

Ateliers d'initiation à L^AT_EX- Cinquième séance

Martin Van den Abbeele

Laurent Bataille

Novembre 2019

1 Introduction

Ces dernières années, L^AT_EX a beaucoup évolué au niveau de son contenu. De nombreux outils ont été explorés et il est désormais possible de dessiner via des lignes de code. Ceci est rendu possible par l'extension TikZ. Si cela peut sembler alambiqué de dessiner directement en lignes de code, le formalisme de cette extension permet d'étendre la philosophie du L^AT_EX à la réalisation de schémas : on définit en lignes de code les options de style d'un digramme et on assemble le tout. Un manuel très bien écrit est disponible à cette adresse <http://math.et.info.free.fr/TikZ/bdd/TikZ-Impatient.pdf>.

Cette approche n'est évidemment pas forcément la plus ergonomique dans tous les cas, il est parfois nettement plus simple de réaliser un schéma dans un interface plus conventionnel, par exemple avec le logiciel Inkscape présenté précédemment ou avec Dia ¹.

Objectifs

- Vous introduire de façon succincte les bases du formalisme de TikZ
- Décrire différents outils basés sur cette extension pour des applications spécifiques
- Expliquer le fonctionnement de chemfig et introduire des outils automatisés compatibles avec cette extension

2 Les Flow Charts avec TikZ

Il existe d'autres extensions T_EX destinées à la représentation schématique, cependant, celles-ci ne sont pas forcément ergonomiques et requièrent un niveau d'utilisation plus avancé sans forcément offrir un rendu plus intéressant.

TikZ nécessite en règle générale de dessiner une ébauche sur papier pour commencer à dimensionner les composantes. Cependant dans le cas des flow charts, le contenu peut-être positionné automatiquement dans un premier temps.

1. <http://dia-installer.de/>

Options et commandes de base pour l'utilisation de TikZ

Pour utiliser TikZ, il vous sera nécessaire d'ajouter les extensions adéquates au document :

```
\usepackage{tikz}
\usepackage{pgfplots}
\usetikzlibrary{shapes,arrows,positioning,babel,calc}
```

Les dessins sont inclus en utilisant l'environnement `tikzpicture`.

```
\begin{tikzpicture}[options]
...
\end{tikzpicture}
```

La première étape est de décrire le style des composantes de votre flow-chart. Par exemple un bouton décision, représenté par un losange comme votre cours de gestion de la qualité

```
\tikzstyle{decision} = [diamond, draw, fill=blue!20,
    text width=4.5em, text badly centered, node distance=3cm, inner
    sep=0pt]
```

Une fois les options de style globales définies, on peut créer les noeuds en utilisant la commande `\node` qui permet de donner un titre à l'élément créé ainsi que de nommer ce noeud. En utilisant le nom de différents il est possible par exemple de symboliser l'interaction de différents noeuds via la commande `\path` et de définir le type de connection souhaité (– dans l'exemple)

```
\begin{tikzpicture}[node distance = 2cm, auto]
    \node [options de style et de positionnement] (nom_de_l'element)
        {Titre apparaissant à l'écran};
    \path [line] (noeuds1) -- (noeud2);
\end{tikzpicture}
```

