

UTILISER DES BOUCLES AVEC TIKZ

Nous avons découvert cet article de Harald LICHTENSTEIN dans la dernière livraison de Die TeXnische Komödie, la revue de nos homologues germanophones. Sa prose limpide et très pédagogique nous a fait penser qu'il pourrait intéresser un public francophone.

Lorsque nous dessinons avec TikZ, il est possible d'économiser un temps précieux en utilisant des boucles. Les boucles permettent l'affichage de chiffres ou de lettres. Le présent article explique comment les utiliser.

Des boucles simples en TikZ avec `\foreach`

Dans le chapitre 2.20 du manuel de TikZ [1] est expliquée la syntaxe des boucles.

code	résultat
<pre> 1 \$\begin{tikzpicture} 2 \foreach \x in {1, 2, 3}{ 3 \node at (\x, 0) {\x} ; 4 } 5 \end{tikzpicture}\$ </pre>	<p style="text-align: center;">1 2 3</p>

Ce premier exemple, très simple, nous donne l'image avec les trois nœuds 1, 2, 3.

La syntaxe fonctionne ainsi :

- les lignes 1 & 5 ouvrent et ferment l'environnement TikZ;
- à la ligne 2, la boucle commence par l'instruction `\foreach`; la variable `\x` est déclarée et prendra successivement les valeurs 1, 2 et 3;
- l'accolade de la fin la ligne 2 marque le début des instructions qui seront répétées;
- ces instructions figurent en ligne 3 : pour chacune des valeurs prises par la variable `\x`, l'instruction `\node` va créer un nœud de coordonnées `$(\x, 0)$` dont le contenu, délimité par les accolades, sera `\x`;
- l'accolade de la ligne 4 met un terme à la boucle.

De la même manière, les valeurs prises par la variable peuvent être des lettres :

code	résultat
<pre> 1 \begin{tikzpicture} 2 \foreach \n in {A, B, C} {\$\n\$} ; 3 \end{tikzpicture} </pre>	<p style="text-align: center;">ABC</p>

Mais dans ce cas, la boucle ne crée pas de nœuds : elle positionne côte-à-côte, sur la même ligne, les différentes valeurs prises par la variable, voilà tout. Les informations fournies à TikZ par ce bref code ne permettent pas de contrôler le positionnement de ces lettres. Par ailleurs, l'affichage des lettres ne fonctionne qu'en mode mathématique : si l'on enlève les balises `$`, rien n'est affiché.

Si la boucle ne doit être répétée que peu de fois, les différentes valeurs prises par la variable peuvent être déclarées explicitement. Mais TikZ peut aussi les déduire des instructions qui lui sont données.

code	résultat
<pre> 1 \begin{tikzpicture} 2 \foreach \n in {A, ..., Z} {\$\n\$} ; 3 \end{tikzpicture} </pre>	

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
```

code
résultat

Les trois points (...) déclarent implicitement les valeurs successivement prises par la variable `\n`.

Il arrive que nous ayons besoin de plusieurs variables dans une même boucle. Nous pouvons alors utiliser la syntaxe suivante :

code	résultat
<pre>1 \$\begin{tikzpicture} 2 \foreach \x/\n in {1/A, 2/B, 3/C}{ 3 \node at (\x, 0) {\n} ; 4 } 5 \end{tikzpicture}\$</pre>	A B C

Dans la boucle `\foreach` sont déclarées deux variables, `\x` et `\n`, séparées par le signe `/`. Entre les accolades, les valeurs prises par ces variables sont explicitement déclarées de la même manière : deux valeurs séparées par le signe `/`.

Malheureusement, la déclaration implicite des valeurs ne fonctionne pas pour plusieurs variables déclarées ensemble : il n'est pas possible d'utiliser les trois points pour les déclarer.

