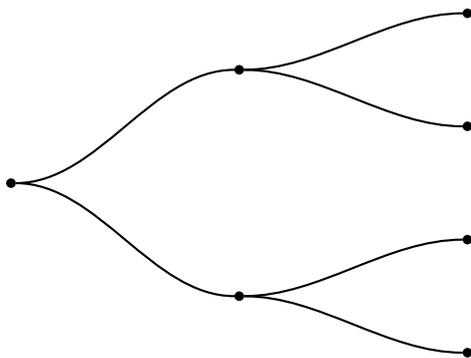


Chronique 9

Compléments sur les arbres

9.1 Un peu de fantaisie

Voici un arbre binaire un peu fantaisiste :



C'est un des arbres de la page 25 dont les arêtes ont été modifiées.

Par défaut, une arête (`\edge`) est un segment, c'est-à-dire de type `\ncline`; on peut modifier globalement la forme des arêtes, ou une par une comme dans le paragraphe suivant.

Les arêtes de l'arbre de ce paragraphe ont été construites avec l'instruction `\nccurve` décrite en page 41; elles sont définies au moyen de l'instruction `\psedge` qu'il suffit de redéfinir ainsi :

```
\def\psedge{\nccurve[angleA=0,angleB=180]}
```

Autrement dit, au lieu d'avoir une arête rectiligne (`\ncline`), on a une arête courbe (`\nccurve`) dont on a défini l'angle de départ à 0° et l'angle d'arrivée à 180° .

À la place de `\def`, on aurait pu utiliser `\renewcommand`.

Et si l'on veut que cette redéfinition des formes des arêtes n'affecte pas les arbres qui vont suivre, on insèrera le code de l'arbre entre deux accolades :

```
{
  \def\psedge{\nccurve[angleB=180]}
  \pstree[treemode=R,treesep=1.5cm,levelsep=3cm,nodesep=0pt]
    {\Tdot}
    {\pstree{\Tdot}{\Tdot \Tdot}}
    {\pstree{\Tdot}{\Tdot \Tdot}}
}
```

Et si on veut des flèches comme arêtes, on peut redéfinir `\psedge` ainsi :

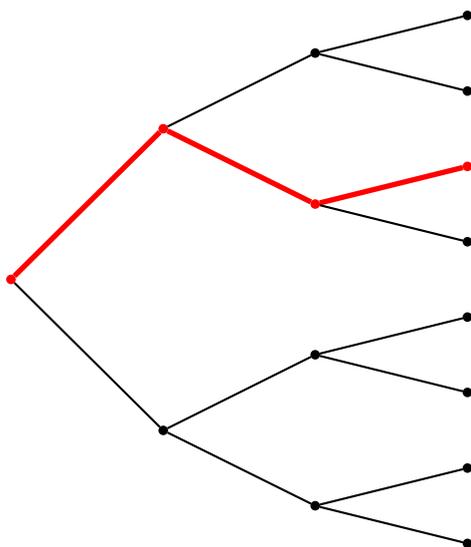
```
\def\psedge{\ncline[arrowsize=3pt 3,nodesep=3pt]{->}}
```

9.2 Chemin coloré

On a vu dans la première chronique consacrée aux arbres (voir page 29) comment colorer ou personnaliser un arbre ou un sous-arbre ; mais comment faire si on veut ne colorer qu'une branche d'un arbre, par exemple pour marquer un parcours ?

On ne peut pas utiliser la méthode du paragraphe précédent qui modifie toutes les arêtes de l'arbre.

Dans l'arbre ci-dessous j'ai coloré en rouge un parcours et les nœuds correspondants :



Les nœuds ont été définis par l'instruction `\Tdot` ; pour les dessiner en rouge, il suffit de rajouter l'option `linecolor=red` au bon endroit.

Pour colorer une arête d'un arbre sans forcément colorer toutes les arêtes de cet arbre, il faut définir un nouveau type d'arête.

On va définir le type `\ARouge` qui sera un segment (`\ncline`) rouge d'épaisseur 2 points :

```
\def\ARouge{\ncline[linecolor=red,linewidth=2pt]}
```

L'instruction `\def` pourrait ici être remplacée par `\newcommand`.

Une fois qu'on a défini cette arête d'un nouveau type, il suffit, pour la faire afficher, d'entrer en option `edge=\ARouge`.

