plus lisible. En particulier, choisissez des noms de variables d'une seule lettre (possiblement une lettre grecque) et non pas des expressions à rallonge vous le feriez dans un programme informatique. Vous pouvez par contre utiliser des noms longs pour les fonctions, pour des raisons d'intelligibilité.

```

Data :  $x, y$ 
Result :  $M$ 

// Initialisation
1  $z \leftarrow 50$ ;

// Boucle principale
2 for  $i \leftarrow 1$  to  $x$  do
3   while  $y \leq 18$  do
4     for  $u$  in  $V$  do
5       if  $u \neq v$  then
6          $a \leftarrow a + 1$ ;
7          $y \leftarrow x + y/2$ ;
8       end
9       if  $x > y$  then
10         $x \leftarrow x \times y$ ;
11      else
12         $y \leftarrow \sqrt{x + y/2}$ ;
13      end
14    end
15  end
    // Un autre commentaire
16   $B \leftarrow (x + y)^a$ ;
17 end
18  $M \leftarrow \|B\|$ ;

```

**Algorithme 1 :** Exemple d'algorithme bidon, représenté sous forme de pseudo-code en utilisant le package `algorithm2e`.

## 5 Références bibliographiques

Le système  $\LaTeX$  peut être couplé avec BibTeX afin de gérer automatiquement les références bibliographiques du document. La procédure consiste d'abord à définir un fichier de type BibTeX décrivant chaque référence en détails. Comme indiqué précédemment, il est recommandé d'utiliser un logiciel spécifique comme JabRef [4] pour réaliser cette tâche. Le fichier `bibliographie.bib` fourni avec ce document est un exemple de fichier BibTeX. Il est placé dans le même dossier que le fichier `.tex` à compiler.

Une référence bibliographique doit toujours indiquer certaines informations, qui dépendent du type de la référence. Ces informations nécessaires sont détaillées dans la Table 4. Les principaux types sont (avec des exemples à consulter dans le document BibTeX associé à ce modèle/tutoriel) :

- **Article** : article publié dans un journal [1, 2, 20] ;
- **Book** : livre complet [7] ;
- **Electronic** : page Web ou autre ressource disponible exclusivement en ligne [15] ;
- **InCollection** : chapitre dans un recueil [5] ;
- **InProceedings** : article publié dans les actes d'une conférence [8, 14] ;
- **MasterThesis** et **PhDThesis** : mémoire de master ou de doctorat [3, 19] ;
- **TechReport** : travail publié en interne (à une université, entreprise, etc.) ou documentation technique [11, 12] ;

Dans tous les cas, il faut indiquer le DOI s'il y en a un, et *sinon* l'URL pour pouvoir récupérer facilement le document concerné.

Dans le fichier `.tex`, une référence est introduite au moyen de la commande `\cite{cle}`, où `cle` est la clé associée dans le fichier BibTeX à la référence désirée. Par exemple la commande

Type BibTeX	Description du champ	Champ BibTeX
<i>Tous les types</i>	Nom des auteurs	author
	Titre de l'article	title
	Année de publication	year
	DOI	doi
	URL ( <b>seulement</b> s'il n'y a pas de DOI)	url
Article	Nom du journal	journal
	Numéro de volume	volume
	Numéro de sous-volume	issue
	Numéros des pages	pages
Book	Titre du livre	title
	Société publiant le livre	publisher
	Ville et pays de cette même société	address
	Titre de la série (si le livre appartient à une série)	series
	Numéro de volume dans cette série (le cas échéant)	volume
Electronic	Entreprise ou institution hébergeant la ressource	organization
	URL de la ressource (obligatoire)	url
InCollection	Même champs que pour Book, avec en plus :	
	Numéro de chapitre dans le recueil	chapter
	Numéros des pages	pages
InProceedings	Personnes ayant constitué le recueil	editor
	Nom de la conférence	booktitle
	Numéro de volume	volume
InCollection	Numéros des pages	pages
	Titre du livre	booktitle
	Numéro de volume	volume
Master/PhDThesis	Numéros des pages	pages
	Université d'inscription	institution
	Type de mémoire (M1, M2, Doctorat, etc.)	type
TechReport	Entreprise/université de rattachement	institution
	Type de rapport (Technique, Scientifique, etc.)	type
	Numéro de série unique du rapport	number

**Table 4.** Champs nécessaires aux différents types de références bibliographiques BibTeX.

`\cite{Wikibooks2010}` permet d'introduire la citation suivante : [15]. Il est aussi possible d'utiliser `\textcite{cle}` pour mentionner le nom de l'auteur, par exemple pour `\textcite{Masuda2016}` on obtient : Masuda et Lambiotte [7].

