

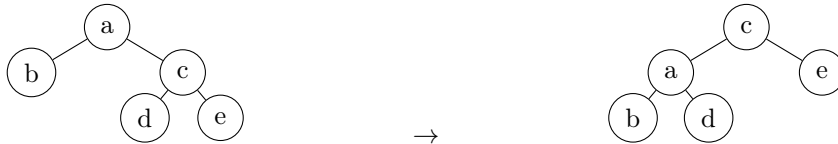
## TD7 : Arbres binaires de recherche auto-équilibrants

### 1 Rotations

L'outil de base pour la plupart des algorithmes de rééquilibrage d'arbres binaires est la *rotation*. La rotation droite transforme un arbre binaire de la façon suivante :



Inversement, la rotation gauche transforme un arbre de la manière suivante :



- a) Montrer que si l'arbre initial est un arbre binaire de recherche, l'arbre obtenu après une rotation est toujours un arbre binaire de recherche.
- b) Rédiger un programme C++ réalisant la rotation droite. Vous utiliserez l'interface suivante :

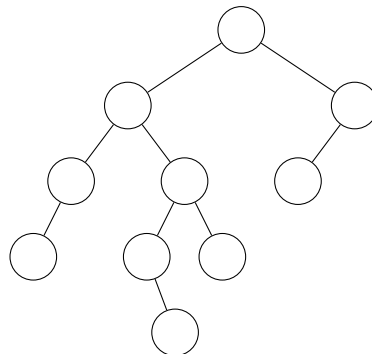
```
struct Noeud {
    int valeur ;
    Noeud* droite ;
    Noeud* gauche ;
} ;

//la valeur de retour est la racine du sous-arbre apres rotation
Noeud* rotation_droite(Noeud* n) ;
```

### 2 Hauteur d'un arbre

Les AVL sont des arbres binaires de recherche qui utilisent des rotations pour se rééquilibrer au fur et à mesure des insertions. Pour déterminer le déséquilibre des nœuds, les AVL utilisent la notion de *hauteur d'un sous arbre*. Étant donné un nœud de l'arbre, sa hauteur est le nombre de nœuds (extrémités comprises) sur le plus long chemin permettant d'atteindre une feuille depuis le nœud.

- c) Sur l'arbre suivant indiquez dans chaque nœud sa hauteur.

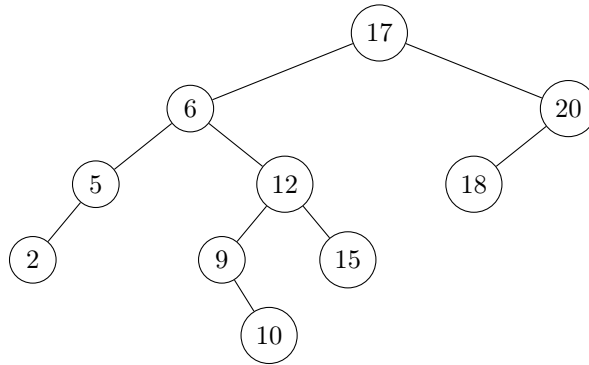


d) Quelle formule permet d'obtenir la hauteur d'un nœud en fonction des hauteurs de ses deux enfants ?

### 3 Hauteur et insertion

Au fur et à mesure des insertions, les AVL recalculent les hauteurs des nœuds de l'arbre. Le rééquilibrage est réalisé par rotation en fonction du déséquilibre entre les hauteurs des enfants. Lorsque la différence entre les hauteurs des enfants atteint 2 ou -2, un rééquilibrage est déclenché sur le nœud.

e) Sur l'arbre suivant, indiquez pour chaque nœud son déséquilibre : la différence entre la hauteur de l'enfant gauche et celle de l'enfant droit. Les hauteurs sont les mêmes que pour l'arbre précédent.