exemple complet

```
\tikzstyle{decision} = [diamond, draw, fill=blue!20,
    text width=4.5em, text badly centered, node distance=3cm, inner
    sep=0pt]
\tikzstyle{block} = [rectangle, draw, fill=blue!20,
    text width=5em, text centered, rounded corners, minimum height
    =4em]
\tikzstyle{line} = [draw, -latex']
\tikzstyle{cloud} = [draw, ellipse,fill=red!20, node distance=3cm,
    minimum height=2em]

\begin{tikzpicture}[node distance = 2cm, auto]
    % Place nodes
    \node [block] (init) {initialize model};
    \node [cloud, left of=init] (expert) {expert};
    \node [cloud, right of=init] (system) {system};
    \node [block, below of=init] (identify) {identify candidate
        models};
    \node [block, below of=identify] (evaluate) {evaluate candidate
        models};
    \node [block, left of=evaluate, node distance=3cm] (update) {
        update model};
    \node [decision, below of=evaluate] (decide) {is best candidate
        better?};
    \node [block, below of=decide, node distance=3cm] (stop) {stop
        };
    % Draw edges
    \path [line] (init) -- (identify);
    \path [line] (identify) -- (evaluate);
    \path [line] (evaluate) -- (decide);
    \path [line] (decide) -| node [near start] {yes} (update);
    \path [line] (update) |- (identify);
    \path [line] (decide) -- node {no}(stop);
    \path [line,dashed] (expert) -- (init);
    \path [line,dashed] (system) -- (init);
    \path [line,dashed] (system) |- (evaluate);
\end{tikzpicture}
```

Défi du jour : reproduire la Figure 1 en utilisant TikZ².

