

Initiation Latex

Simon Halfon, ENS Cachan

October 16, 2015

1 Installation

Selon votre plateforme:

- Linux: `$ apt-get texlive`
- MacOS:
 1. installer MacPort ou HomeBrew
 2. `$ sudo port install tex-live`
- Windows: allez voir sur OpenClassroom, par exemple.

2 Editeur et compilation

Latex est un langage de programmation, vous pouvez donc écrire le code Latex dans n'importe quel éditeur. Bien sûr, certains sont plus adaptés.

EDITEUR

- Emacs + AucTex
- Votre éditeur préféré + terminal
- Kile (Linux)
- TexShop (MacOS)
- TexNicCenter (Windows)

COMPILATION

- `C-c C-c`
- `$ pdflatex monfichier.tex`
- Interface newbie
- Interface newbie
- Interface newbie

3 Structure d'un document

3.1 Version primitive

La structure d'un document est la suivante:

- Votre document commence par

```
\documentclass{article}
```

- Le header: les packages, les environnements, les définitions et les commandes que vous définissez
- Enfin, le corps du document: commence par

```
\begin{document}
```

et se termine par

```
\end{document}
```

Je vous invite à consulter le code source du document que vous êtes en train de lire. Vous le trouverez sur ma page (www.lsv.ens-cachan.fr/~halfon/). Ne vous alarmez pas pour ma façon d'écrire les accents, vous n'avez pas à en faire autant.

3.2 La commande input

Si votre document est long, par exemple un rapport de stage d'une vingtaine de pages, il est préférable de le séparer en plusieurs fichiers, comme vous le faites (comme vous devriez le faire) pour du code classique.

Le code source du présent document ressemblerait alors au code que vous pouvez voir dans *exemple.tex*, disponible sur ma page également. Je n'ai pas écrit les fichiers *installation.tex*, *editeur.tex*, *structure.tex*, etc., mais ils contiendraient ni plus ni moins que le contenu exact de chaque section. Lorsque Latex voit un *input*, c'est comme s'il collait le code du fichier à l'emplacement du *input*.

C'est particulièrement utile pour co-écrire des articles: les différents auteurs peuvent modifier différentes parties de l'article sans avoir de difficultés de fusion lors de la synchronisation.

4 Un peu de syntaxe

Il y a énormément de littérature sur Latex, à commencer par le Wiki. Néanmoins, si vous cherchez comment dessiner un symbole étrange, que vous ne sauriez nommer pour faire une recherche Web pertinente, vous pouvez utiliser Detexify. Par exemple, si vous décidez de faire votre DM de calculabilité sur l'ordinateur, comment trouveriez-vous le symbole \perp ?

Autre élément de syntaxe incontournable: les commentaires, annoncés par le symbole % (ligne courante seulement).

Pour le reste, vous allez vous entraîner avec l'archive *Squelette* que vous trouvez également sur ma page.

5 La bibliographie

De loin la fonctionnalité la plus simple à utiliser dans Latex (c'est ironique). La page de référence bibliographique est générée automatiquement par Latex, lorsqu'il lit les deux lignes suivantes en toute fin de code:

```
\bibliographystyle{plain}
\bibliography{biblio}
```

La première donne le style de la bibliographie. Il y en a plein, allez jeter un oeil sur Internet. Le style définit l'affichage des références: *Auteur - Titre - Année*, ou dans tel ou tel sens, avec ou sans année etc. Le style modifie aussi l'encart devant la référence: soit chaque référence se voit attribuer un numéro (par exemple [1]), et lorsque vous citez cet article dans le texte, à l'aide de la commande

```
\cite{tag}
```

il s'affiche alors [1]. Vous pouvez aussi obtenir des références de la forme [AKS02] si l'article cité est écrit par Agrawal, Kayal et Saxena en 2002; ou encore des encart custom.

La seconde ligne donne le nom du fichier dans lequel se trouve vos références bibliographiques, ici *biblio.bib* (vous savez où le trouver). Vous pouvez aller jeter un oeil à la syntaxe d'un .bib, mais ça a peu d'importance: si vous voulez citer l'article *PRIMES is in P*, tapez *PRIMES is in P bibtex* dans un moteur de recherche, et vous obtiendrez ce que vous voulez:

```
@ARTICLE{AKS02,
  author = {Manindra Agrawal and Neeraj Kayal and Nitin Saxena},
  title = {PRIMES is in P},
  journal = {Ann. of Math},
  year = {2002},
  volume = {2},
  pages = {781--793}
}
```

Retenez simplement que le premier mot (ici j'ai mis AKS02) est le tag, pour citer l'article il faudra écrire:

```
\cite{AKS02}
```

C'est une convention de nommage de tag relativement pratique, je vous la recommande (vous verrez qu'elle n'est pas respectée dans mon .bib, bien sûr). Autre convention: il est très moche que la référence soit en début de ligne, on utilise donc l'espace insécable:

```
[...] comme on le voit dans~\cite{AKS02}.
```

Et là, c'est le plus drôle, pour compiler le tout:

- `$ pdflatex monfichier.tex` une première fois;
- `$ bibtex monfichier` Attention, pas d'extension, et il s'agit bien du nom du fichier .tex, pas du .bib;
- `$ pdflatex monfichier.tex` une seconde fois;
- `$ pdflatex monfichier.tex` une troisième fois ! J'avais prévenu que c'était simple.

Ceci ne vaut que lorsque vous modifiez le .bib. Le reste du temps, compilez normalement, éventuellement deux fois s'il y a eu des changements de labels (*cf* section suivante).

