

Chronique 1

Verbatim et plus

Ayant fait un peu de Python avec la distribution `Pyzo`, j'ai souhaité intégrer le listing des fonctions créées dans des documents \LaTeX .

Puis décoller ce listing du bord gauche de la marge, l'encadrer et en numéroté les lignes.

Voici mon cheminement en prenant pour exemple une fonction `maxi()` très simple qui donne le maximum d'une liste.

1.1 Verbatim

Définition de « verbatim » d'après [Le Robert](#) :

« Compte rendu écrit fournissant le mot à mot d'une déclaration, d'un débat oral. »

L'environnement `verbatim` permet de donner le programme de la fonction que l'on colle depuis l'éditeur Python.

	<pre>\begin{verbatim} def maxi(L): max=L[0] for k in L: if k>max: max=k return max \end{verbatim}</pre>				
En tapant		on obtient		<pre>def maxi(L): max=L[0] for k in L: if k>max: max=k return max</pre>	

À noter la version étoilée de `verbatim` qui matérialise les espaces en les remplaçant par des `_`.

	<pre>\begin{verbatim*} def maxi(L): max=L[0] for k in L: if k>max: max=k return max \end{verbatim*}</pre>				
En tapant		on obtient		<pre>def_maxi(L): max=L[0] for_k_in_L: if_k>max: max=k return_max</pre>	

L'étoile fonctionne aussi pour la version en ligne de `verbatim`, c'est-à-dire `\verb`.

<code>\verb*!oui ou non!</code>	donne	<code>oui_ou_non</code>
---------------------------------	-------	-------------------------

1.2 Environnement changemargin

L'environnement `verbatim` a le désavantage de désactiver les environnements de positionnements horizontaux (`flushleft`, `flushright` ou `center`).

On ne peut même pas insérer devant le `\begin{verbatim}` un `\hspace{}` pour décoller le listing de la marge gauche. Ni même un `\quad` ou un `\qquad`.

Encore un échec en tentant de modifier temporairement la marge gauche avec quelque chose comme `\setlength{\parindent}{}`.

Heureusement, on trouve beaucoup de documentation en ligne. Dans les Cahiers GUTenberg n° 23 de 1996, j'ai trouvé l'environnement `changemargin` qui permet de redéfinir la marge gauche et la marge droite de ce qui se trouve entre le `\begin` et le `\end`. Il suffit de recopier la définition ci-dessous du nouvel environnement dans le préambule du document.

```
\newenvironment{changemargin}[2]{\begin{list}{}{%\nsetlength{\topsep}{0pt}%\nsetlength{\leftmargin}{0pt}%\nsetlength{\rightmargin}{0pt}%\nsetlength{\listparindent}{\parindent}%\nsetlength{\itemindent}{\parindent}%\nsetlength{\parsep}{0pt plus 1pt}%\naddtolength{\leftmargin}{#1}%\naddtolength{\rightmargin}{#2}%\nitem }{\end{list}}
```

Puis de rentrer, par exemple, pour décaler de 2 cm la marge gauche :

```
\begin{changemargin}{2cm}{0cm}\n...\nend{changemargin}
```

Évidemment, ça fonctionne pour décaler le `verbatim` !

1.3 Package moreverb

En utilisant un package supplémentaire, on peut arranger la présentation des textes en `verbatim`. On va utiliser `moreverb` (il en existe d'autres !) dont la documentation s'appelle [moreverb.pdf](#).

On commencera donc par entrer : `\usepackage{moreverb}`.

1.3.1 Encadrement

On peut encadrer un environnement `verbatim` en utilisant l'environnement `boxedverbatim` qui est compatible avec l'environnement `changemargin`.

En tapant	<pre>\begin{boxedverbatim} def maxi(L): max=L[0] for k in L: if k>max: max=k return max \end{boxedverbatim}</pre>	on obtient	<div style="border: 1px solid black; padding: 5px;"> <pre>def maxi(L): max=L[0] for k in L: if k>max: max=k return max</pre> </div>
-----------	------------------------------------------------------------------------------------------------------------------------------------------------------	------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------

1.3.2 Numérotation de lignes

L'environnement `listing` numérote automatiquement les lignes situées entre le `\begin` et le `\end`. Il faut entrer un paramètre obligatoire qui est le numéro de la première ligne.

	<code>\begin{listing}{1}</code>				
	<code>def maxi(L):</code>			<code>1 def maxi(L):</code>	
	<code>max=L[0]</code>			<code>2 max=L[0]</code>	
En tapant	<code>for k in L:</code>		on obtient	<code>3 for k in L:</code>	
	<code>if k>max:</code>			<code>4 if k>max:</code>	
	<code>max=k</code>			<code>5 max=k</code>	
	<code>return max</code>			<code>6 return max</code>	
	<code>\end{listing}</code>				

L'environnement `listing` est compatible avec l'environnement `changemargin`.

Dans un `listing` très long, on peut vouloir ne numéroté que les lignes espacées d'un `pas`; il faut donc entrer le paramètre `pas` en option (entre crochets).

	<code>\begin{listing}[2]{10}</code>				
	<code>def maxi(L):</code>			<code>10 def maxi(L):</code>	
	<code>max=L[0]</code>			<code>max=L[0]</code>	
En tapant	<code>for k in L:</code>		on obtient	<code>12 for k in L:</code>	
	<code>if k>max:</code>			<code>if k>max:</code>	
	<code>max=k</code>			<code>14 max=k</code>	
	<code>return max</code>			<code>return max</code>	
	<code>\end{listing}</code>				

Il ne me reste plus qu'à trouver comment on peut, à la fois, encadrer et numéroté les lignes!