2. La couleur immonde du fond des boîtes est similaire à celle de l'image originale

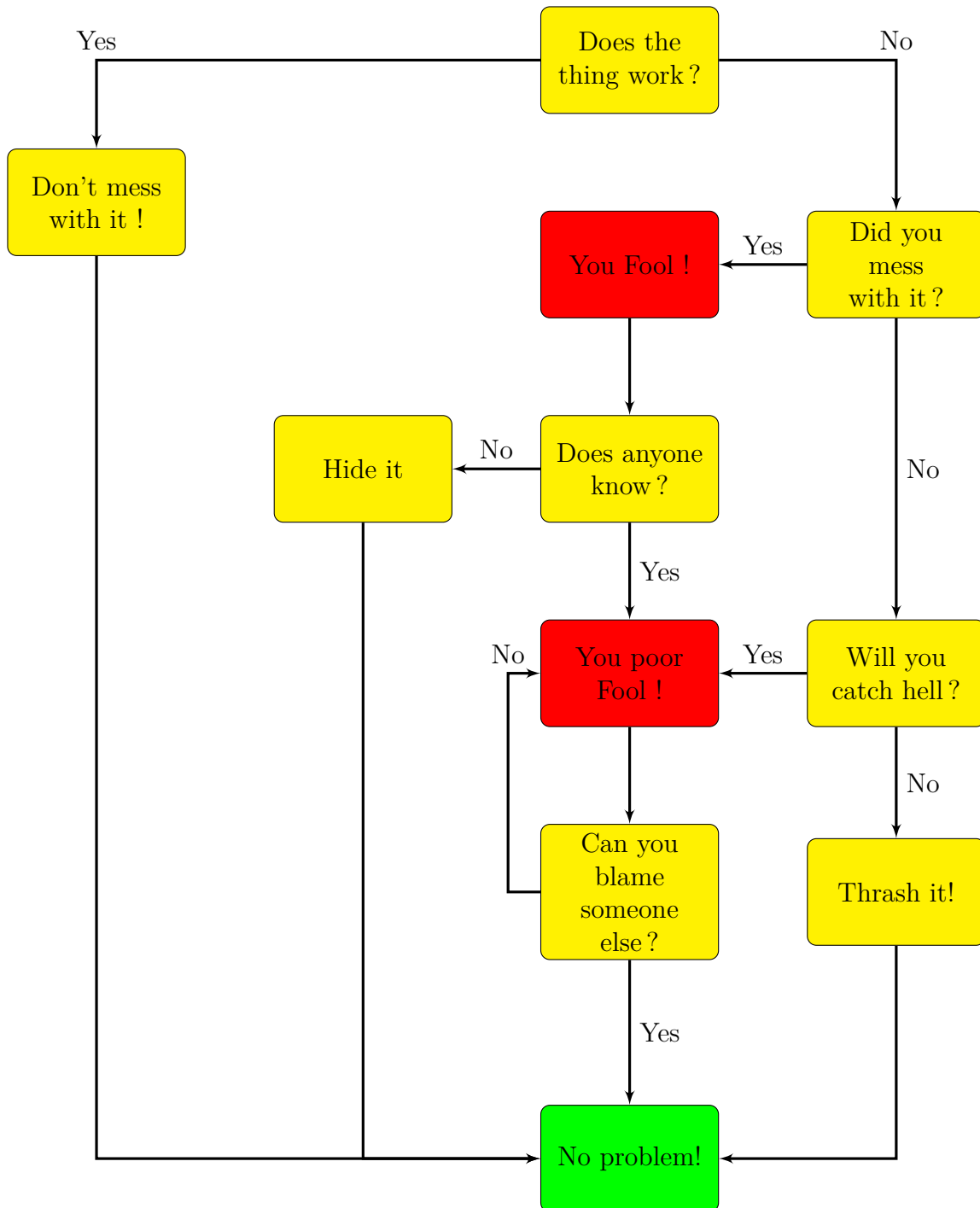


FIGURE 1 – La stratégie de l'évitement en un diagramme

Sachez également qu'il existe des outils d'export vers TikZ partant de Inkscape, transcrivant un diagramme en TikZ, via l'extension <https://github.com/kjellmf/svg2tikz>. L'outil de conversion ne fonctionnant pas toujours parfaitement pour des schémas complexes, il est plutôt recommandé d'exporter ses schémas en pdf_{tex} avant de les importer sous L^AT_EX (voir annexes du TP sur les images). Il existe par ailleurs un équivalent de paint version Tikz. Pour le faire fonctionner, il est nécessaire d'employer l'extension freetikz³ et d'utiliser l'éditeur suivant : <http://homepages.inf.ed.ac.uk/cheunen/freetikz/freetikz.sty>. Ceci vous permet de transcrire des croquis directement en TikZ pour éventuellement les retravailler plus tard.

3 Les circuits électriques avec TikZ

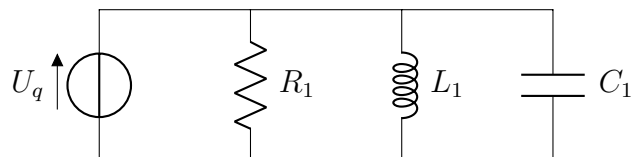
Maintenant que vous êtes des pros en matière de diagrammes, il est temps de vous initier à la représentation de circuits. Concrètement, il est nécessaire de réaliser un schéma papier avant de se lancer dans le code ; Toute la symbolologie usuelle est définie par l'extension, le manuel donne les commandes pour l'ensemble des composants utilisables avec l'extension :

<http://mirrors.ibiblio.org/CTAN/graphics/pgf/contrib/circuitikz/doc/circuitikzmanual.pdf>

Il est nécessaire d'inclure dans le préambule la ligne `\usepackage{circuitikz}`. Un exemple basique avec des composants simples pour illustrer la syntaxe utilisée :

```
\begin{circuitikz}
  \draw (0,0)
    to[V,v=$U_q$] (0,2) % The voltage source
    to[short] (2,2)
    to[R=$R_1$] (2,0) % The resistor
    to[short] (0,0);
  \draw (2,2)
    to[short] (4,2)
    to[L=$L_1$] (4,0)
    to[short] (2,0);
  \draw (4,2)
    to[short] (6,2)
    to[C=$C_1$] (6,0)
    to[short] (4,0);
\end{circuitikz}
```

Ce qui se transcrit comme :



Comme exercice nous vous proposons de représenter cette fine et pas du tout caricaturale représentation de la situation financière d'un(e) étudiant(e) gembloutoise, disponible à la Figure 2...