Dernière chose: vous remarquerez que j'ai en fait utilisé le même fichier *biblio.bib* pour *latex.tex* et *beamer.tex*, et que ces deux documents ne citent pas les mêmes articles. Par défaut, Latex ne met sur la page de référence que les articles que vous citez explicitement dans votre texte, votre fichier .bib peut donc contenir tous les articles du monde, seuls ceux que vous citez seront affichés. Si vous voulez tout de même qu'un article que vous ne citez pas apparaisse en référence, il faut ajouter la commande

```
\nocite{Nadine}
```

en fin de fichier. Observez que *Nadine* apparaît bien dans mes références, alors que je ne la cite jamais.

6 Les labels et les références

Definition 1. *Un label est un machin qui permet de faire référence à un truc.*

Vous remarquerez que la définition 1 est un peu opaque. Elle contraste parfaitement avec la clarté de la section 5. Dans le code source, vous verrez que j'utilise des conventions pour nommer mes labels, *sec:section* pour une section, *def:definition* pour une définition, etc. Il s'agit simplement d'une convention, Latex s'en contre fiche. Moi non, respectez cette convention.

Autre chose: n'écrivez **JAMAIS** "comme vous le voyez à la section 2" dans votre code source ! Le but de Latex est de vous simplifier la vie: votre code fonctionne toujours si vous ajoutez une section. Pour les labels comme pour les citations, il faut deux compilations à Latex pour tout comprendre.

7 Les Beamer

Pour cette section, téléchargez l'archive *beamer.zip*.

Pour faire un beamer, il faut commencer par donner la bonne classe à votre document:

```
\documentclass{beamer}
```

Vous pouvez ensuite spécifier un thème pour votre beamer: le thème déterminera la position et la forme de votre menu, les choses qui s'affichent ou non sur vos slides, etc. Vous trouverez une nomenclature de thème sur Internet.

La suite du document est la même que d'habitude: les packages, les commandes que vous définissez, le titre du beamer, puis:

```
\begin{document}
```

Quelques commandes spécifiques aux beamers:

- Une slide est délimitée par

```
\begin{frame}  
\end{frame}
```

- Vous pouvez demander la génération automatique d'une slide de sommaire, ainsi que le rappel du sommaire à chaque début de section (voir le code commenté après la slide de titre dans *beamer.tex*).
- La commande

```
\pause
```

vous permet d'afficher votre slide en plusieurs fois. C'est très pratique si vous avez un slide chargé et que vous voulez forcer votre auditoire à ne lire que la partie dont vous parlez actuellement. En effet, sachez que lorsque vous dévoilez un slide, votre public le lira, temps pendant lequel il ne vous écoute plus (pour vous en convaincre, "observez vous" à la prochaine conférence à laquelle vous assistez, c'est effectivement ce que vous faites). C'est la raison pour laquelle on vous recommande de ne pas mettre de texte du tout dans une slide. S'il y a du texte, alors dévoilez le au fur et à mesure, pour que l'auditoire ne lise que la partie dont vous parlez.

Remarque: notez le numéro de la slide 3 et de la suivante. C'est un petit détail qui devrait vous encourager à utiliser `\pause` et pas à copier votre slide deux fois. L'autre raison de ne pas faire ça est une raison d'alignement: Latex gère la mise en page tout seul, et notamment centre votre texte. Si vous utilisez la commande `pause`, Latex tient compte du texte à venir et lui laisse la place.

- Les images: ce n'est absolument pas spécifique aux beamers, et la syntaxe pour les mettre dans les autres types de documents Latex est exactement la même. Néanmoins, comme je n'en ai pas mis avant, j'en profite pour vous indiquer d'observer cette syntaxe.

Il faut distinguer les images flottantes des figures. Cela a moins d'importance dans un beamer, puisque une slide est spatialement petite, mais dans un article, Latex placera la figure où bon lui semble, pas forcément là où vous l'avez indiqué dans le code. L'idée est qu'une figure a une légende

(`\caption`) et idéalement un label, auquel vous faites référence dans le texte. Donc votre figure n'a pas besoin d'être présente directement sous le texte qui lui fait référence.

Si vous voulez forcer la position d'une image, alors ne mettez pas l'environnement `figure`. En contre partie, vous perdez la possibilité de mettre une légende. C'est tout à fait naturel: plus besoin de mettre un légende puisque vous savez précisément où la figure atterrira, vous pouvez donc la décrire dans le texte.

- Le package `Tikz`: n'est pas non plus spécifique au beamer. C'est un outil très puissant, et son utilisation demande un peu d'apprentissage. Il y a énormément de documentation en ligne (officielle pour commencer, et de réponses aux questions sur divers forums).
- Les block

8 Conclusion

Latex permet de faire beaucoup de chose, mais demande une certaines prise en main. Je vous conseille de miser sur l'expérience: vous apprendrez les choses au fur et à mesure, dès lors que vous en aurez besoin. Je ne pense pas qu'il soit utile de se bourrer le crane avec tout un gros tas de syntaxe que vous aurez vite fait d'oublier si vous ne l'utilisez pas rapidement et/ou régulièrement.

La communauté Latex est assez développée, vos questions "comment faire ce truc qui serait bien pratique de faire ?" ou "que signifie cette erreur obscure ?" ont sans doute été déjà posée et résolue sur des forums (le plus sérieux étant sans doute StackOverflow).

References

- [1] Manindra Agrawal, Neeraj Kayal, and Nitin Saxena. Primes is in p. *Ann. of Math*, 2:781–793, 2002.
- [2] Nadine. Je suis dans la cuisine. *Grenadine & Tajine*, 12(8), 2042.