```
1  \begin{tikzpicture}
2  \foreach \x/\n in {1/A, .../..., 26/Z}{ << SYNTAXE ERRONÉE
3    \node at (\x, 0) {\n} ;
4  }
5
6  \foreach \x/\n in {1/A, ..., 26/Z}{ << SYNTAXE ERRONÉE
7    \node at (\x, 0) {\n} ;
8  }
9  \end{tikzpicture}
```

Des boucles complexes en TikZ avec `\foreach`

Une astuce, ou plus exactement une option fournie à l'instruction `\foreach`, nous permet de nous tirer de ce mauvais pas sans avoir à déclarer explicitement chaque paire de valeurs.

code
<pre>1 \begin{tikzpicture}[x = 0.35 cm, y = 0.35 cm] 2 \foreach \n [count=\x] in {A, ..., Z}{ 3 \node [anchor = base] at (\x, 0) {\n} ; 4 } 5 \end{tikzpicture}</pre>

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

code
résultat

Quelles nouvelles instructions ont donc été ajoutées ?

- En ligne 1, nous avons réduit l'espacement standard des abscisses : originellement espacées de 1 cm, les valeurs entières sont désormais espacées de 0,35 cm, de manière à ce que les 26 lettres de l'alphabet entrent sur une ligne ;
- en ligne 2, le code devient intéressant : nous introduisons le compteur `[count=\x]`, qui va incrémenter la variable `\x` à chaque boucle. De cette manière, la boucle va

parcourir toutes les lettres de l'alphabet tout en générant les valeurs `\x` nécessaires aux coordonnées des nœuds ;

- en ligne 3, nous passons le paramètre `[anchor = base]` aux nœuds générés par la boucle, de manière à ce que les lettres J et Q, qui en capitales présentent un jambage, ne dépassent pas la hauteur de X ; la position du nœud `(\x, 0)` suit ce paramètre, et la définition du nœud est complétée par son contenu `\n`.

Lors de la rédaction – toujours en cours – d'un livre consacré aux méthodes de cryptographie, nous avons eu recours à une figure carrée formée par les lettres de l'alphabet : la *table de Vigenère*. La composition de celle-ci nous a causé quelques migraines, jusqu'à ce que les boucles TikZ nous fournissent le résultat désiré.

Nous avons utilisé une option offerte par les boucles TikZ : l'entrelacement.

code

```

1 \begin{tikzpicture}[x = 0.35 cm, y = 0.35 cm]
2 \foreach \m [count=\y] in {Z, ..., A}{
3   \foreach \n [count=\x] in {\m, ..., Z}{
4     \node [anchor = base] (P\x\y) at (\x, \y) {\n} ;
5   }
6 }
7 \end{tikzpicture}

```

résultat

Quelles nouvelles instructions ont été maintenant ajoutées ?

- En ligne 2, les valeurs sont incrémentées en sens inverse : TikZ va successivement attribuer à la variable `\m` une lettre de l'alphabet, de Z à A ; cette instruction forme la boucle extérieure ;
- à l'intérieur de celle-ci, en ligne 3, la boucle intérieure définit toujours la variable `\n`, avec la particularité qu'elle hérite la valeur `\m` de la boucle extérieure ; cette valeur est utilisée pour définir l'une des extrémités de la plage que prendront les valeurs successives : en toute logique, la première boucle donnera les valeurs de Z à... Z, la deuxième de Y à Z, etc. ; ces incréments nous montrent que nous approchons du triangle désiré ;

- la troisième ligne définit les nœuds eux-mêmes, avec une particularité : ces nœuds sont nommés au sein des parenthèses ($P\backslash x\backslash y$) précédant le positionnement du nœud⁴⁷; nous remarquerons que les variables $\backslash x$ et $\backslash y$ sont utilisées lors de cette opération de nommage : ainsi, le premier nom sera P11 et le dernier P2626.

Nous obtenons donc une figure comprenant 26 lignes, chacune d'entre elles composée d'une à vingt-six lettres :

- lors de la première boucle, $\backslash m = Z$, $\backslash y = 1$, $\backslash n = Z$ et $\backslash x = 1$; TikZ va donc composer cette ligne selon l'instruction suivante :

```
1 \node [anchor = base] (P11) at (1, 1) {Z} ;|
```

Il s'agit du Z formant le coin inférieur du triangle de lettres;

- lors de la dernière boucle, $\backslash m = A$, $\backslash y = 26$, $\backslash n = Z$ et $\backslash x = 26$; TikZ va donc composer cette ligne selon l'instruction ci-après :