3. <http://homepages.inf.ed.ac.uk/cheunen/freetikz/freetikz.sty>

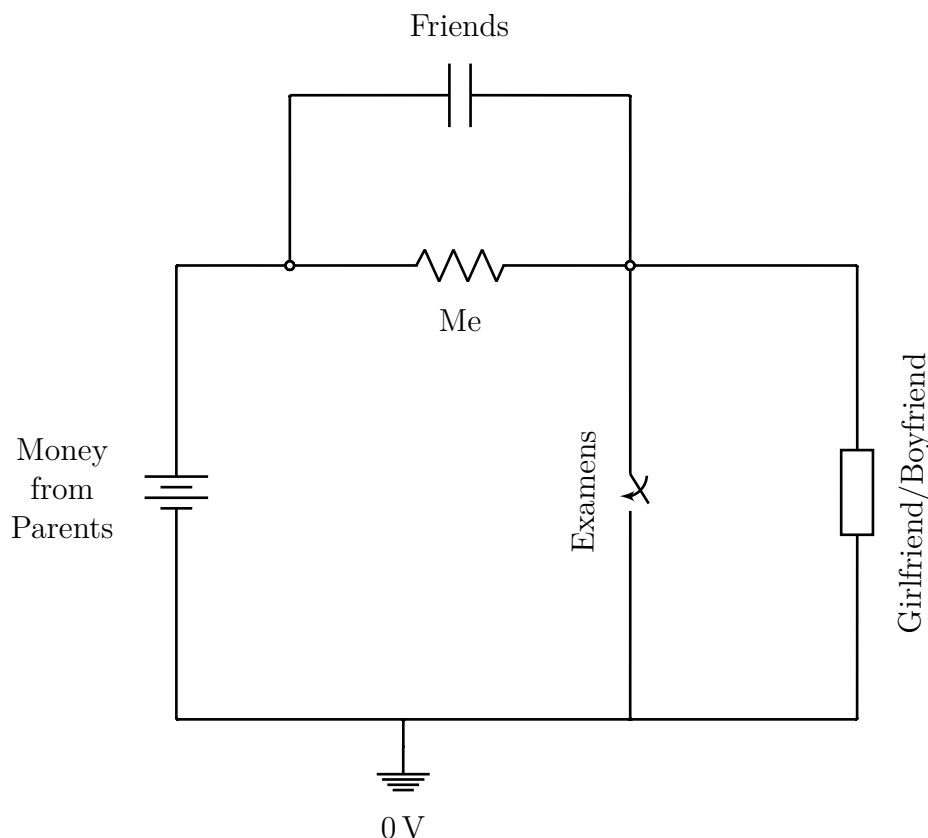


FIGURE 2 – Toute ressemblance avec des faits réels est fortuite et est le fruit d'un malheureux hasard

3.1 Les circuits électriques sous Inkscape

Pour des circuits complexes, il peut être nécessaire d'utiliser un interface graphique pour manipuler directement les composants. Il est possible de sélectionner la symbologie nécessaire sous forme de fichiers svg sur Wikipedia dans le bon répertoire : https://commons.wikimedia.org/wiki/Category:SVG_electrical_symbols Le tout s'assemble en utilisant l'outil d'importation d'inkscape, l'outil "magnétisme" peut faciliter les choses pour aligner facilement les composants. L'étape finale consiste en l'uniformisation de l'épaisseur des traits en modifiant les propriétés des objets.

Il existe désormais une extension Inkscape permettant d'obtenir des résultats de très haute qualité, elle-même basée sur circuittikz, elle est disponible sur <https://github.com/fsmMLK/inkscapeCircuitSymbols> .

4 Les molécules chimiques avec chemfig

La communauté L^AT_EX utilise principalement deux extensions dédiées à la représentation de molécules chimiques, **chemfig** et **XyMTeX**. Ce tutoriel s'attardera uniquement sur la première, **XyMTeX** demandant une prise en main plus longue, mais offrant une plus grande liberté d'utilisation, par ailleurs si cette extension vous intéresse, son créateur a publié un manuel extrêmement complet⁴.

