

**□ Exercice 1 : Nombre d’arêtes**

1. Rappeler la définition d’un arbre binaire.
2. Soit  $a$  un arbre binaire à  $n$  noeuds ( $n \geq 1$ ), montrer que  $a$  possède  $n - 1$  arêtes.
3. On rappelle l’implémentation des arbres en OCaml utilisée en cours :

```
1 type ab =  
2   | Vide  
3   | Noeud of ab * int * ab;;
```

En utilisant cette implémentation, écrire une fonction `nb_aretes` de signature `ab -> int` et qui renvoie le nombre d’arêtes d’un arbre binaire

**□ Exercice 2 : Reconstruction****□ Exercice 3 : Tester si un arbre est un ABR**

1. Rappeler la définition d’un arbre binaire de recherche
2. Proposer deux méthodes de complexité linéaire permettant de vérifier qu’un arbre est bien un ABR.
3. Donner l’implémentation de l’une au moins des méthodes.