```
1 \node [anchor = base] (P2626) at (26, 26) {Z} ;|
```

Il s'agit du Z formant le coin supérieur droit du triangle.

Pour compléter la table de Vigenère, il nous faut à présent en composer la moitié inférieure gauche.

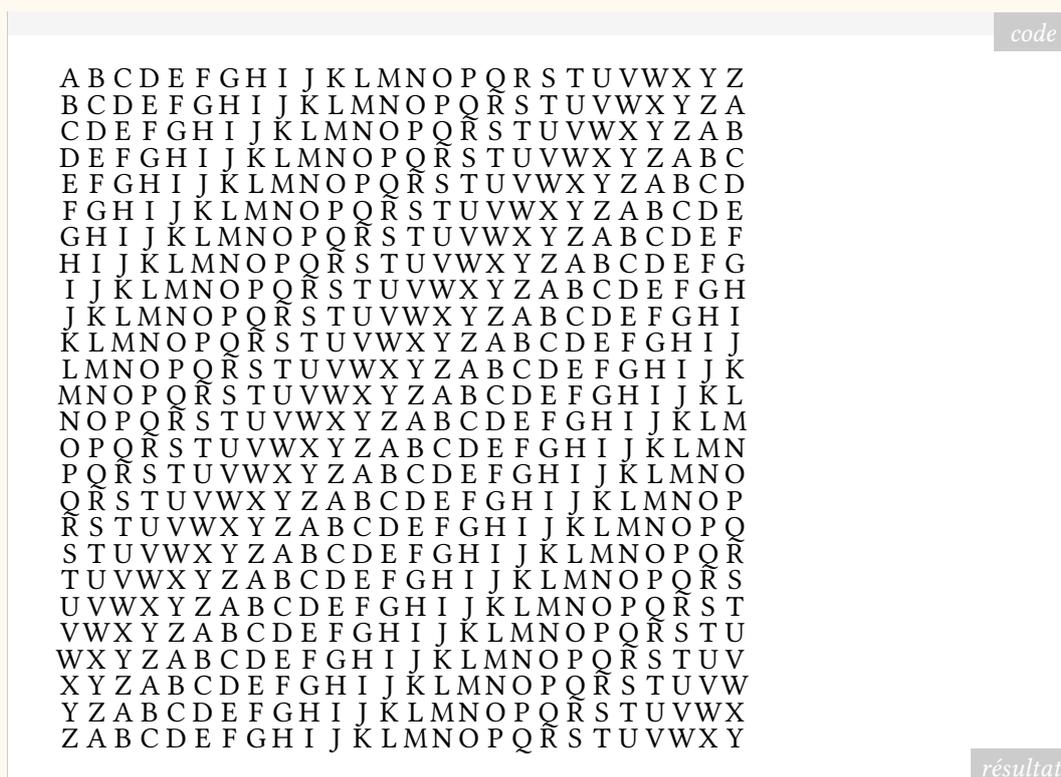
```

1 \begin{tikzpicture}[x = 0.35 cm, y = 0.35 cm]
2   % boucle générant le triangle supérieur
3   \foreach \m [count=\y] in {Z, ..., A}{
4     \foreach \n [count=\x] in {\m, ..., Z}{
5       \node [anchor = base] (P\backslash x\backslash y) at (\backslash x, \backslash y) {\backslash n} ;
6     }
7   }
8
9   % boucle générant le triangle inférieur
10  \foreach \m [count=\y] in {A, ..., Y}{
11    \foreach \n [count=\x] in {\m, ..., A}{
12      \node [anchor = base] (P\backslash x\backslash y) at (27-\backslash x, 26-\backslash y) {\backslash n} ;
13    }
14  }
15 \end{tikzpicture}

```

code

47. Ce positionnement du nœud est défini par $\text{at}(x, y)$.



Pour conclure cette étude, quelles nouvelles instructions avons-nous ajoutées ?

De la ligne 9 à la ligne 14, nous avons inséré un nouvel entrelacement de boucles. La boucle extérieure parcourt les lettres de l'alphabet, dans l'ordre alphabétique, mais seulement jusqu'à la lettre Y : en effet, la ligne supérieure de la table, qui se termine par un Z, existe déjà.

La boucle intérieure hérite à nouveau des valeurs de la boucle extérieure et commence par écrire un A sur la deuxième ligne⁴⁸ de la figure, en dernière position ; nous obtenons ce résultat par le calcul, les coordonnées étant obtenues grâce à $(27-\backslash x, 26-\backslash y)$.

Ainsi, lors de la première itération de la boucle générant le triangle inférieur, $\backslash m = A$, $\backslash y = 1$, $\backslash n = A$ et $\backslash x = 1$; TikZ va donc composer cette ligne selon l'instruction suivante :