Attention, n'utilisez pas de caractères accentués lorsque vous définissez une clé. Vous pouvez mettre plusieurs références dans une même commande `\cite`, par exemple `\cite{Wikibooks2010, Wikibooks2011}` donne [15, 16].

Une fois les références citées dans le fichier  $\LaTeX$ , il faut le compiler avec BibTeX (voir la Section 2.1), puis le *recompiler* avec PDFLaTeX. Tout ça se fait automatiquement si vous avez choisi d'utiliser Overleaf.

BibTeX et  $\LaTeX$  vont s'occuper de numéroter les références automatiquement. La commande `\MyBibliography` permet d'insérer une liste des références, généralement on la place à la fin du document (voir ci-dessous, dans le code source). La commande `\addbibresource{monfichier.bib}`, qui est présente en début de document, permet d'indiquer l'emplacement et le nom du fichier BibTeX à utiliser (ici : `monfichier.bib`). Pour vous aider, le fichier `bibliographie.bib` est fourni à titre d'exemple.

Toutes les informations que vous intégrez à votre rapport et qui n'ont pas été produites par

vous-même doivent être explicitement référencées. Vous devez donc indiquer tout document que vous utilisez, y compris ce qui est récupéré sur internet. Cela vaut pour tout type d'information : document texte, images, vidéos, code source, algorithmes, idées diverses.

Autrement dit, vous avez le droit de réutiliser absolument tout ce que vous désirez. Par contre, vous devez à chaque fois indiquer exactement ce que vous réutilisez, et à qui/où vous l'avez pris. De plus, vous devez expliquer ce que vous avez modifié (par exemple s'il s'agit d'un algorithme) et la nature de ces modifications.

## 6 Conclusion

Vous devez impérativement consulter le code source de ce document, qui est abondamment commenté, et compléter les informations apparaissant dans le PDF produit par sa compilation.

$\LaTeX$  nécessite un minimum d'effort pour être pris en main, mais il n'y a rien d'insurmontable, notamment pour des étudiants en informatique (d'autant plus s'ils sont déjà familiers de langages comme HTML). Le contrôle obtenu et le rendu du document sont largement supérieurs à celui des logiciels de traitement de texte, y compris les logiciels commerciaux comme MS Word.

À noter qu'un très grand nombre d'extensions existent pour traiter des besoins spécifiques : diagrammes particuliers, documents de présentation de type Powerpoint, etc. Il est aussi possible d'intégrer du code Python ou R (ou autre langage) dans un document  $\LaTeX$ , afin par exemple de mettre à jour des tables à chaque compilation, et ainsi tenir compte de l'évolution de données externes. À vous d'explorer toutes ces possibilités !

