Chapitre 7

Arbres AVL





Arbres AVL

- *Avantages et inconvénients d'un ABR*:

  Très efficace pour les insertions et suppressions mais s'ils sont complètement déséquilibrés ils transforment la recherche en algorithme O(n) (cas des arbres dégénérés)

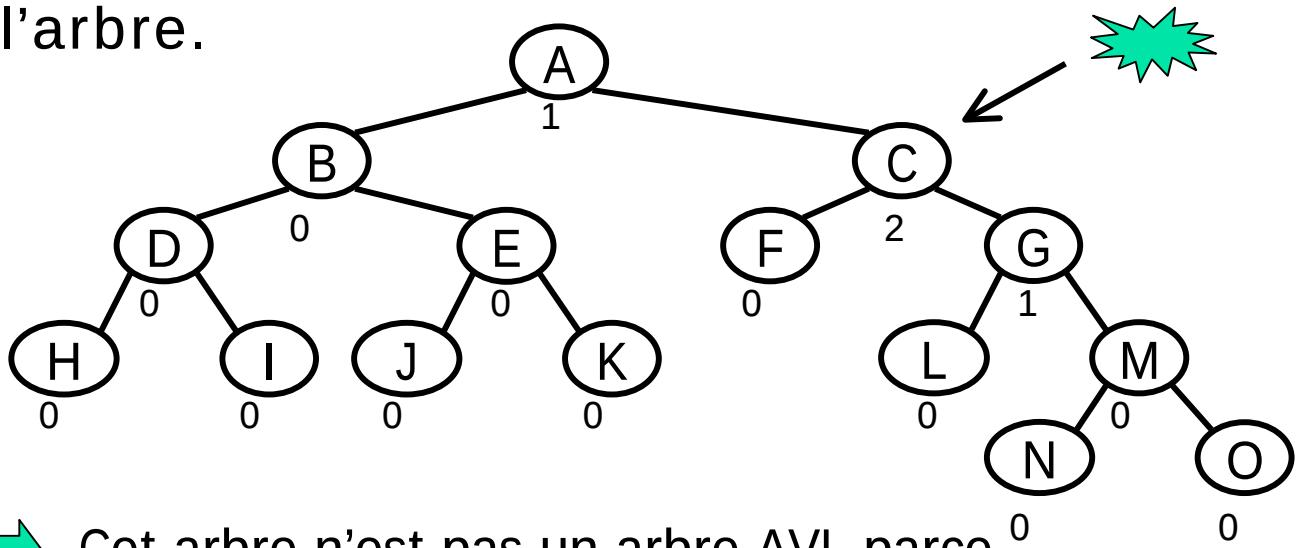
- *Solution* : Éviter les variations dans les longueurs des branches de l'arbre :

  - **Première idée** : arbres complètement équilibrés : très coûteux pour les insertions et les suppressions

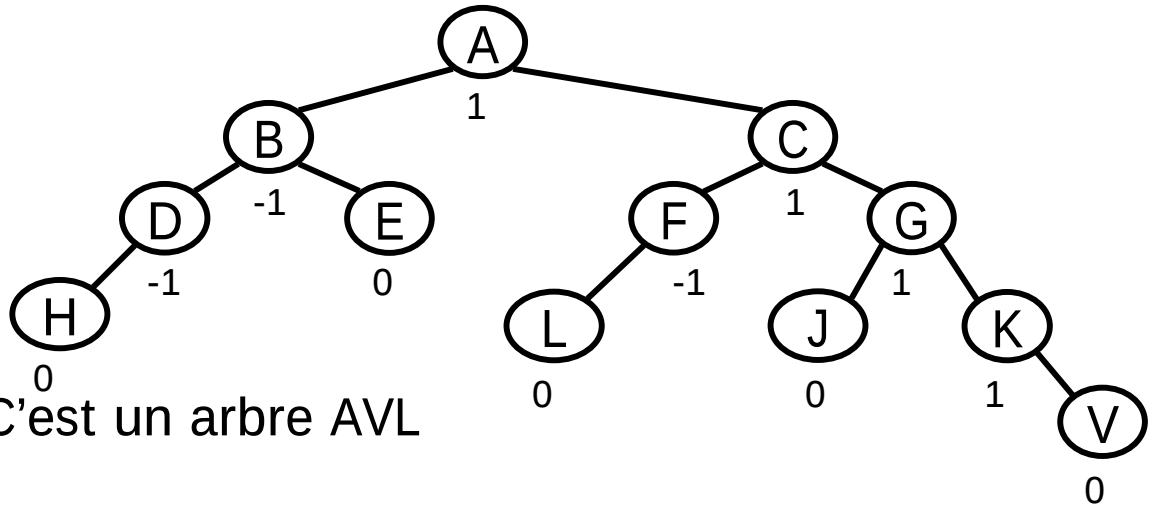
  - **Deuxième idée** : arbres AVL (Adelson-Velsskii-Landis) : AB équilibrés tel que pour chaque nœud les hauteurs de ses deux sous arbres gauche et droit diffèrent au plus de 1

Arbres AVL

- Les AVL garantissent la rapidité de l'exécution de l'algorithme de recherche cependant en doit effectuer des rotations complexes lors de l'insertion et la suppression afin de maintenir l'équilibre de l'arbre.



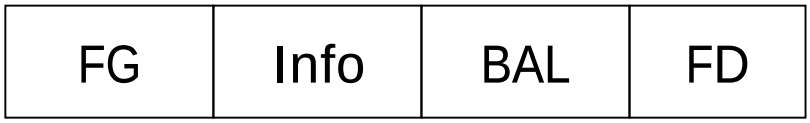
➡ Cet arbre n'est pas un arbre AVL parce qu'il est déséquilibré au niveau du nœud C



➡ C'est un arbre AVL

Arbres AVL

- Représentation chaînée :



- **BAL = H(SAD) - H(SAG).**
  Ce nombre doit être égal à –1, 0 ou 1 pour que l'arbre soit un arbre AVL.
  - Si **BAL = 0**: H(SAD) = H(SAG).
    → nœud équilibré balancé.
  - Si **Bal = 1**: H(SAD) > H(SAG).
  - Si **Bal = -1**: H(SAD) < H(SAG).

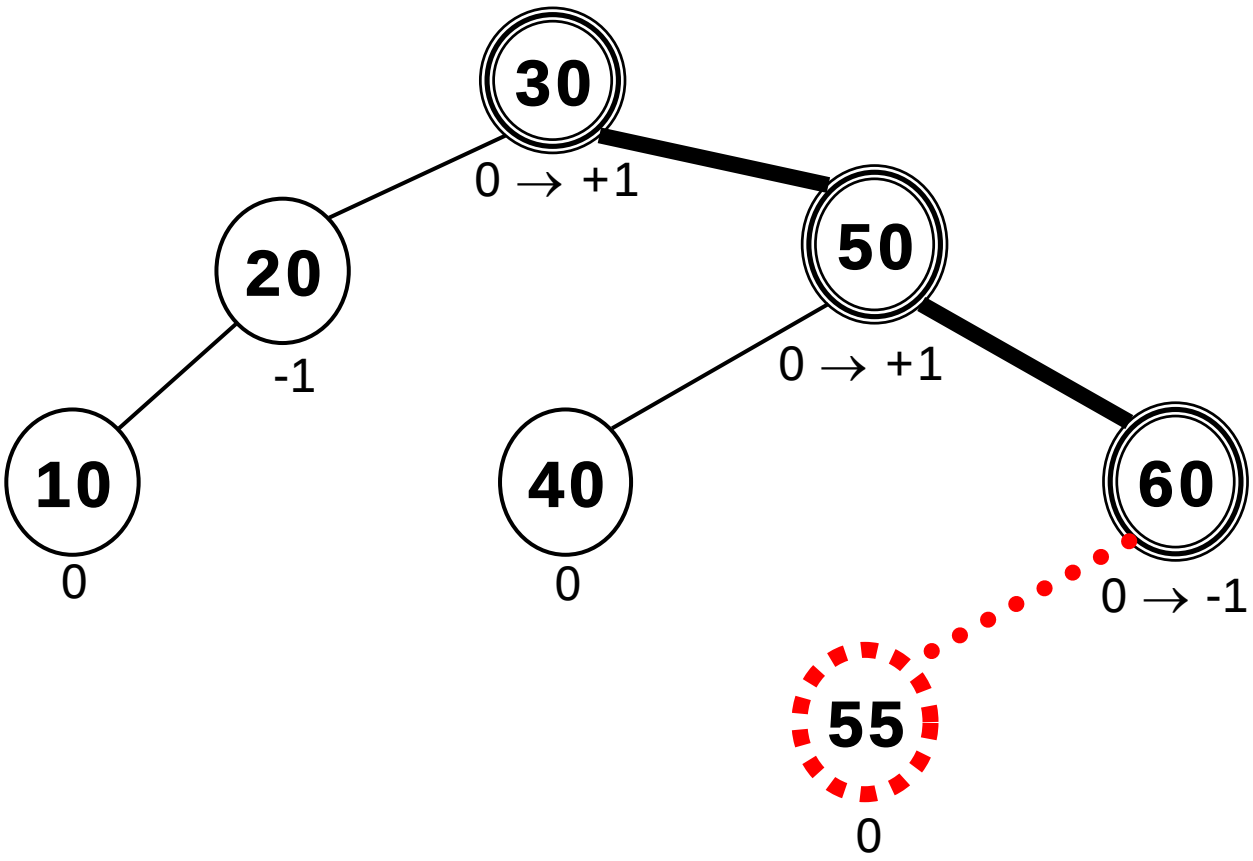
Arbre AVL : insertion (1)

