

1 Généralités

1. Généralités

Exercice 1. *Dessinez des arbres*

Question 1.1. *Dessinez les arbres décrits ci-dessous*

1. $\langle 1; \Delta; \Delta \rangle$
2. $\langle 3; \langle 1; \Delta; \Delta \rangle; \Delta \rangle$
3. $\langle 3; \langle 1; \langle 1; \Delta; \Delta \rangle; \Delta \rangle; \langle 4; \Delta; \langle 5; \langle 9; \Delta; \Delta \rangle; \Delta \rangle \rangle \rangle$

Question 1.2. *Construisez ces arbres avec les opérations primitives.*

Exercice 2. *Nombre d'arbres*

On nomme *squelette* d'un arbre la forme de l'arbre indépendamment de son contenu.

Question 2.1. *Combien y a-t-il de squelettes d'arbres binaires de taille n , lorsque $n \in \llbracket 0, 5 \rrbracket$?*

On désigne par c_n le nombre (de squelettes) d'arbres binaires de taille n .

Question 2.2. *Trouvez une formule de récurrence pour c_n .*

Question 2.3. *Programmez le calcul de c_n et tentez de trouver expérimentalement (en TP) la croissance de c_n en fonction de n .*

2 Parcours d'arbres

2. Parcours d'arbres

Exercice 3. *Nombre de feuilles*

Question 3.1. *Réalisez une fonction calculant le nombre de feuilles d'un arbre.*

Exercice 4. *Dégénéré*

Un arbre est *dégénéré* s'il a au moins deux nœuds et s'il n'a qu'une seule branche.

Question 4.1. *Réalisez un prédicat pour tester si un arbre binaire est dégénéré.*

Exercice 5. *Conversion en liste*

Question 5.1. *Réalisez une fonction qui construit la liste des étiquettes des nœuds d'un arbre*

1. dans l'ordre préfixé ;
2. dans l'ordre postfixé ;
3. dans l'ordre infixé.

Exercice 6. *Affichage d'un arbre*

Question 6.1. *Réalisez une procédure nommée `afficherArbre` qui affiche l'arbre passé en paramètre sous la forme d'un triplet $\langle e; g; d \rangle$ s'il n'est pas vide. L'arbre vide pourra être affiché avec le mot vide.*

Exercice 7. *Nœuds et profondeur*

Question 7.1. Réalisez une fonction qui construit la liste des nœuds d'un arbre a passé en paramètre situés à une profondeur p passée en paramètre.

Exercice 8. Recherche

Question 8.1. Réalisez une fonction `recherche(e,a)` qui donne un sous-arbre de a dont la racine est e s'il y en a, l'arbre vide sinon.

Il est possible que e figure plusieurs fois dans a . Dans ce cas envisagez plusieurs recherches possibles :

1. dans l'ordre préfixé des nœuds ;
2. dans l'ordre postfixé ;
3. dans l'ordre infixé.

Question 8.2. Que donnent ces trois recherches sur l'arbre ci-dessous lorsqu'on recherche 1 ?

$$\langle 1; \langle 1; \langle 1; \Delta; \Delta \rangle; \langle 1; \Delta; \Delta \rangle \rangle; \langle 1; \Delta; \Delta \rangle \rangle$$

Exercice 9. Égalité d'arbres

Deux arbres peuvent être

- *logiquement égaux* : c'est le cas lorsqu'ils sont tous deux vides, ou bien ont même racine et même sous-arbres gauche et droit ;
- ou *physiquement égaux* : lorsqu'en plus ils occupent la même zone mémoire.

Question 9.1. Réalisez deux prédicats pour tester ces deux égalités.