

# « Programmation récursive » DEUG MIAS — UPMC

queinnec

Octobre 2003 — janvier 2004

## Exercice 1 – Arbres binaires : Interrogation électronique

Cette interrogation porte sur les arbres binaires.

Avec le menu «add teachpack», chargez le teachpack

```
collects/drscheme/tools/mias/exos/arbre-bin/tpabbr.ss
```

Puis écrivez toutes les fonctions demandées en utilisant la barrière d'abstraction des arbres binaires :

```
;; ; ab-arbre : alpha * ArbreBin[alpha] * ArbreBin[alpha] -> ArbreBin[alpha]
;; ; (ab-arbre e B1 B2) rend l'arbre binaire forme de la racine d'etiquette e,
;; ; du sous-arbre gauche B1 et du sous-arbre droit B2
```

```
;; ; ab-vide : -> ArbreBin[alpha]
;; ; (ab-vide) rend l'arbre binaire vide
```

```
;; ; ab-vide? : ArbreBin[alpha] -> bool
;; ; (ab-vide? B) rend vrai ssi B est l'arbre vide
```

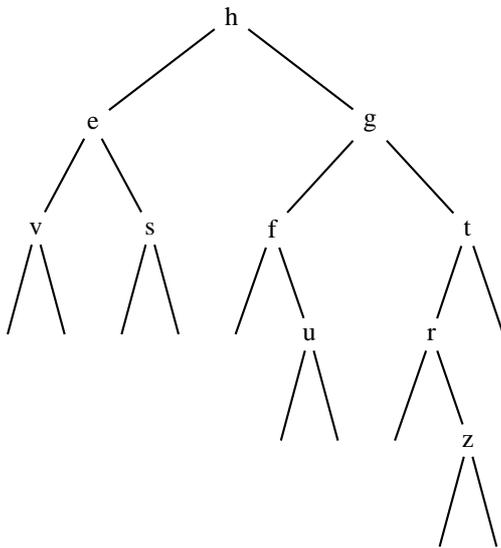
```
;; ; ab-etiquette : ArbreBin[alpha]/non vide/-> alpha
;; ; (ab-etiquette B) rend l'etiquette de la racine de l'arbre B
```

```
;; ; ab-ss-arbre-gauche : ArbreBin[alpha]/non vide/-> ArbreBin[alpha]
;; ; (ab-ss-arbre-gauche B) rend le sous-arbre gauche de B
```

```
;; ; ab-ss-arbre-droit : ArbreBin[alpha]/non vide/-> ArbreBin[alpha]
;; ; (ab-ss-arbre-droit B) rend le sous-arbre droit de B
```

**Question 1 :** Écrire la fonction `ab-nombre-noeuds` qui rend le nombre de nœuds d'un arbre binaire. On compte uniquement les nœuds portant une étiquette, par exemple l'arbre de la figure suivante a 10 nœuds.

Écrire une fonction de test `test-ab-nombre-noeuds`.



**Question 2 :** Écrire la fonction `ab-nombre-noeuds-niveau` qui, étant donné un entier naturel  $k$  et un arbre binaire  $B$  rend le nombre de nœuds étiquetés à niveau  $k$  dans  $B$ . La racine d'un arbre non vide est à niveau 1, et le niveau d'un nœud est égal au niveau de son père augmenté de 1.

Par exemple l'arbre de la figure précédente a 1 nœud à niveau 1, 2 nœuds à niveau 2, 4 nœuds à niveau 3, 2 nœuds à niveau 4 et 1 nœud à niveau 5.

Écrire une fonction de test `test-ab-nombre-noeuds-niveau`.

**Question 3 :** Écrire une fonction `ab-suffixe` qui, étant donné un arbre binaire, retourne la liste de ses étiquettes en ordre suffixe. Par exemple pour l'arbre de la figure précédente, cela donne la liste

(v s e u f z r t g h).

Écrire une fonction de test `test-ab-suffixe`.