- **Première étape** : ajouter le nouveau nœud dans l'arbre comme s'il s'agit d'une insertion dans **ABR**. Le chemin de la racine au nouveau nœud est appelé **chemin de recherche**

- **Deuxième étape** : MAJ les champs BAL de tous les nœuds du **chemin de recherche**
- **Troisième étape** : Vérifier si l'arbre est AVL. Si l'arbre est déséquilibré → le rendre AVL.

Arbre AVL : insertion (2)

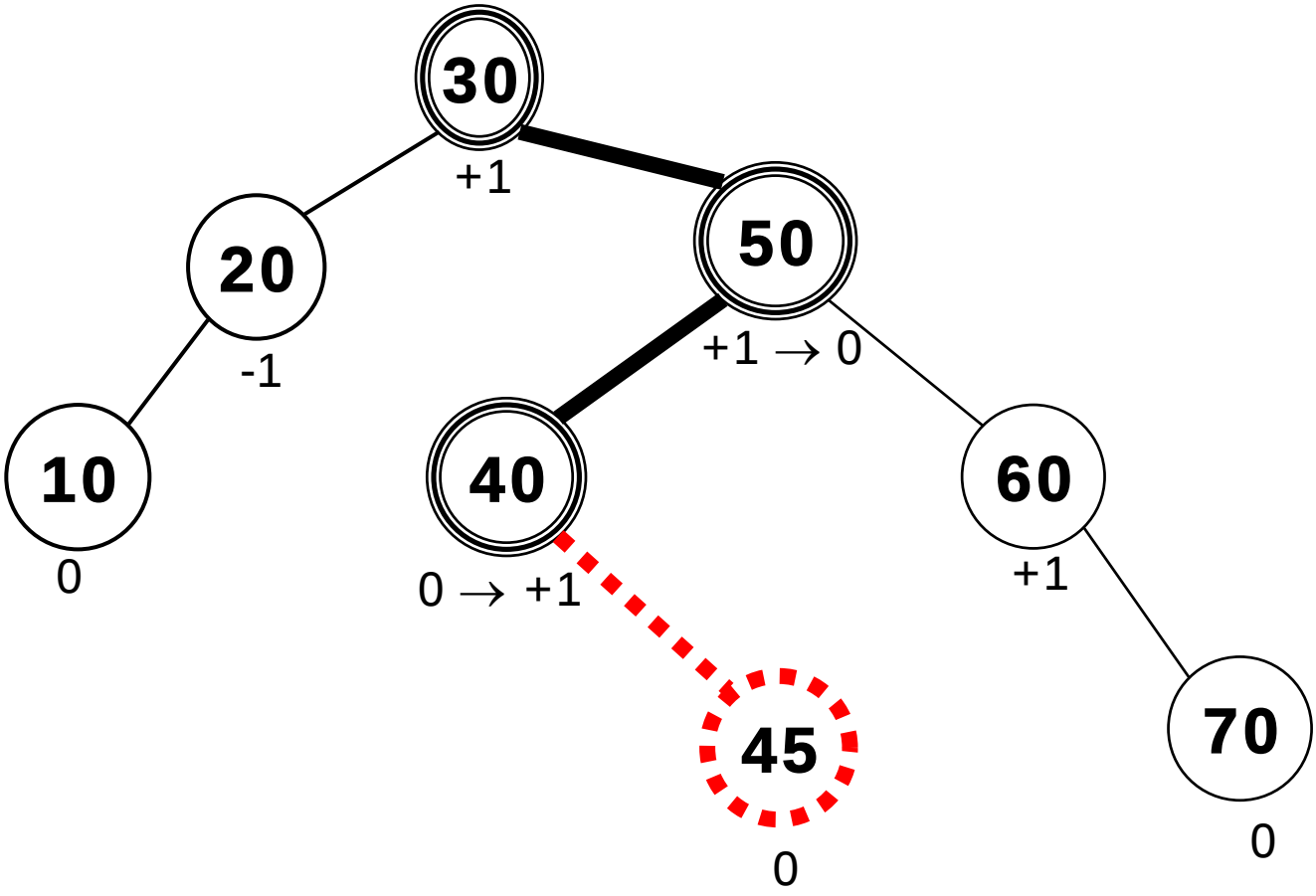
- Exemple 1: insérer(55)



L'arbre résultat est AVL

Arbre AVL : insertion (3)

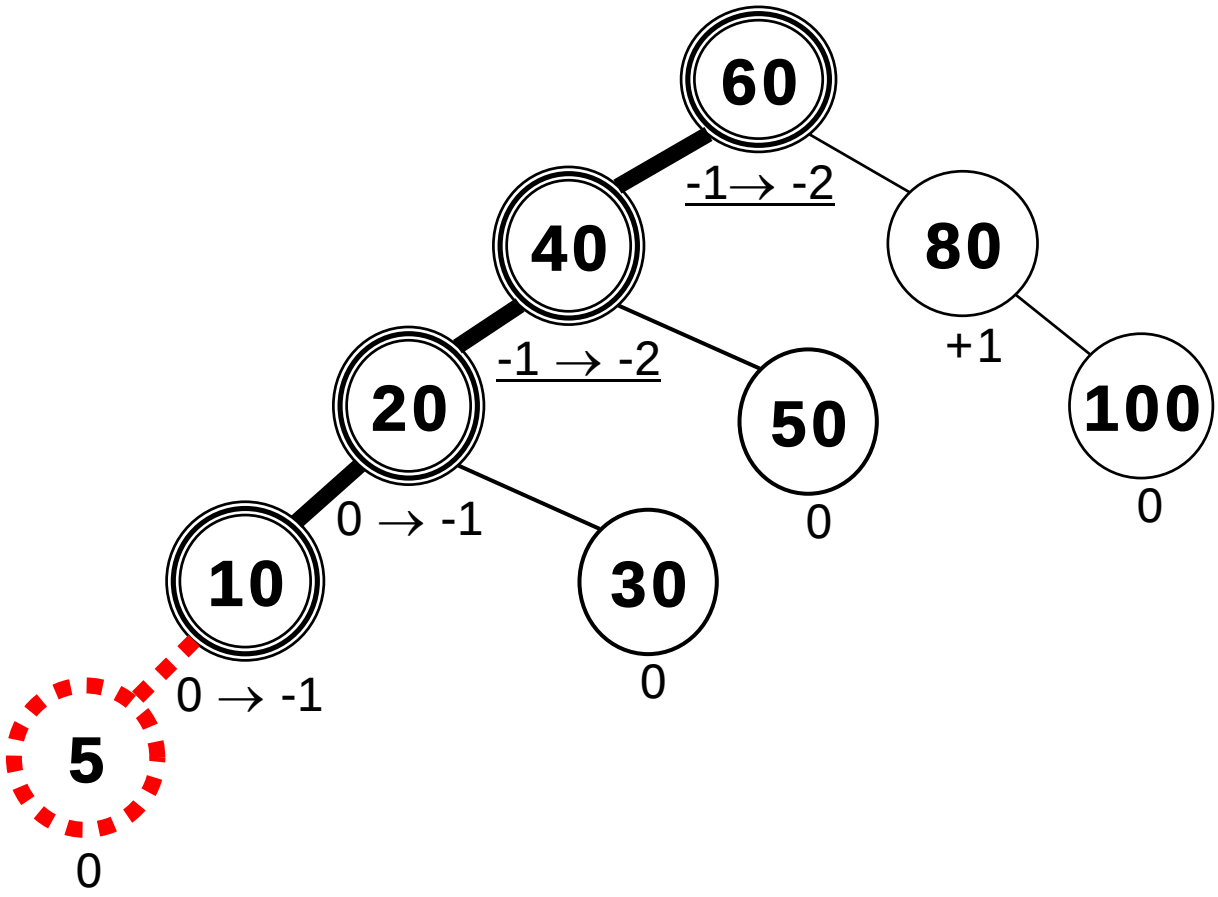
- Exemple 2: insérer(45)