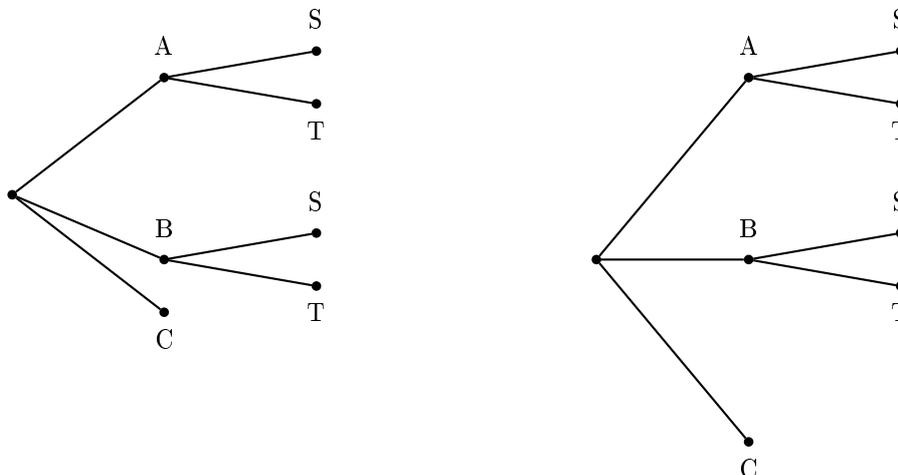
Il s'agit donc d'une utilisation locale de `edge`, contrairement à la redéfinition de `\psedge`.

L'arbre binaire de ce paragraphe a été dessiné en tapant :

```
\def\ARouge{\ncline[linecolor=red,linewidth=2pt]}
\psset{treemode=R,nodesep=0mm,levelsep=20mm,treesep=10mm}
\pstree{\Tdot[linecolor=red]}
{
  \pstree{\Tdot[linecolor=red,edge=\ARouge]}
  {
    \pstree{\Tdot}{\Tdot \Tdot}
    \pstree{\Tdot[linecolor=red,edge=\ARouge]}
    {\Tdot[linecolor=red,edge=\ARouge] \Tdot}
  }
  \pstree{\Tdot}
  {
    \pstree{\Tdot}{\Tdot \Tdot}
    \pstree{\Tdot}{\Tdot \Tdot}
  }
}
```

9.3 Répartition des branches

Voici deux arbres représentant la même situation. Lequel est le plus joli ?



Celui de droite, bien sûr !

Dans le tracé d'un arbre, les branches sont dessinées « au mieux » en fonction des informations dont dispose L^AT_EX.

L'arbre de gauche a donc été défini sans précaution particulière : trois branches partant de la racine, deux branches partant de la première branche, deux partant de la deuxième, et aucune partant de la troisième ; le résultat n'est pas très beau.

Pour tracer l'arbre de droite, j'ai tracé trois branches partant de la racine, puis deux branches partant de chacune de ces branches de première génération. Mais comme la troisième branche n'a pas de successeurs, j'ai utilisé l'instruction `\phantom` (voir chronique 8 de la saison 1) pour ne pas afficher le sous-arbre du bas.

Voici les codes utilisés ; pour l'arbre de gauche :

```
\psset{treemode=R,nodesep=0mm,levelsep=20mm,treesep=7mm}
\pstree{\Tdot}
{
  \pstree{\Tdot~[tnpos=a]{A}}
  {\Tdot~[tnpos=a]{S} \Tdot~[tnpos=b]{T}}
  \pstree{\Tdot~[tnpos=a]{B}}
  {\Tdot~[tnpos=a]{S} \Tdot~[tnpos=b]{T}}
  \Tdot~[tnpos=b]{C}
}
```