```
1 \node [anchor = base] (P11) at (26, 25) {A} ;
```

Il s'agit du A situé à l'extrémité droite de la deuxième ligne de la table.

Et la dernière itération de la boucle donne les valeurs $\backslash m = Y$, $\backslash y=25$, $\backslash n = Y$ et $\backslash x = 1$; TikZ va donc composer cette ligne selon l'instruction suivante :

```
1 \node [anchor = base] (P125) at (26, 1) {Y} ;
```

Il s'agit du Y situé à l'extrémité droite de la dernière ligne de la table, qui est désormais complète.

Harald LICHTENSTEIN

48. Rappelons qu'une figure TikZ commence en bas à gauche, au point de coordonnées $(0, 0)$. La ligne inférieure de notre table commence au point de coordonnées $(1, 1)$. Dans notre texte, nous désignons les lignes de la figure par des adjectifs ordinaux, qui sont évidemment utilisés dans le sens de lecture de la langue française : ainsi, la *première* ligne est celle du haut.

La version originale de cet article est parue dans le numéro 2/2021 de *Die T_EXnische Komödie*, pages 48 à 52, sous le titre *Schleifen in Tikz*.

Traduction de Patrick BIDEAULT

Références

- [1] Till TANTAU et al. *The Pgf package. Create PostScript and PDF graphics in TeX*. Version 3.1.9a. URL : <https://ctan.org/pkg/pgf> (visité le 02/07/2021).

BRÈVES TYPOGRAPHIQUES

La biennale de Chaumont

Inauguré en 2016 dans la préfecture de la Haute-Marne, le *Signe* de Chaumont est le centre d'art contemporain dédié au design graphique. En plus du Festival international de l'affiche et du graphisme, il accueille désormais une biennale internationale de design graphique, qui a lieu cette année.

Son programme, placé sous le thème de la viralité, est très riche ; c'est évidemment la propagation virale des informations et des images qui est examinée, et plus largement la viralité des représentations. Sont présentées diverses expositions monographiques consacrées à Fanette MELLIER, à Michiel SCHUURMAN, au studio iranien Kargah, à Michel LE PETIT DIDIER. Enfin, les écrans du *Design in Motion Festival* d'Amsterdam ont fait le déplacement jusqu'à Chaumont, ce qui permet aux visiteurs de découvrir le meilleur du graphisme en mouvement⁴⁹ ; qui sait, peut-être qu'un utilisateur de (L^A)T_EX créera une passerelle entre les logiciels que nous utilisons et *DrawBot* ou *p5.js* ! Quoi qu'il en soit, une visite au *Signe* de Chaumont est toujours enrichissante, et cet été plus particulièrement. La biennale dure jusqu'au 21 novembre prochain.

In memoriam Wolfgang WEINGART

Nous apprenons le décès du célèbre typographe allemand Wolfgang WEINGART.

Né en février 1941 dans la vallée de Salem, sur les rives allemandes du lac de Constance, où il passe son enfance, il déménage à l'adolescence à Lisbonne, avec ses parents, où il réside entre 1954 et 1956. Ce séjour aura une influence importante sur la vie de ce créateur joyeusement subversif. De retour en Allemagne, c'est à Stuttgart, à la Merz Akademie d'Albrecht Leo Merz qu'il commence en 1958 des études artistiques. L'imprimerie de l'école lui permet découvrir la composition typographique. Il s'essaye également à la création de caractères, sur linoléum et sur bois.

Il travaille ensuite chez Druck + Graphik Wilhelm Ruwe e. K., une imprimerie réputée où il apprend la composition au plomb. Il y fait la connaissance du graphiste de la maison, l'artiste Karl August HANKE (1914 – 1994), qui aura une influence déterminante sur la pratique de WEINGART ; il l'encouragea notamment à étudier en Suisse. Après avoir fait à Bâle, en 1963, la connaissance d'Emil RUDER et d'Armin HOFFMAN, WEINGART y déménage en 1964 pour s'établir comme graphiste et y suivre en auditeur libre les cours de la *Schule für Gestaltung Basel* (l'école de design de Bâle, également connue sous le nom de *Kunstgewerbeschule Basel*). Il commence à publier, notamment le très beau *Kinder 1 – Orient Zeichnen – Sonne*

49. L'auteur de ces lignes a publié un article sur ce festival dans le numéro 85 de *Graphê*, la revue de l'association pour la promotion de l'art typographique.