L'arbre résultat est AVL

Arbre AVL : insertion (4)

- Exemple 3 : insérer(5)



Ce n'est pas un arbre AVL
→ transformation

Arbre AVL : insertion (5)

Un arbre est déséquilibré si certains de ses sous arbres ont un déséquilibre supérieur à 1 en valeur absolue (-2 ou +2)

-a- Soit **A** le plus petit de ces sous arbres cad **A** est tel que ses SAG et SAD sont équilibrés

-b- Transformer **A** pour le rendre équilibré

-c- En rendant **A** équilibré cela ne veut pas dire que l'arbre complet est devenue équilibré dans ce cas il faut recommencer la transformation (étape (a))

Arbre AVL : insertion (6)

Cas 1:

BAL(racine(A)) = -2:

H(SAG(A)) > H(SAD(A))

→ Les opérations à pratiquer dépendent de la valeur de la balance du **FG de A** qui peut être –1, +1 ou 0 (puisque le SAG de A est équilibré)

Cas 2:

BAL(racine(A)) = +2:

H(SAD(A)) > H(SAG(A))

→ Les opérations à pratiquer dépendent de la valeur de la balance du **FD de A** qui peut être –1, +1 ou 0 (puisque le SAD de A est équilibré)



Arbre AVL: différents cas

SI: BAL(racine(A))=-2 et BAL(FG(A))=-1

SI: BAL(racine(A))=-2 et BAL(FG(A))=0

SI: BAL(racine(A))=-2 et BAL(FG(A))=+1

SI: BAL(racine(A))=+2 et BAL(FD(A))=+1

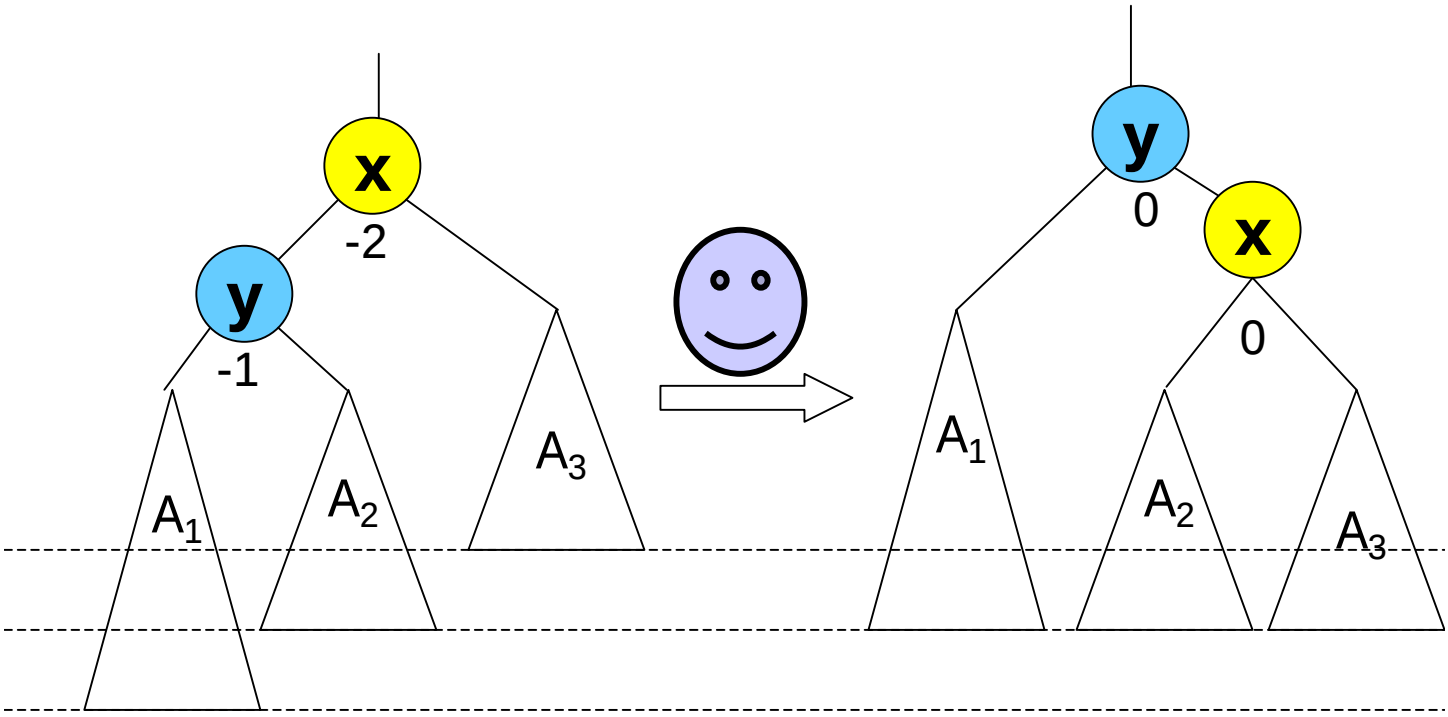
SI: BAL(racine(A))=+2 et BAL(FD(A))=0

SI: BAL(racine(A))=+2 et BAL(FD(A))=-1

Arbre AVL : insertion (7)

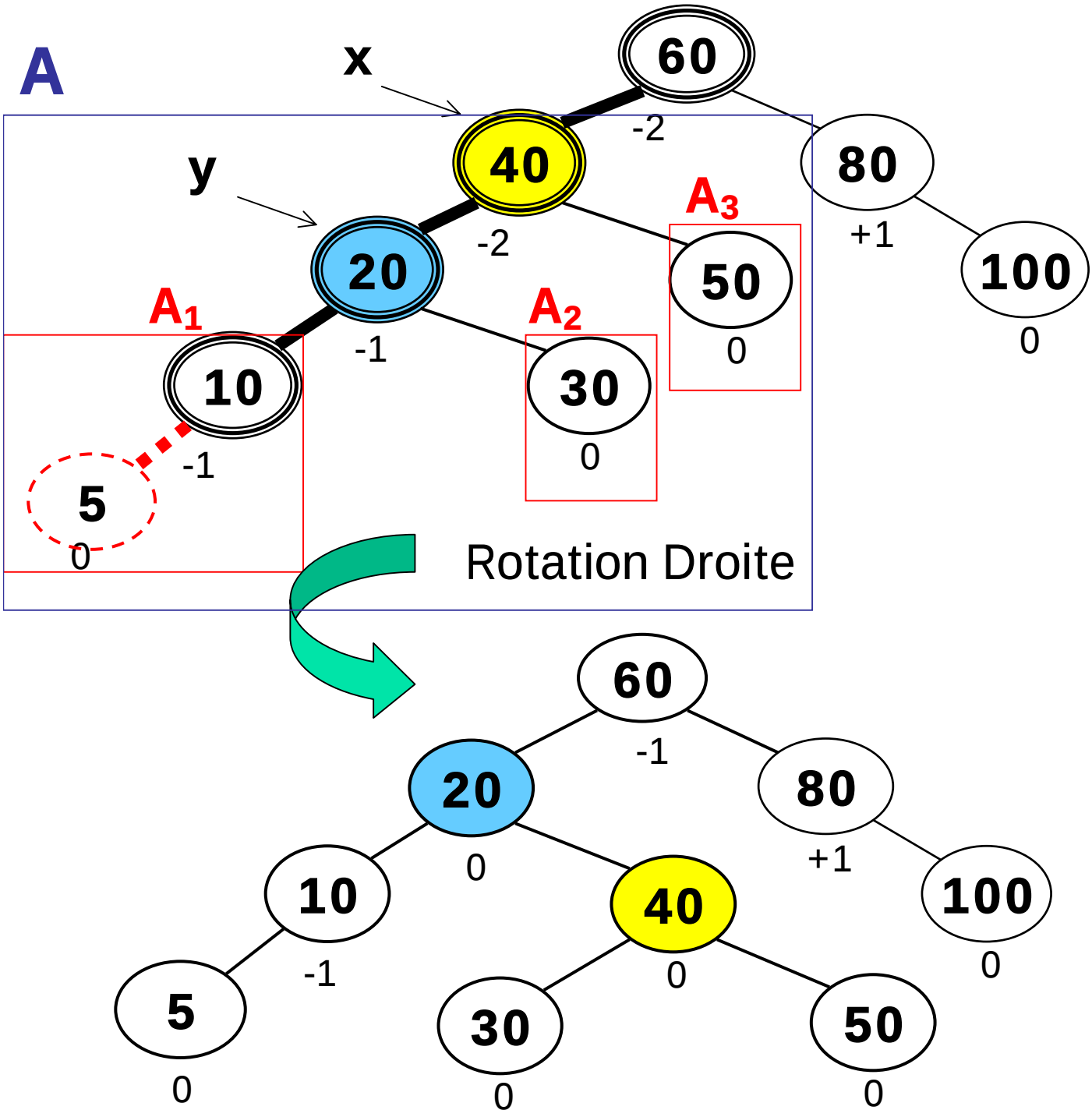
- **SI**: BAL(Ref_Racine(A))=-2 et BAL(FG(A))=-1
- **Solution** : Rotation droite