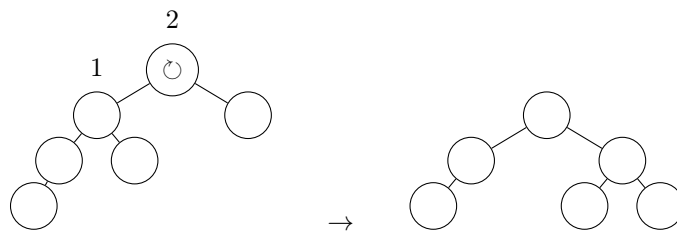
f) Dans l'arbre précédent, insérer successivement puis 13 puis 3 puis 14 puis 4 et recalculer les hauteurs. En déduire les nœuds dont la hauteur est susceptible d'être mise à jour lors d'une insertion.

g) Selon le nombre de nœuds de l'arbre, est-il toujours possible d'avoir un déséquilibre nul sur tous les nœuds ?

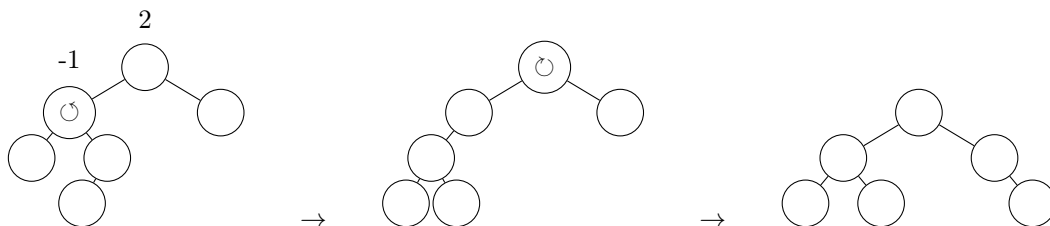
### 4 Application des rotations

Dans les AVL, après chaque insertion, en remontant depuis la feuille insérée, les déséquilibres sont vérifiés. Dès qu'un nœud a un déséquilibre trop grand (2 ou -2), un ensemble de rotations est appliqué pour corriger ce déséquilibre. On peut montrer que lorsqu'un nœud est trop déséquilibré, il n'y a que quatre cas possibles.

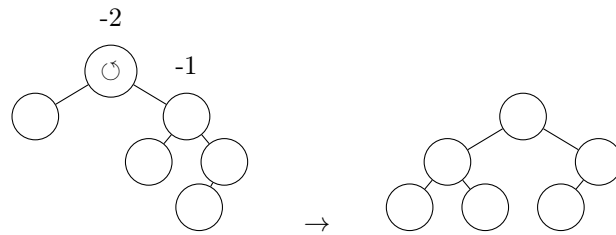
**déséquilibre 2 et gauche de déséquilibre 1 :** rotation droite sur le nœud.



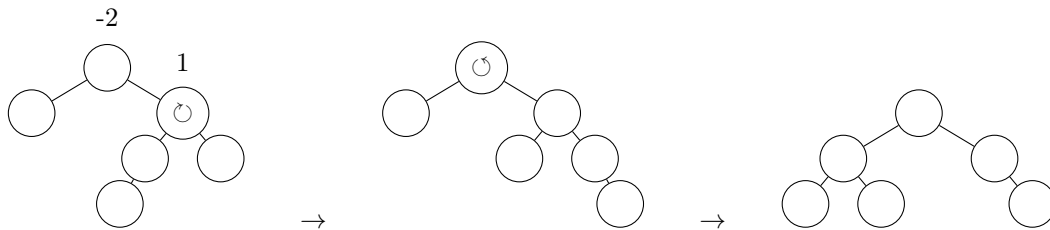
**déséquilibre 2 et gauche de déséquilibre -1 :** rotation gauche à gauche puis rotation droite sur le nœud.



déséquilibre -2 et droit de déséquilibre -1 : rotation gauche sur le nœud.



déséquilibre -2 et droit de déséquilibre 1 : rotation droite à droite puis rotation gauche sur le nœud.



**h)** À partir d'un arbre vide, insérez successivement les valeurs 13, 7, 9, 10, 12, 15, 1, 4, 8 dans cet ordre, et après chaque insertion vérifiez les déséquilibres et rééquilibrez l'arbre selon les cas énoncés ci-dessus.