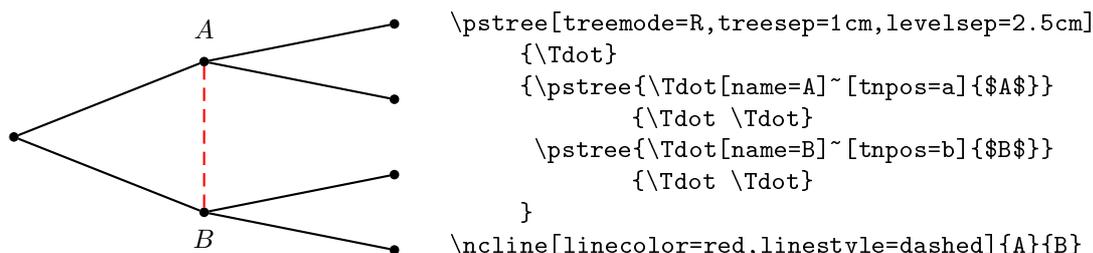
et pour l'arbre de droite :

```
\psset{treemode=R,nodesep=0mm,levelsep=20mm,treesep=7mm}
\pstree{\Tdot}
{
  \pstree{\Tdot~[tnpos=a]{A}}
  {\Tdot~[tnpos=a]{S} \Tdot~[tnpos=b]{T}}
  \pstree{\Tdot~[tnpos=a]{B}}
  {\Tdot~[tnpos=a]{S} \Tdot~[tnpos=b]{T}}
  \pstree{\Tdot~[tnpos=b]{C}}
  {\phantom{\Tdot~[tnpos=a]{S} \Tdot~[tnpos=b]{T}}}
}
```

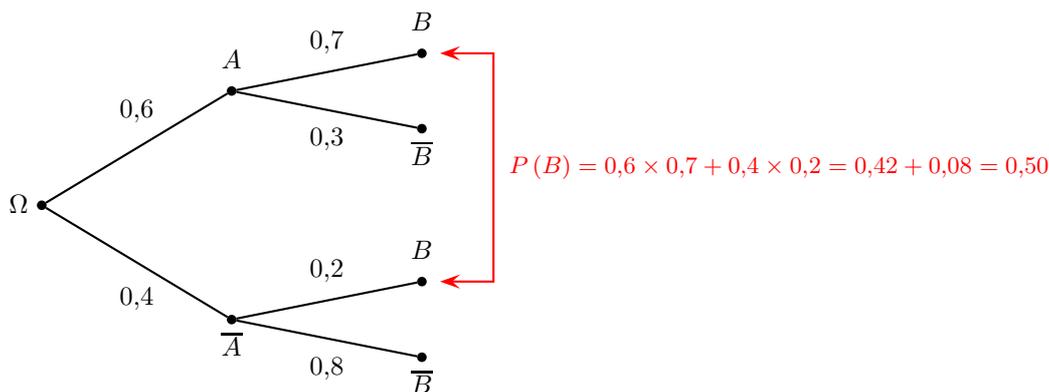
9.4 Nom des nœuds

On peut, dans un arbre, tracer des arcs entre des nœuds dont l'un n'est pas le successeur de l'autre ; il suffit pour cela de nommer les nœuds que l'on veut relier au moyen de l'option `name=...`, puis de tracer un segment ou un arc entre les deux nœuds qui ont été nommés.

Dans l'exemple ci-dessous on relie les nœuds au moyen d'un segment tracé avec l'instruction `\ncline` :



Voici un autre exemple, peut-être plus exploitable :



```

\pstree[treemode=R,treesep=1cm,levelsep=2.5cm]
{\Tdot~[tnpos=1]{$\Omega$}}
{
\pstree{\Tdot~[tnpos=a]{$A$}\taput{$0,6$}}
{\Tdot[name=b1]~[tnpos=a]{$B$}\taput{$0,7$}
\Tdot~[tnpos=b]{$\overline{B}$}\tbput{$0,3$}}
\pstree{\Tdot~[tnpos=b]{$\overline{A}$}\tbput{$0,4$}}
{\Tdot[name=b2]~[tnpos=a]{$B$}\taput{$0,2$}
\Tdot~[tnpos=b]{$\overline{B}$}\tbput{$0,8$}}
}
\psset{arrowsize=3pt 3}
\ncbar[arm=20pt,nodesep=5pt,linecolor=red]{<->}{b1}{b2}
\Aput{\red \small $P(B) = 0,6 \times 0,7 + 0,4 \times 0,2 = 0,42 + 0,08 = 0,50$}

```

Les explications de `\ncbar` et `arm` se trouvent à la page 44, et on parle la première fois de `nodesep` en page 36.

À la place de `\ncbar[arm=20pt,nodesep=5pt,linecolor=red]{<->}{b1}{b2}` on peut mettre : `\nccurve[nodesep=5pt,angleA=-30,angleB=30,linecolor=red]{<->}{b1}{b2}`

L'utilisation de `\nccurve` est expliquée en page 41.