Le sous arbre A:



Arbre AVL : insertion (8)

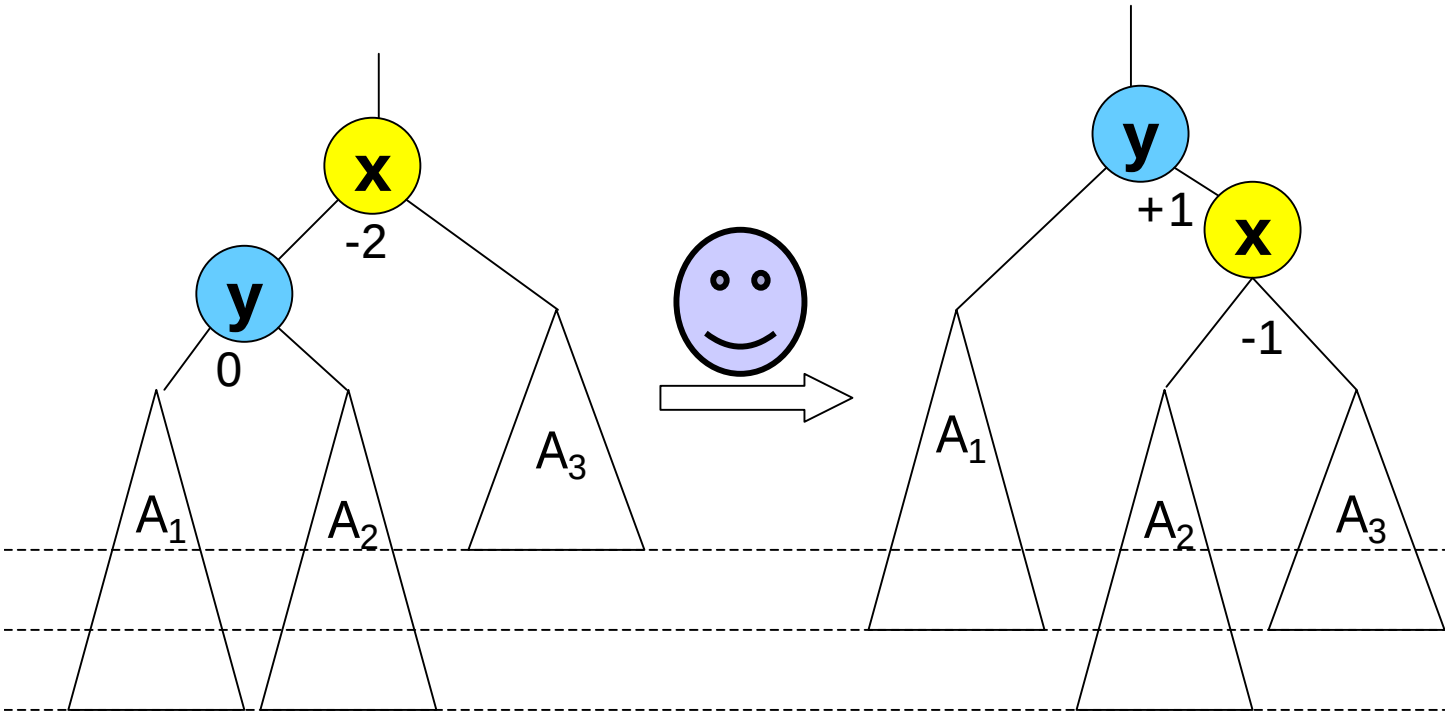
- **Illustration : insérer (5)**



Arbre AVL : insertion (9)

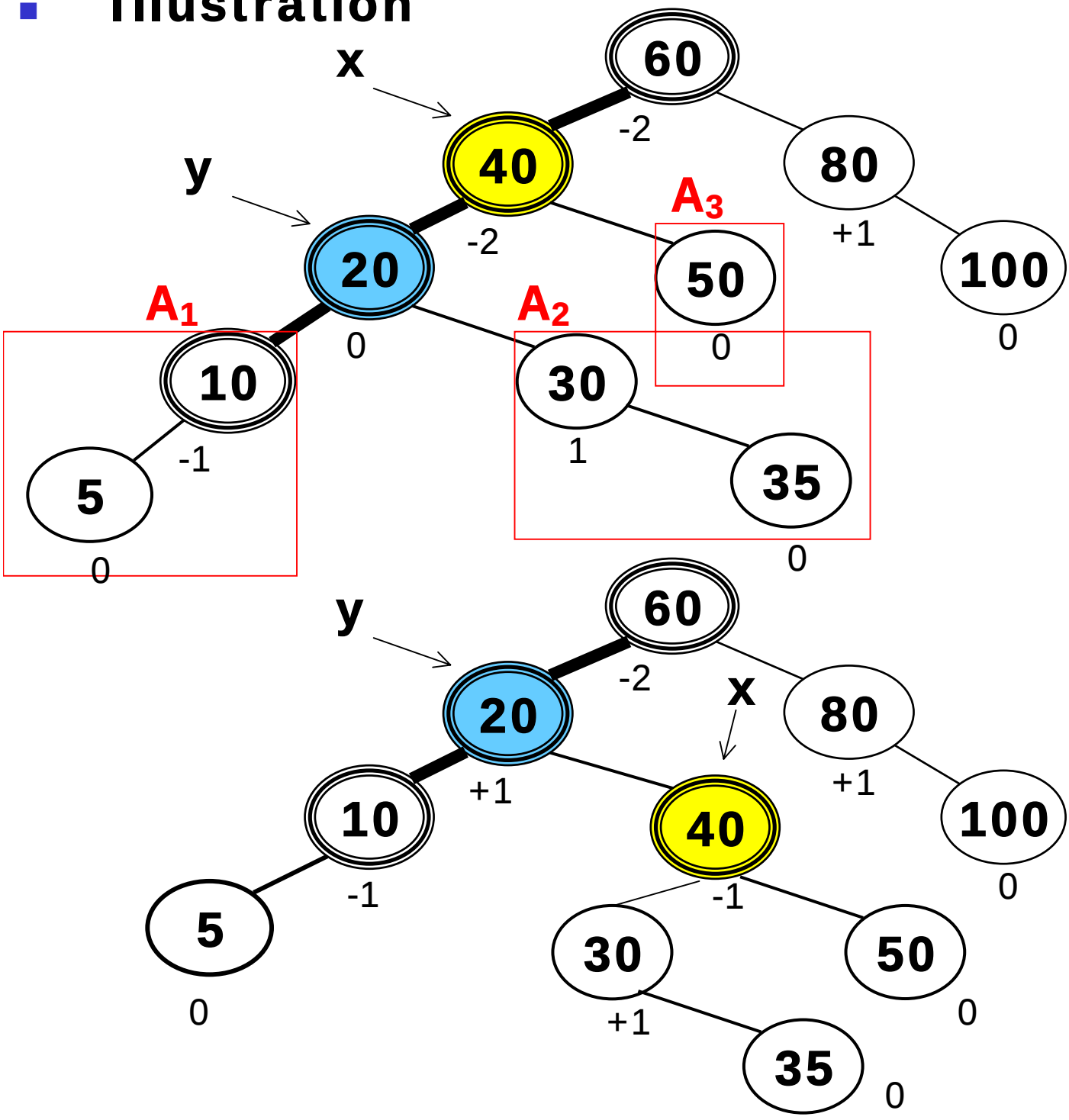
- SI: BAL(Ref_Racine(A))=-2 et BAL(FG(A))=0
- Solution : Rotation droite

