

Une classe arbre binaire

Projet individuel à rendre sur Eureka avant le 2 octobre

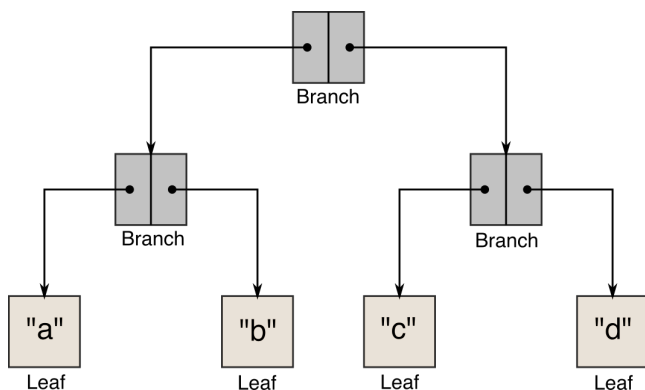
De manière similaire aux listes et flux déjà étudiés, nous pourrions définir une structure d'arbre binaire ainsi :

```
sealed trait Tree[+A]
case class Leaf[A](value:A) extends Tree[A]
case class Branch[A](left:Tree[A], right:Tree[A]) extends Tree[A]
```

Nous pourrions ensuite construire un arbre de profondeur 2 ainsi :

```
Branch( Branch( Leaf("a"), Leaf("b") ), Branch( Leaf("c"), Leaf("d") ) )
```

Sa représentation serait alors :



Voici un autre exemple de profondeur 3 avec des entiers :

```
Branch( Branch( Leaf(3), Branch( Leaf(2), Leaf(4) ) ), Leaf(1) )
```

Questions

Exercice 1

Écrivez une fonction `size` qui retourne le nombre de nœuds (branches et feuilles) dans un arbre passé en paramètre.

```
def size[A](tree:Tree[A]):Int
```

Exercice 2

Écrivez une fonction `maximum` qui retourne la valeur maximum dans un arbre d'entiers `Tree[Int]` (sans bien-sûr assumer que ce dernier est trié !). En Scala l'opération `x.max(y)` (ou `x max y`) est disponible sur les entiers.

```
def maximum(tree:Tree[Int]):Int
```

Exercice 3

Écrivez une fonction `depth` qui retourne la distance maximum entre la racine de l'arbre et une feuille.

```
def depth[A](tree:Tree[A]):Int
```

Exercice 4

Écrivez une fonction `map` similaire à celle des listes qui transforme chaque élément de l'arbre par une fonction passée en paramètre et retourne un nouvel arbre.

```
def map[A,B](tree:Tree[A], f:A => B):Tree[B]
```

Exercice 5 (difficile)

Généralisez `size`, `maximum`, `depth` et `map` en écrivant une fonction `fold` qui abstrait leur similarités. Rappelez-vous ce que nous avons vu sur les listes : `foldRight(1, Nil)(Cons(_,_))` reconstruit la liste. Il existe une relation forte entre `Nil` et l'élément zéro du fold ainsi qu'entre `Cons` et la fonction du fold. Pouvez-vous vous en inspirer pour le fold sur les arbres. Rappelez-vous qu'un arbre n'est constitué que de deux choses, des feuilles et des branches.

Essayez de réécrire `size`, `maximum`, `depth` et `map` avec votre fold.