

## Exercice 1 : Description d'arbres binaires

On s'intéresse dans cet exercice à des arbres dont chaque nœud possède au plus 2 fils. On souhaite décrire ces arbres par une représentation textuelle, comme suit :

- une feuille est représentée par "f" ;
- un nœud unaire (qui a un seul fils) est représenté par "u" ;
- un nœud binaire (qui a deux fils) est représenté par "b".

On utilise de plus les conventions suivantes : une feuille est représentée directement par "f", et on choisit d'entourer les sous-arbres des nœuds unaire et binaire de parenthèses, avec séparation par une virgule dans le cas binaire. On aura par exemple les arbres et leur représentation textuelle de la figure 1. Les chaînes  $b(f)$  ou  $u(u)$  ne représentent pas un arbre.

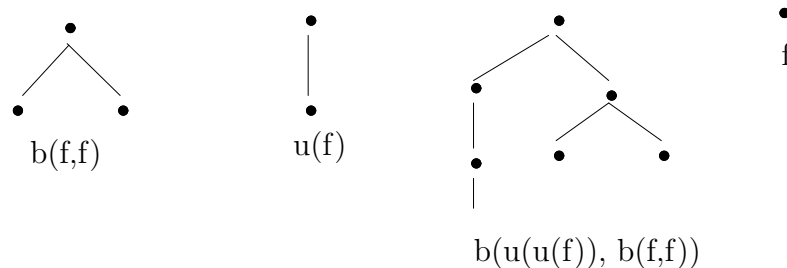


FIG. 1 – Un arbre

**Q 1.1 :** Donner une grammaire algébrique qui décrit l'ensemble des représentations textuelles des arbres décrits plus hauts. □

**Q 1.2 :** Cette grammaire est-elle ambiguë ? Est-elle LL(1) ? □

On souhaite associer à chaque nœud un caractère, pour ainsi interpréter nos arbres comme des cas très particuliers d'arbres lexicographiques<sup>1</sup> : ils décrivent des ensembles ordonnés de mots, qu'on lit en glissant sur les branches de la racine vers les feuilles. Par exemple, les arbres de la figure 2 décrivent les ensembles  $\{ la, le \}$ ,  $\{ do \}$  et  $\{ aime, are, art \}$ .

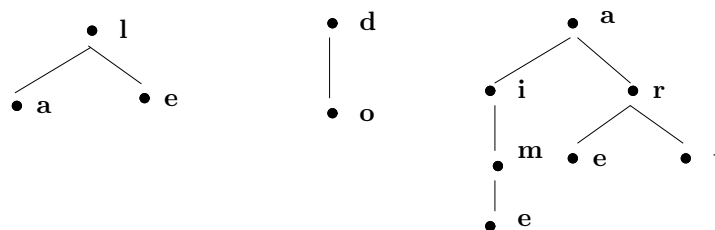


FIG. 2 – Un arbre décoré

**Q 1.3 :** Proposer une modification de la grammaire qui permette d'associer à chaque nœud un caractère, de telle sorte que la grammaire obtenue soit LL(1). □

**Q 1.4 :** Attribuer la grammaire pour connaître la longueur du plus long mot représenté par l'arbre. Préciser le type de l' / des attribut(s) utilisé(s) et s'il est / sont hérité(s) ou synthétisé(s). □

**Q 1.5 :** On oublie la question précédente. Attribuer la grammaire pour connaître le nombre de mots représentés par l'arbre. Préciser le type de l' / des attribut(s) utilisé(s) et s'il est / sont hérité(s) ou synthétisé(s). □

<sup>1</sup>Un arbre lexicographique correspond à un cas bien plus général.

**Q 1.6 :** On oublie la question précédente. Attribuer la grammaire pour associer à chaque feuille le mot représenté par sa branche. Préciser le type de l' / des attribut(s) utilisé(s) et s'il est / sont hérité(s) ou synthétisé(s).  $\square$

**Q 1.7 :** Continuer l'attribution de la grammaire pour associer à l'axiome l'ensemble des mots représentés par l'arbre. Préciser le type de l' / des attribut(s) utilisé(s) et s'il est / sont hérité(s) ou synthétisé(s).  $\square$

**Q 1.8 :** En reprenant l'arbre de droite de la figure 2, dessiner les dépendances de données entre attributs et trouver un ordre qui permette de les calculer.  $\square$

**Q 1.9 :** La grammaire de la question 7 est-elle LL(1)-attribuée?  $\square$

**Q 1.10 :** Donner un analyseur récursif descendant LL(1) qui reconnaît les arbres et calcule les attributs de la question 7.  $\square$

**Q 1.11 :** Expliquer comment il est possible d'attribuer la grammaire pour associer à l'axiome l'ensemble des mots représentés par l'arbre, sans utiliser d'attribut hérité.  $\square$

## Exercice 2 : Évaluation d'entiers et bases

On considère la grammaire suivante qui décrit des nombres octaux et décimaux : son axiome est *nombre*, ses terminaux sont  $\{\text{chiffre}, \text{o}, \text{d}\}$ , ses non-terminaux sont  $\{\text{nombre}, \text{base}, \text{suiteChiffre}\}$  :

$\text{nombre} \rightarrow \text{suiteChiffre base} \quad \text{base} \rightarrow \text{o} \mid \text{d}$   
 $\text{suiteChiffre} \rightarrow \text{chiffre} \mid \text{suiteChiffre chiffre}$

**Q 2.1 :** Donner un arbre syntaxique pour le mot `chiffre chiffre d`.  $\square$

La valeur du nombre est par exemple 17 pour `21o` ( $17 = 2 \cdot 8 + 1$ ) et 21 pour `21d`. Noter que `324o` a pour valeur  $(3 \cdot 8 + 2) \cdot 8 + 4 = 3 \cdot 8^2 + 2 \cdot 8^1 + 4 \cdot 8^0 = 212$ .