Le sous arbre A:



Arbre AVL : insertion (9)

- **Illustration**



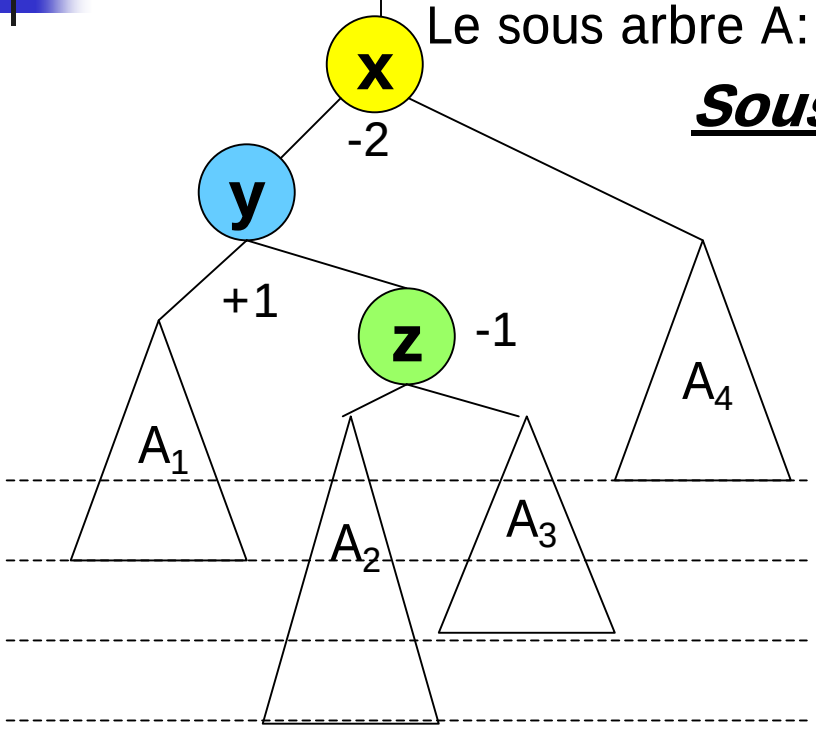


Arbre AVL: insertion (11)

- **SI**: BAL(Ref_Racine(A))=-2 et BAL(FG(A))=+1
- Dans ce cas il y a 3 sous cas possibles du FD du FG(A):

Arbre AVL: insertion (12)

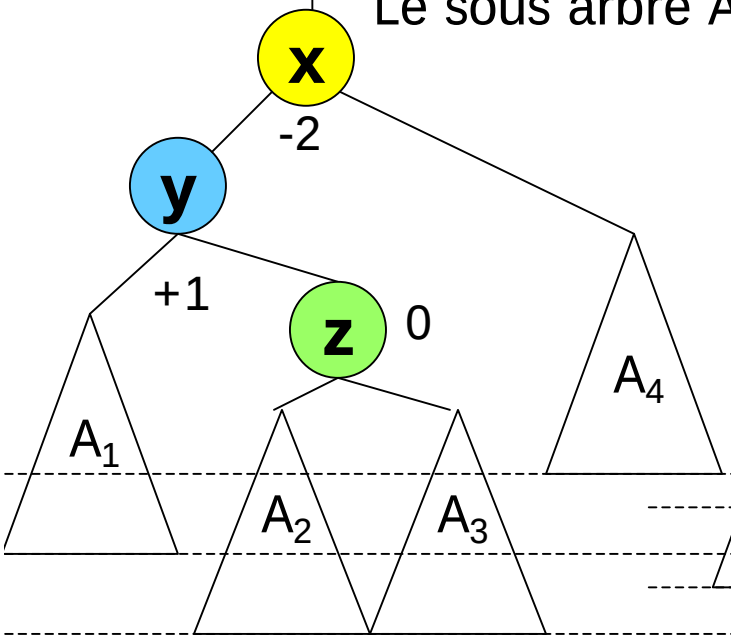
Le sous arbre A:



Sous-cas 1

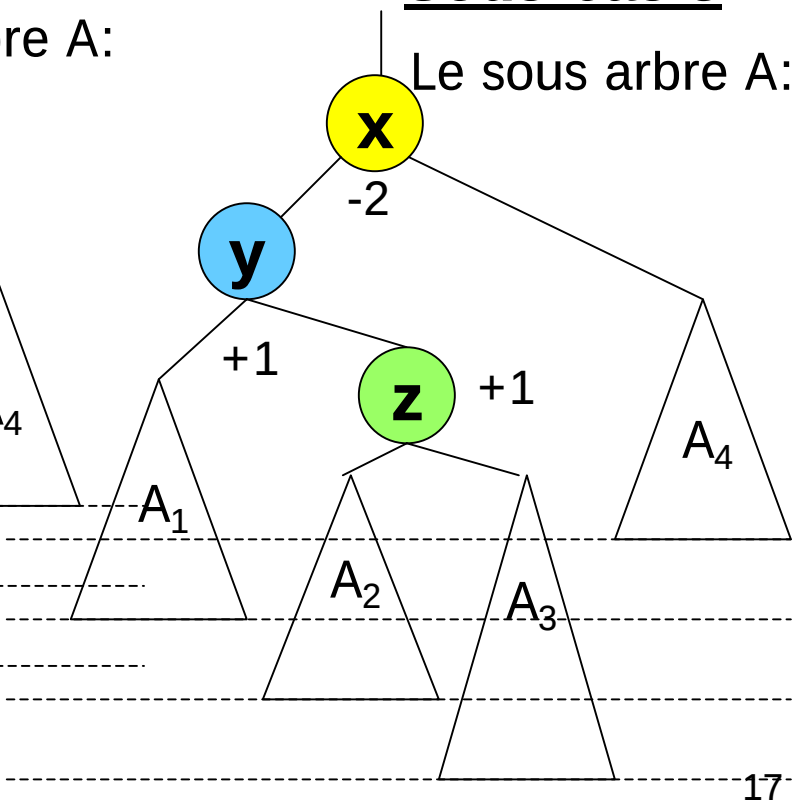
Sous-cas 2

Le sous arbre A:



Sous-cas 3

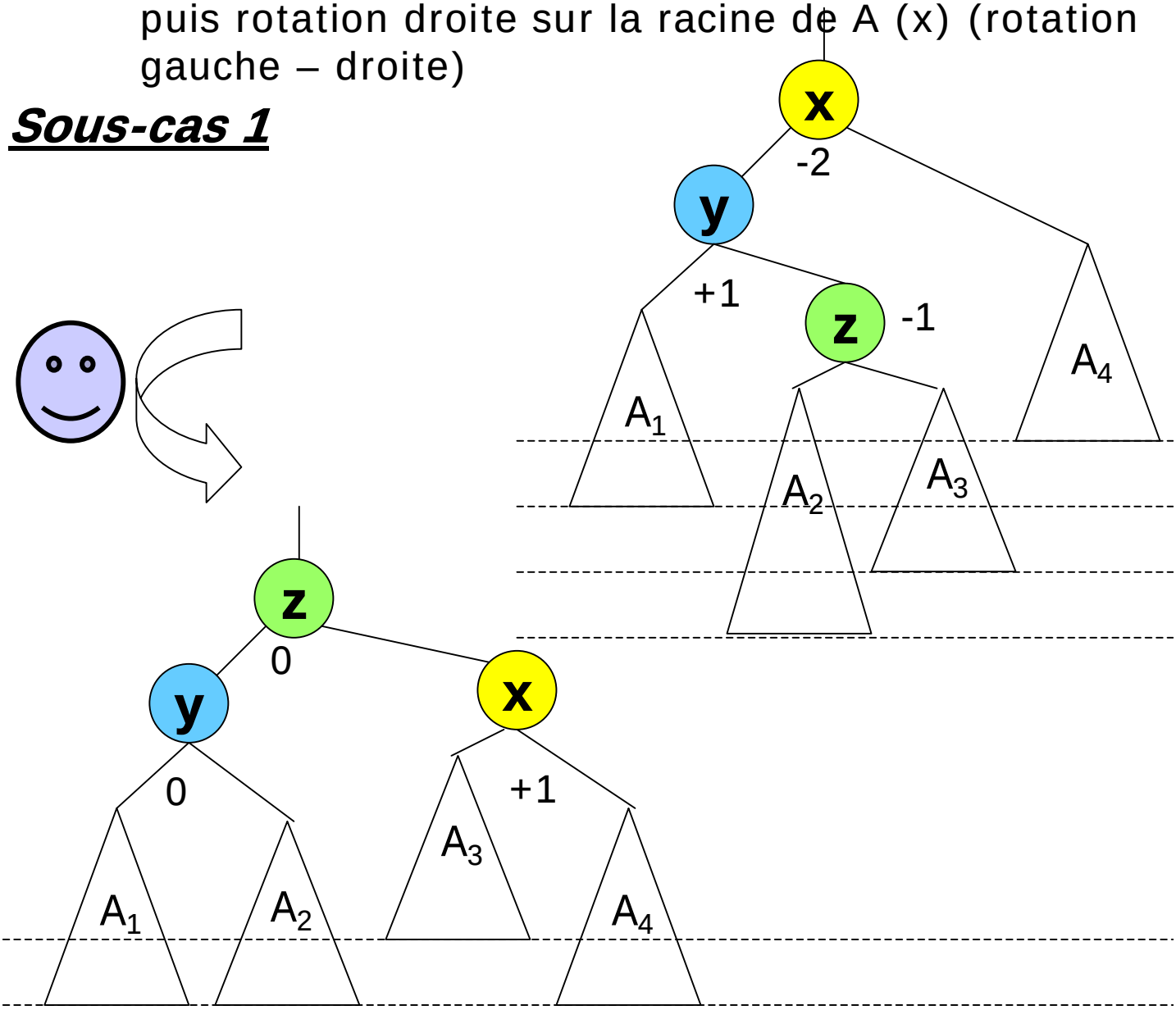
Le sous arbre A:



Arbre AVL: insertion (13)

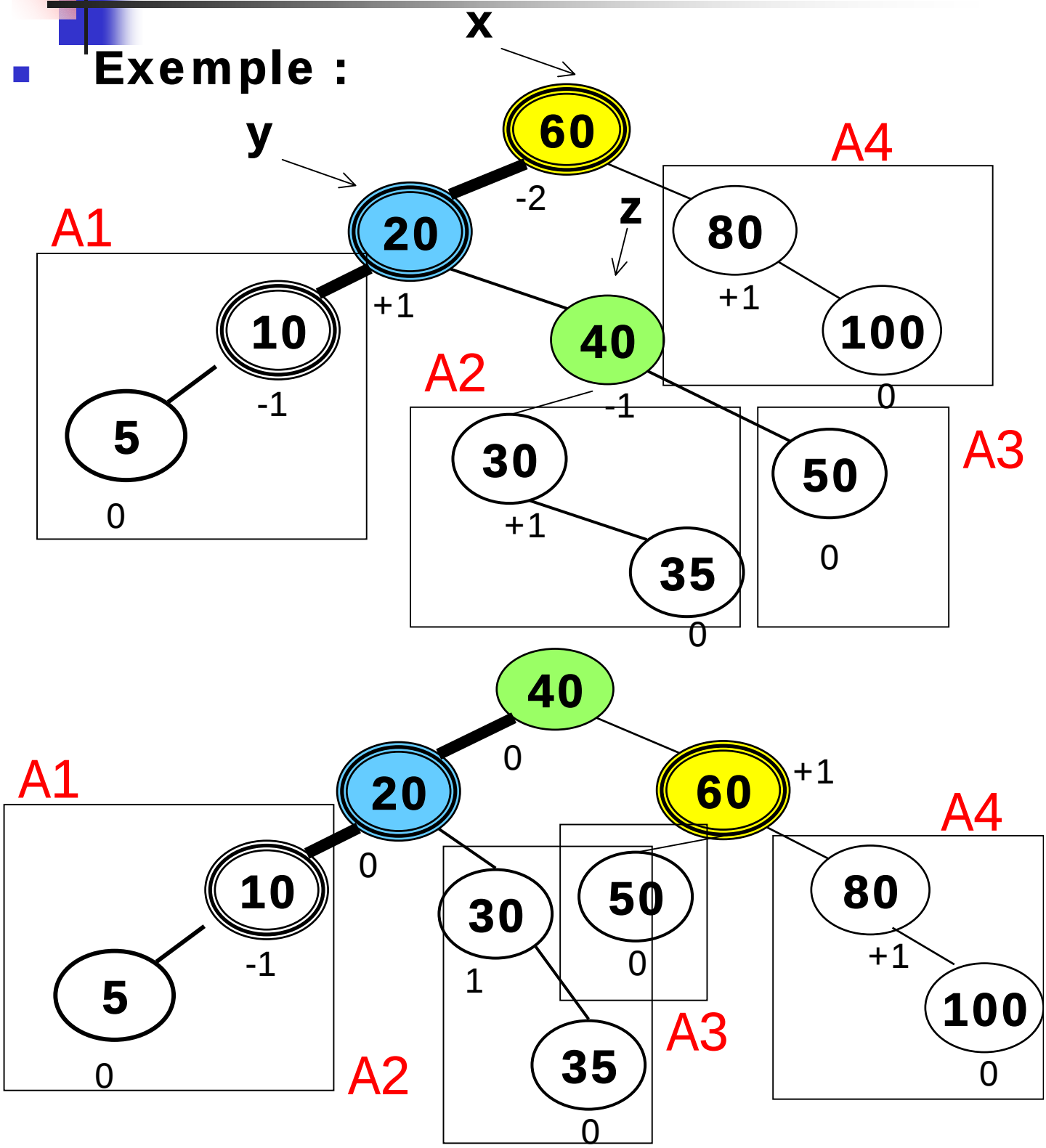
- **SI**: BAL(Ref_Racine(A))=-2 et BAL(FG(A))=+1 et BAL(FD(FG(A)))=-1
- **Solution** : Rotation gauche du fils gauche de A (y) puis rotation droite sur la racine de A (x) (rotation gauche – droite)

Sous-cas 1



Arbre AVL : insertion (14)