La documentation de chemfig est disponible à l'adresse suivante :
<http://mirrors.ibiblio.org/CTAN/macros/generic/chemfig/chemfig-fr.pdf>

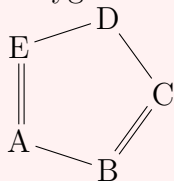
Chemfig - Commandes de base

Le concept le plus basique à comprendre est de comment définir l'inclinaison d'une connexion entre atomes. On distingue trois formalismes différents tous adaptés à une situation spécifique :

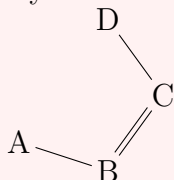
- Unités par défaut (pas de 45°) : `\chemfig{A-[1]B-[7]C}`
- Angles absolus : `\chemfig{A-[:50]B-[:-25]C}`
- Angles relatifs : `\chemfig{A-[:50]B-[:-25]C}`

L'étape suivante est de définir des polygones réguliers, quelques exemples pratiques :

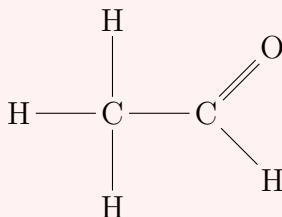
- Polygones réguliers - `\chemfig{A*5(-B=C-D-E=)}`



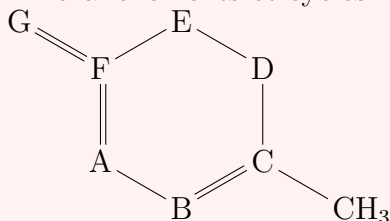
- Cycles incomplets : `\chemfig{A*5(-B=C-D)}`



- Embranchements : `\chemfig{H-C(-[2]H)(-[6]H)-C(=[1]O)-[7]H}`



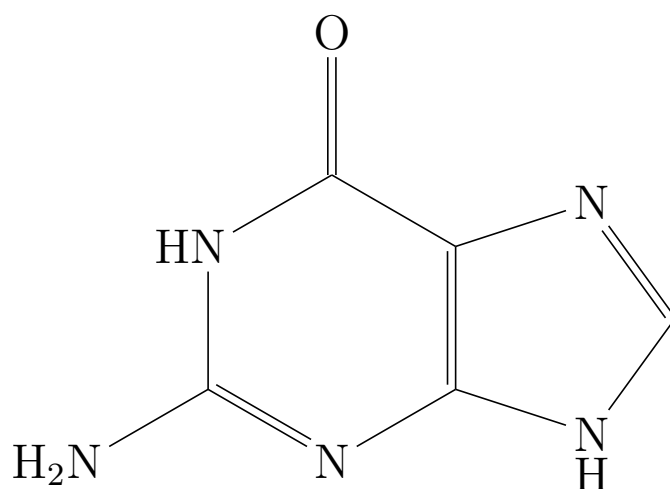
- Embranchements et cycles : `\chemfig{A*6(-B=C(-CH_3)-D-E-F(=G)=)}`



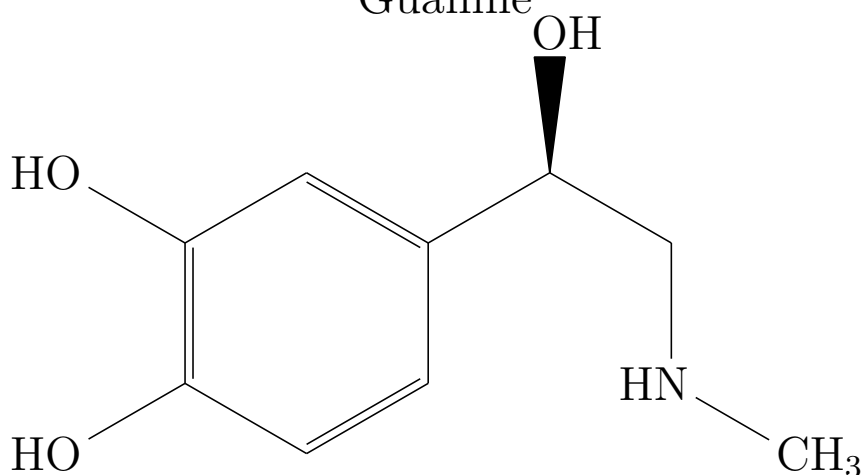
On remarquera que le sens conventionnel est dirigé dans le sens trigonométrique !

