Correction exercice 1

Examen 2012 algorithmique L3

Exercice 1

1. La valeur maximale est celle du nœud le plus à droite dont le fils droit est vide.

   Fonction maxABR(a : ABR) : entier
   – Si a est vide alors
     ◦ renvoyer 0
   – Si $D(a)$ est vide alors
     ◦ renvoyer Val(a)
   – renvoyer maxABR($D(a)$)

   Complexité : chaque étape descend d'un cran dans l'arbre, donc $\Theta(h)$.

2. Le nombre de nœuds internes est égal à 1 plus la somme du nombre de nœuds du fils gauche et du fils droit.

   Fonction nbABR(a : ABR) : entier
   – Si a est vide alors
     ◦ renvoyer 0
   – Renvoyer 1+ nbABR($G(a)$)+ nbABR($D(a)$)

   Chaque nœud est visité exactement une fois, donc complexité $\Theta(n)$.

3. Ce sont des arbres "filiformes" pour lesquels chaque nœud a au plus un fils non vide. Cette caractérisation nous donne l'algorithme suivant.

   Fonction filiforme(a : ABR) : booléen
   – Si a est vide alors
     ◦ renvoyer Vrai
   – Si $D(a)$ est vide alors
     ◦ Si $G(a)$ est vide alors
       · renvoyer Vrai
     ◦ sinon
       · renvoyer filiforme($G(a)$)
   – sinon
     ◦ Si $G(a)$ est vide alors
       · renvoyer filiforme($D(a)$)
     ◦ sinon
       · renvoyer Faux

4. L'algorithme va renvoyer un couple (profondeur,valeur) pour pouvoir comparer la profondeur de deux nœuds.

   Fonction prof(a : ABR) : (entier, entier)
   – Si a est vide alors
     ◦ renvoyer (0, 0)
   – Si $G(a)$ et $D(a)$ sont vides alors

◦ renvoyer (0, Val(a))
– $(p_g, v_g) \leftarrow$ prof($G(a)$)
– $(p_d, v_d) \leftarrow$ prof($D(a)$)
– Si $p_g > p_d$ alors
 ◦ renvoyer $(1 + p_g, v_g)$
– sinon
 ◦ renvoyer $(1 + p_d, v_d)$

La fonction finale est donc :

Fonction profond(a : ABR) : entier
– $(p, v) \leftarrow$ prof(a)
– renvoyer v

Chaque nœud est visité exactement une fois donc complexité $\Theta(n)$.

5. (a) On définit une fonction h sur les nœuds. La valeur de h sur un arbre vide est 0.

 Procédure insertion(a : ABR, v : entier)
 – Si a est vide alors
  ◦ $a \leftarrow$ Arbre(v, ArbreVide(), ArbreVide())
  ◦ $h(a) \leftarrow 1$
 – Si $v >$ Val(a) alors
  ◦ insertion($D(a)$, v)
 – sinon
  ◦ insertion($G(a)$, v)
 – $h(a) \leftarrow 1 + \max(h(G(a)), h(D(a)))$

 (b) Il suffit de visiter seulement le sous-arbre le plus profond.

 Fonction prof2(a : ABR) : entier
 – Si a est vide alors
  ◦ renvoyer 0
 – Si $G(a)$ et $D(a)$ sont vides alors
  ◦ renvoyer Val(a)
 – Si $h(G(a)) > h(D(a))$ alors
  ◦ renvoyer prof2($G(a)$)
 – sinon
  ◦ renvoyer prof2($D(a)$)

 On descend dans l'arbre à chaque étape, donc la complexité est $\Theta(h)$.