**Q 2.2 :** Attribuez la grammaire de telle sorte qu'un attribut de l'axiome donne la valeur du nombre reconnu. Précisez pour chaque attribut le symbole auquel il est attaché, son type, s'il est hérité ou synthétisé, et si sa valeur est fournie par l'analyseur lexical.  $\square$

**Q 2.3 :** Cette grammaire est-elle L-attribuée?  $\square$

**Q 2.4 :** Quel contrôle sémantique manque-t-il dans cette grammaire attribuée (en quoi est-elle trop permissive)?  $\square$

**Q 2.5 :** On remplace la production  $\text{suiteChiffre} \rightarrow \text{suiteChiffre chiffre}$  par la production  $\text{suiteChiffre} \rightarrow \text{chiffre suiteChiffre}$ . Le langage engendré est-il le même que précédemment? Est-il nécessaire de modifier l'attribution pour attacher à l'axiome la valeur du nombre reconnu?  $\square$

## Exercice 3 : Calculs ensemblistes

On s'intéresse à la grammaire  $G_E$  suivante des calculs ensemblistes (très simplifiés). Les terminaux sont  $\{\text{e}, \cup, \setminus, (, ), \mathcal{C}\}$  où **e** représente un ensemble décrit en énumération et l'opérateur  $\mathcal{C}$  est l'opérateur de prise de complémentaire.

$$E \rightarrow \text{e} \mid ( E ) \mid \mathcal{C} E \mid E \cup E \mid E \setminus E$$

On souhaite calculer à partir de la grammaire la valeur de l'expression ensembliste. Pour ce faire, on suppose donnée une interface Java **Ensemble** :

```

public interface Ensemble {
    public Ensemble complementaire();
    public Ensemble union(Ensemble e);
    public Ensemble difference(Ensemble e);
    public static Ensemble ensembleVide;
    ...
}

```

On suppose aussi que le terminal **e** possède un attribut **val** synthétisé de type **Ensemble** fixé par l'analyseur lexical.

**Q 3.1 :** Pourquoi est-ce qu'attribuer la grammaire  $G_E$  n'a aucun sens? □

**Q 3.2 :** Sachant que les opérateurs binaires sont associatifs à gauche, que l'opérateur de prise de complémentaire est prioritaire sur celui de différence, qui est prioritaire sur celui d'union, donner une grammaire  $G_E^1$  équivalente à  $G_E$ , non ambiguë. □

**Q 3.3 :** Attribuer  $G_E^1$  pour calculer la valeur de l'ensemble décrit. □

**Q 3.4 :** Cette grammaire est-elle S-attribuée? L-attribuée? □

**Q 3.5 :**  $G_E^1$  est-elle LL(1)? Si ce n'est pas le cas donner une grammaire  $G_E^2$  qui est équivalente à  $G_E^1$  et est LL(1). □

**Q 3.6 :** Recommencer l'attribution précédente, cette fois sur  $G_E^2$  en se concentrant sur les productions qui diffèrent d'avec  $G_E^1$ . □

## Exercice 4 : Boîtes Tex

Dans un langage de formatage à la Latex on utilise des boîtes pour positionner les caractères les uns par rapport aux autres. Il y a deux types de boîtes :

- les boîtes *horizontales* : les éléments y sont disposés en ligne les uns derrière les autres, alignés sur leur bord supérieur;
- les boîtes *verticales* : les éléments y sont disposés en colonne les uns au dessus des autres, alignés à gauche.

Un élément d'une boîte peut être un caractère ou une boîte. On utilise la grammaire suivante :

$G = (V_T, V_N, \text{boite}, P)$  avec  $V_T = \{\mathbf{h}, \mathbf{v}, (, ), \mathbf{x}\}$ ,  $V_N = \{\text{boite}, \text{listeb}\}$  et  $P$  contient les productions :

```

boite  →  x | h ( listeb ) | v ( listeb )
listeb →  boite | boite listeb

```

Par exemple la description de boîte  $\mathbf{v}(\mathbf{x} \ \mathbf{h}(\mathbf{x} \ \mathbf{v}(\mathbf{x} \ \mathbf{x}) \ \mathbf{x}) \ \mathbf{x})$  correspond à :

```

x
x x x
  x
x

```

On souhaite attribuer  $G$  pour connaître les dimensions d'une boîte, cad sa largeur et sa hauteur, sachant qu'un caractère a comme largeur et hauteur 1. On suppose données les classes Java **Dimension** et **Orientation** :

```

public class Dimension {
    ...
    public int getH(){...};
    public int getL(){...};
}

public class Orientation {
    ...
    public static final HORZ = ...;
    public static final VERT = ...;
}

```

```
public void setH(int h){...};      }  
public void setL(int l){...};  
}
```

**Q 4.1 :** Attribuer  $G$  (en utilisant des instructions Java) pour calculer la dimension d'une boîte, préciser pour chaque attribut à quel terminal/non terminal il est attaché, s'il est hérité ou synthétisé et quel est son type. □

On souhaite effectuer le calcul de dimension de boîte en parallèle d'une analyse syntaxique LL(1).

**Q 4.2 :** La grammaire  $G$  est-elle LL(1) ? Sinon donner une grammaire  $G'$  qui l'est. □

**Q 4.3 :** Sans modifier les productions de  $G'$ , donner une grammaire LL-attribuée qui effectue le calcul. □

**Q 4.4 :** Comment implanter cette analyse sémantique dans un analyseur récursif descendant ? □