# Bibliographie

- [1] Jean-Valère COSSU, Vincent LABATUT et Nicolas DUGUÉ. « A review of features for the discrimination of Twitter users : application to the prediction of offline influence ». In : *Social Network Analysis and Mining* 6 (2016), p. 1–23. DOI : [10.1007/s13278-016-0329-x](https://doi.org/10.1007/s13278-016-0329-x) (cf. p. 11).
- [2] Santo FORTUNATO. « Community detection in graphs ». In : *Physics Reports* 486.3-5 (2010), p. 75–174. DOI : [10.1016/j.physrep.2009.11.002](https://doi.org/10.1016/j.physrep.2009.11.002) (cf. p. 11).
- [3] Antoine GERBAUD. « Modélisation de réseaux d'interaction par des graphes aléatoires ». PhD Thesis. Grenoble, FR : Université de Grenoble, 2010. URL : <http://www.theses.fr/s123781> (cf. p. 11).
- [4] JABREF. *JabRef Documentation*. JabRef. 2008. URL : <http://jabref.sourceforge.net/documentation.php> (cf. p. 4, 11).
- [5] V. LABATUT et J.-M. BALASQUE. « Detection and Interpretation of Communities in Complex Networks : Methods and Practical Application ». In : *Computational Social Networks : Tools, Perspectives and Applications*. Sous la dir. d'A. ABRAHAM et A.-E. HASSANIEN. Springer, 2012. Chap. 4, p. 81–113. DOI : [10.1007/978-1-4471-4048-1\\_4](https://doi.org/10.1007/978-1-4471-4048-1_4) (cf. p. 11).
- [6] LATEX PROJECT. *LaTeX – A document preparation system*. LaTeX Project. 2010. URL : <http://www.latex-project.org/> (cf. p. 3).
- [7] N. MASUDA et R. LAMBIOTTE. *A Guide to Temporal Networks*. World Scientific, 2016. DOI : [10.1142/9781786341150](https://doi.org/10.1142/9781786341150) (cf. p. 11, 12).
- [8] Antônio MAUTTONÉ, Martine LABBÉ et Rosa FIGUEIREDO. « A Tabu Search approach to solve a network design problem with user-optimal flows ». In : *VALIO/EURO Conference on Combinatorial Optimization*. Buenos Aires, 2008, p. 1–6 (cf. p. 11).
- [9] MIKTEX. *MiKTeX... typesetting beautiful documents...* MiKTeX. 2015. URL : <http://miktex.org/> (cf. p. 3).
- [10] ORACLE. *VirtualBox Documentation*. Oracle. 2011. URL : <http://www.virtualbox.org/wiki/Documentation> (cf. p. 3).
- [11] Dimitris C. PARASKEVOPOULOS, Tolga BEKTAS, Teodor Gabriel CRAINIC et Chris N. POTTS. *A Cycle-Based Evolutionary Algorithm for the Fixed-Charge Capacitated Multi-Commodity Network Design Problem*. Rapp. tech. Montréal : Centre interuniversitaire de recherche sur les réseaux d'entreprise, la logistique et le transport, 2013, p. 1–31 (cf. p. 11).
- [12] M. ROSVALL et C. T. BERGSTROM. *Fast stochastic and recursive search algorithm*. Rapp. tech. Umeå, SE : Department of Physics, Umeå University, 2009 (cf. p. 11).
- [13] TEXMAKER. *Manuel de l'utilisateur*. Texmaker. 2011. URL : [http://www.xmlmath.net/texmaker/doc\\_fr.html](http://www.xmlmath.net/texmaker/doc_fr.html) (cf. p. 4).
- [14] Y.-C. WEI et C.-K. CHENG. « Towards efficient hierarchical designs by ratio cut partitioning ». In : *IEEE International Conference on Computer Aided Design*. 1989, p. 298–301. DOI : [10.1109/ICCAD.1989.76957](https://doi.org/10.1109/ICCAD.1989.76957) (cf. p. 11).
- [15] WIKIBOOKS. *Guide pour la programmation en LaTeX*. Wikibooks. 2010. URL : <http://fr.wikibooks.org/wiki/LaTeX> (cf. p. 3, 6, 9, 11, 12).
- [16] WIKIBOOKS. *Guide to the LaTeX markup language*. Wikibooks. 2011. URL : <http://en.wikibooks.org/wiki/LaTeX> (cf. p. 3, 6, 9, 12).
- [17] WIKIPEDIA. *Flowchart*. Wikipedia. 2011. URL : <http://en.wikipedia.org/wiki/Flowchart> (cf. p. 10).
- [18] WIKIPEDIA. *LaTeX*. Wikipedia. 2011. URL : <http://fr.wikipedia.org/wiki/LaTeX> (cf. p. 3).

- [19] Richard T. WONG. « Accelerating Benders decomposition for network design ». PhD Thesis. Massachusetts Institute of Technology, 1978 (cf. p. 11).
- [20] Richard T. WONG. « Worst-Case Analysis of Network Design Problem Heuristics ». In : *SIAM Journal on Algebraic Discrete Methods* 1.1 (1980), p. 51–63. DOI : [10.1137/0601008](https://doi.org/10.1137/0601008) (cf. p. 11).