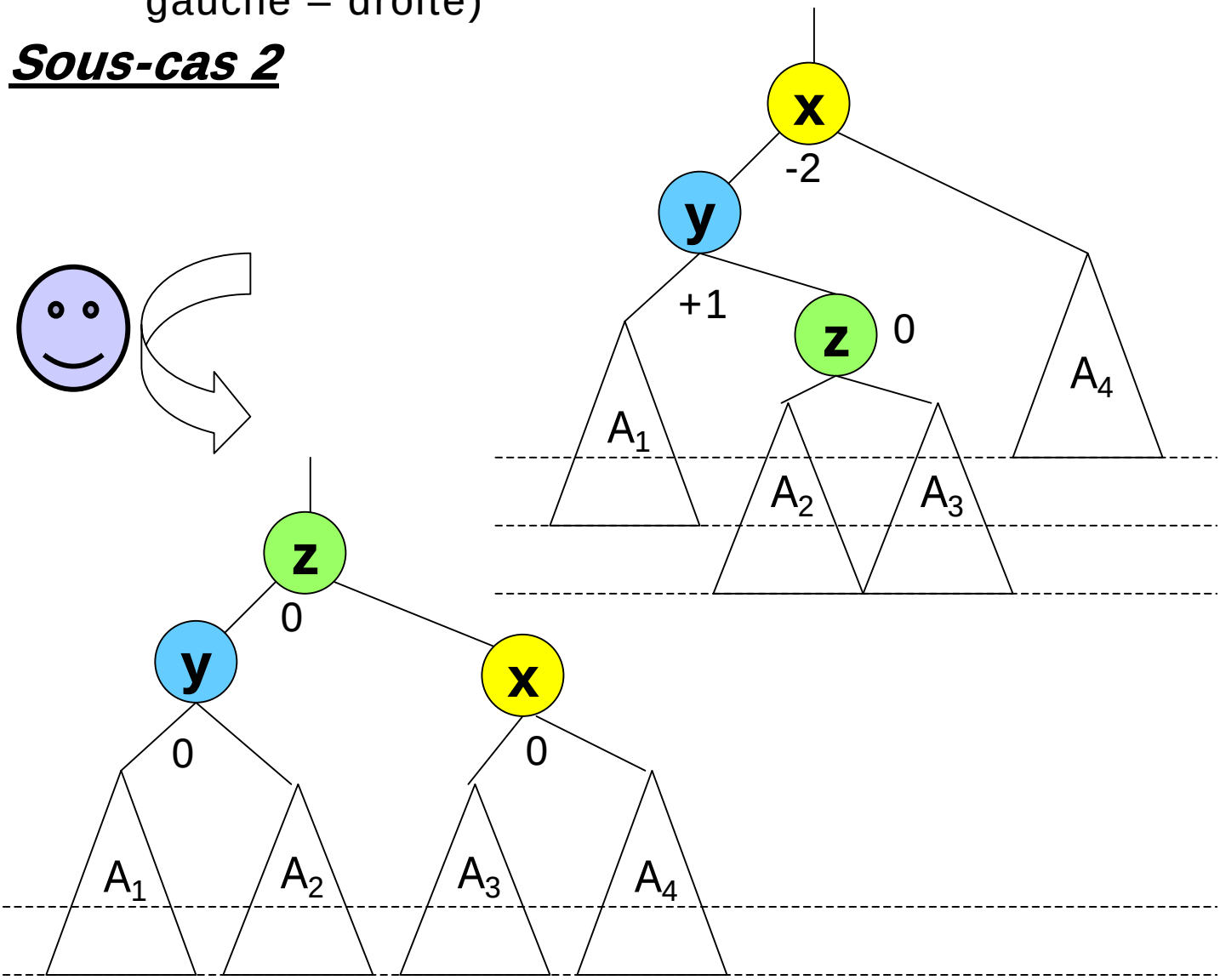
- **Exemple :**



Arbre AVL: insertion (15)

- **SI**: BAL(Ref_Racine(A))=-2 et BAL(FG(A))=+1 et BAL(FD(FG(A)))=0
- **Solution** : Rotation gauche du fils gauche de A (y) puis rotation droite sur la racine de A (x) (rotation gauche – droite)

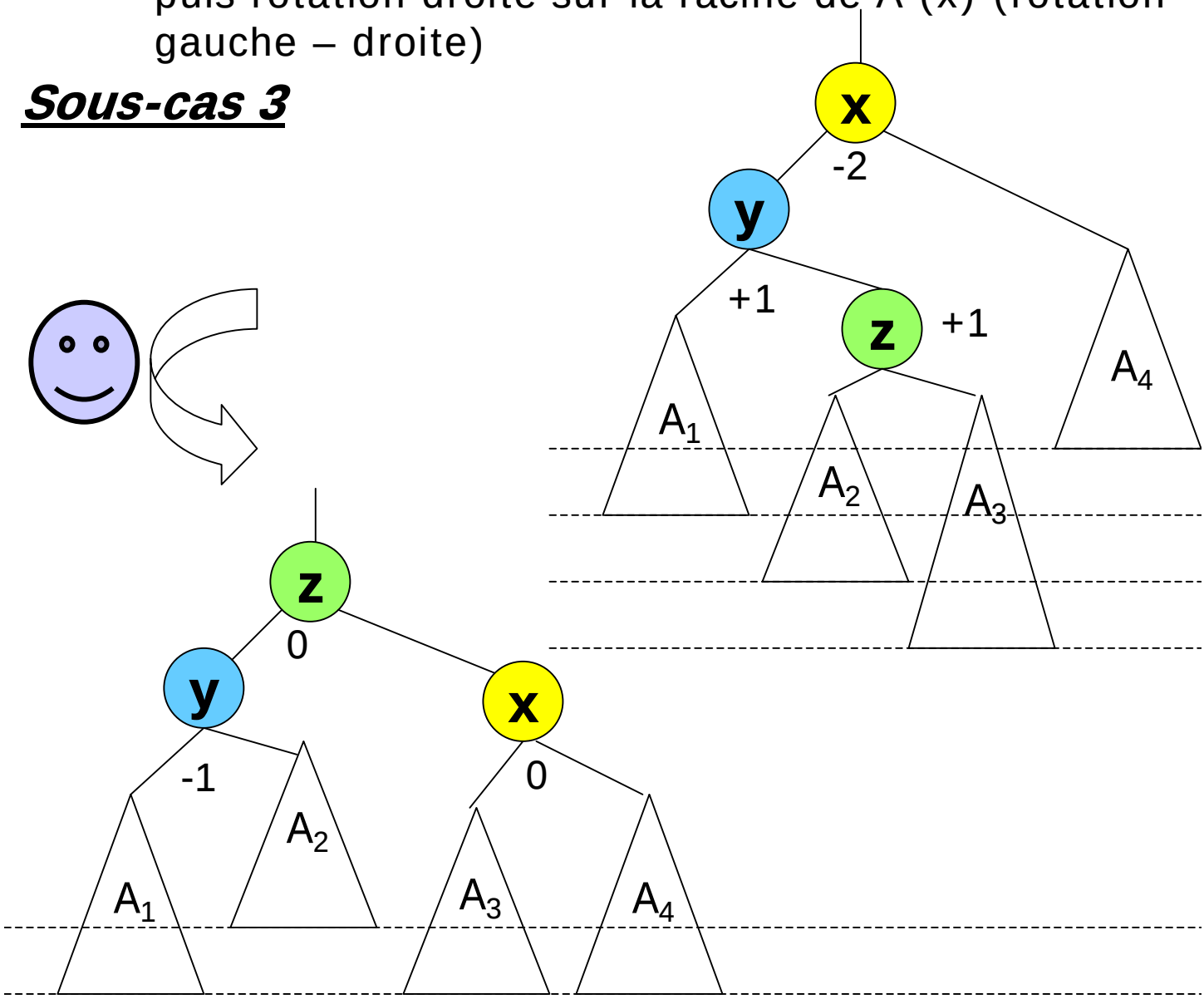
Sous-cas 2



Arbre AVL: insertion (14)

- **SI**: BAL(racine(A))=-2 et BAL(FG(A))=+1 et BAL(FD(FG(A)))=+1
- **Solution** : Rotation gauche du fils gauche de A (y) puis rotation droite sur la racine de A (x) (rotation gauche – droite)

Sous-cas 3



Arbre AVL: insertion (15)

- **SI**: BAL(racine(A))=-2 et BAL(FG(A))=-1

Solution : Rotation droite

- **SI**: BAL(racine(A))=-2 et BAL(FG(A))=0

Solution : Rotation droite

- **SI**: BAL(racine(A))=-2 et BAL(FG(A))=+1

Solution : Rotation gauche du fils gauche de A (y) puis rotation droite sur la racine de A (x) (rotation gauche – droite)