4. <http://ctan.cs.uu.nl/macros/latex/contrib/xyntex/doc/XyMTeX-manual.pdf>

Exercice - Dessiner la guanine ou l'adrénaline



Guanine



Adrénaline

Pour des représentations plus complexes mais avec des patterns répétitifs, il est possible de définir des morceaux de molécules et leur connectique via la commande `\submmol`

Exemple : dessiner une molécule de cellulose, tiré de <https://tex.stackovernet.com/fr/q/32222>

```
% this submol starts at the left-most bond (to C4)
% and ends at the right-most bond (from C1);
% it does not use the ?-syntax to avoid unwanted
% connections between different instances of the
% glucose rings:
\definesubmol{glucoBeta}{
  -[:10,.7]4
  (
    -[: -10](-[:150,0.7]-[2,0.7]OH)
    -[:10]{\color{red}{0}}-[: -50,.75]
  )
  -[: -50](-[:170]HO)
  -[:10](-[: -55,0.7]OH)
  -[: -10]1(-[6,0.7]H)
  -[:10,.7]0}
```



```
% this submol is the same as the one before except
% that every bond has been changed to the opposite direction:
\definesubmol{turned-glucoBeta}{
  -[: -10, .7]4
  (
    -[:10](-[: -150, 0.7] - [6, 0.7] OH)
    -[: -10]{\color{red}{0}} - [:50, .75]
  )
  -[:50](-[: -170] H0)
  -[: -10](-[:55, 0.7] OH)
  -[:10]1(-[2, 0.7] H)
  -[: -10, .7]0
}
```

```
\chemfig{H0!\glucoBeta!\turned-glucoBeta!\glucoBeta!\turned-glucoBeta
}H}
```

Un autre exemple est :

```
\definesubmol\Me[H_3C]{CH_3}
\chemfig{*6((-!\Me)=(-!\Me)-(-!\Me)=(-!\Me)-(-!\Me)-)}
```

Comment éviter de coder des molécules complexes à la main ?

- Dessiner une molécule ou télécharger un modèle existant sur https://py-chemist.com/mol_2_chemfig/home et utiliser l'extension `mol2chemfig` téléchargeable sur <https://ctan.org/pkg/mol2chemfig>. Le package n'est pas disponible nativement sur Overleaf, utilisez le fichier `.sty` inclu dans le zip et l'inclure dans votre document avec `\usepackage{mol2chemfig.sty}`.
- Se servir de modèles d'hydrates de carbone disponible via l'extension `carbohydrates`, téléchargeable sur <https://github.com/cgnieder/carbohydrates>.

A Dessiner des schémas de la théorie des poutres

L'extension `stanli`, permet de créer des schémas de la théorie des poutres et d'analyse de structure avec TikZ en 2D <https://ctan.org/pkg/stanli>. Cette extension fonctionne de façon similaire à TikZ, on commence par définir les points via la commande :

```
\point{Nom_de_Point}{x}{y}
```

On définit ensuite la connection entre les points

```
%TypePoutre, TypeAppuiDébut et TypeAppuiFin sont des nombres entiers,
% se référer à la doc
\beam{TypePoutre}{Nom_du_début}{Nom_de_la_fin}[TypeAppuiDébut][
TypeAppuiFin];
```

