

Objectifs

À la fin de cette séance, vous devriez être capable de :

- parcourir un arbre binaire dans tous les sens ;

Exercice 1 : Réflexes

On considère des arbres binaires dont les noeuds ont 0, 1 ou 2 fils.

On note h , n et f respectivement la hauteur, le nombre de noeuds et le nombre de feuilles d'un arbre.

Un arbre de hauteur h est dit *équilibré* si :

- toutes ses feuilles sont aux niveaux h ou $h - 1$ (si $h \geq 1$),
- s'il y a des noeuds unaires (des noeuds ayant exactement 1 fils), ils sont au niveau $h - 1$ (si $h \geq 1$).

Dans le cas d'un arbre équilibré, quel est l'ordre de grandeur (justifiez) :

1. de h en fonction de n ?
2. de n en fonction de h ?
3. de f en fonction de n ?

Exercice 2 : Échauffement

Écrivez 3 fonctions qui calculent respectivement la hauteur, le nombre de noeuds et le nombre de feuilles d'un arbre binaire.

Pour écrire ces fonctions, vous avez besoin de manipuler un type abstrait `arbre`. Vous préciserez les primitives d'accès nécessaires.

Exercice 3 : Feuille haute

Pour un arbre binaire A , écrivez une fonction qui calcule la hauteur de la feuille la plus haute (la feuille la plus proche de la racine)

1. par un parcours en profondeur d'abord récursif,
2. par un parcours en largeur d'abord.
3. Donnez les principes permettant la preuve (de correction totale) de ces deux algorithmes (récursif et itératif).

Exercice 4 : Feuilles à la profondeur p

Pour un arbre binaire A et une profondeur p , écrivez une fonction qui calcule le nombre de feuilles à la profondeur p

1. par un parcours en profondeur d'abord récursif,
2. par un parcours en largeur d'abord.

Pour chacun de ces deux algorithmes (récursif et itératif), que pouvez vous dire de son coût ? Y-a-t-il un algorithme moins coûteux que l'autre ?