De façon similaire on peut définir les charges s'appliquant sur le modèle :

```
% Charge ponctuelle
\load {TypeCharge}{Nom_du_pointdapplication}[orientation][longueur];
% Charge linéique
\lineload {TypeCharge}{Nom_du_début}{Nom_de_la_fin}[options];
```

B Arbres dichotomiques

TikZ propose un outil natif pour représenter les arbres. Toutefois, un autre outil plus simple d'utilisation émerge, via le package **forest**⁵. Il est nécessaire de représenter le style et les paramètres des arbres dans les paramètres de style de l'environnement. De nombreux exemples sont repris dans la documentation officielle :

<http://mirrors.ibiblio.org/CTAN/graphics/pgf/contrib/forest/forest-doc.pdf> .

La mécanique de code de l'arbre est assez simple :

```
[nom du noeud[nom du noeud enfant [nom du noeud petit-fils]]]
```

L'exemple suivant est tiré d'une discussion comparant différents types d'outils de représentation des arbres dicotomique <https://tex.stackexchange.com/a/60578>

```
\documentclass{standalone}
\usepackage{forest}
\begin{document}
\begin{forest}
[country
  [name [France]] [city[ Paris ]] [region
    [name [Nord-Pas de Calais] ]
    [population [3996] ]
    [city [Lille] ]
  ]
  [region
    [name [Valle du Rhone] ]
    [city [Lyon] ]
    [city [Valence] ]
  ]
]
\end{forest}
\end{document}
```

Exercice : implémentez l'arbre de votre famille préférée de l'univers de Game of Thrones !

5. forest appelle le package tikz en interne

C Les posters sous L^AT_EX

Il existe de nombreux environnements/packages basés sur TikZ permettant de créer des posters. Dans le cadre de ce tutoriel, nous nous attachons à présenter des exemples simples, c'est pourquoi nous avons choisi tikzposter. Un exemple de poster présentant cet outil est disponible ici ! Le code minimaliste repris dans ce document permet d'étudier rapidement la syntaxe des commandes et environnements introduits par ce package :

```
\documentclass[25pt, a0paper, portrait, margin=0mm, innermargin=15mm,
  blockverticalspace=15mm, colspace=15mm, subcolspace=8mm]{tikzposter}
\title{Using tikzposter}
\author{Pascal Richter, Elena Botoeva, Richard Barnard, \& Dirk Surmann}
\institute{}\usetheme{Autumn}\usecolorstyle[colorPalette=
  BrownBlueOrange]{Germany}

\begin{document}
%Version épurée moche mais mettant en avant le fonctionnement du code
\maketitle
\begin{columns}
  \column{0.55}
    \block{Creating the document}{The document...}
    \note[targetoffsetx=-.05\textwidth,targetoffsety=9.5cm,innersep
      =.4cm,angle=-45,connection]{Optional arguments...}
    \block{The title matter}{The title...}\block{Blocks}{Blocks are
      ...}
    \note[targetoffsetx=24cm, targetoffsety=-9cm,rotate=1,angle=270,
      radius=8cm,width=.75\textwidth,innersep=.4cm]{Youcan...}
  \column{0.45}
    \block{Columns}{By default,...}
    \begin{subcolumns}
      \subcolumn{.45}
        \block{Subcolumns}{If you...}
      \subcolumn{.5}
        \block{}{An example...}
    \end{subcolumns}
    \block[titlewidthscale=.8,bodywidthscale=.9,titleoffsety=9.5mm,
      bodyoffsety=9mm]{Changing the Poster's Appearance}{If the
      default...}
\end{columns}
\block[titleoffsety=-1cm,bodyoffsety=-1cm]{Sample document}{This poster
  ...}
\end{document}
```

Ces document reprennent la majorité des thèmes/options de personnalisation utilisables pour votre poster :

- <https://bitbucket.org/surmann/tikzposter/src/master/separatefiles/styleguide.pdf>
- <https://bitbucket.org/surmann/tikzposter/downloads/themes.pdf>