- **SI**: BAL(racine(A))=+2 et BAL(FD(A))=+1

Solution : Rotation gauche

- **SI**: BAL(racine(A))=+2 et BAL(FD(A))=0

Solution : Rotation gauche

- **SI**: BAL(racine(A))=+2 et BAL(FD(A))=-1

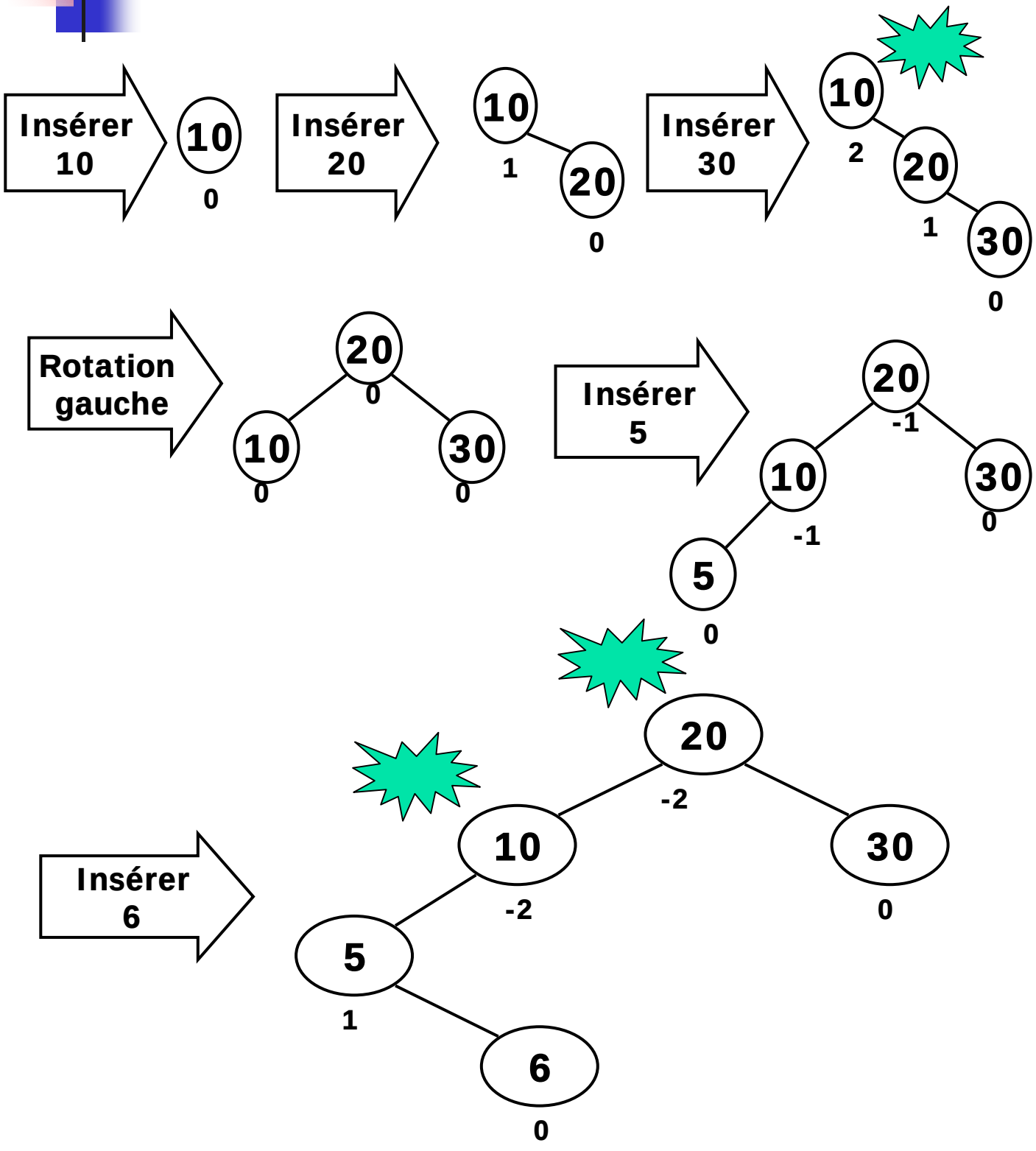
Solution : Rotation droite du fils droit de A (y) puis rotation gauche sur la racine de A (x) (rotation droite – gauche)

Travail à faire

- **Soit la séquence suivante: 10, 20, 30, 5, 6, 25, 45, 35, 50 et 47**

→ Créer l'arbre AVL correspondant

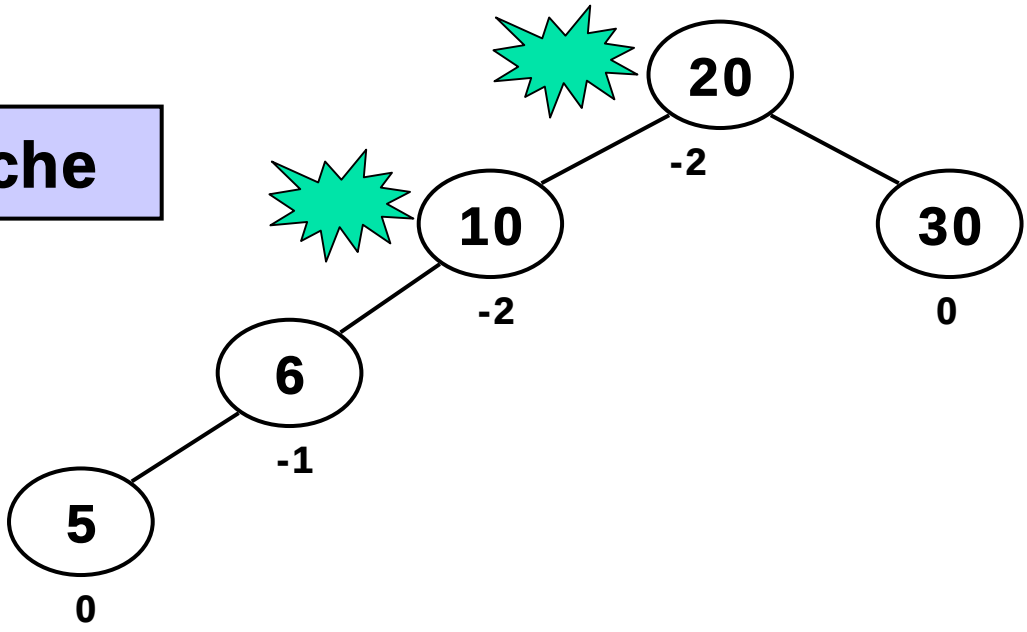
Création d'un arbre AVL



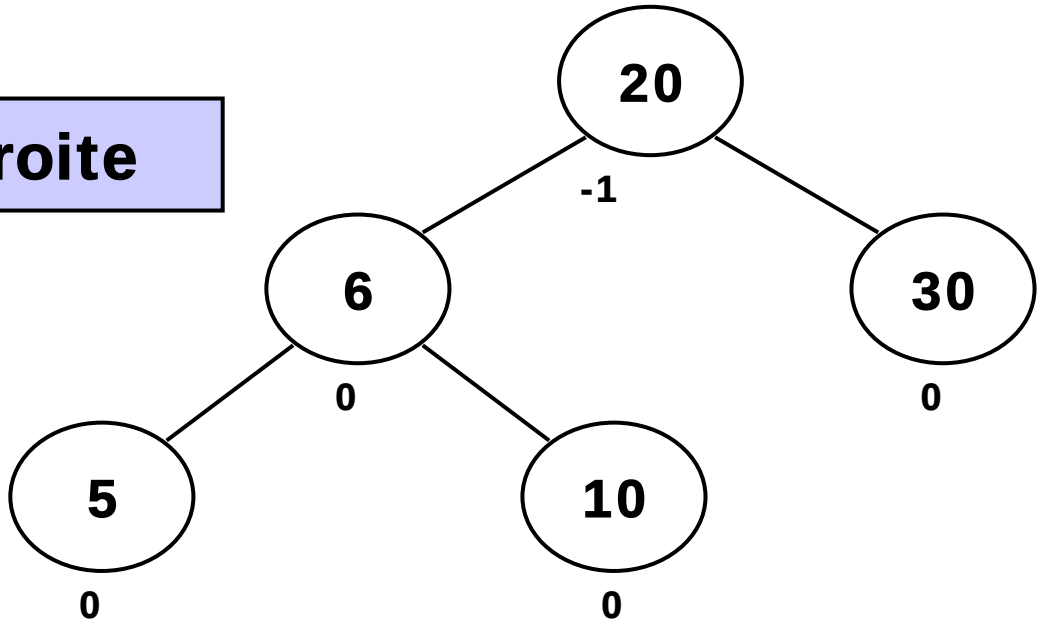
Création d'un arbre AVL

Rotation gauche-droite

gauche

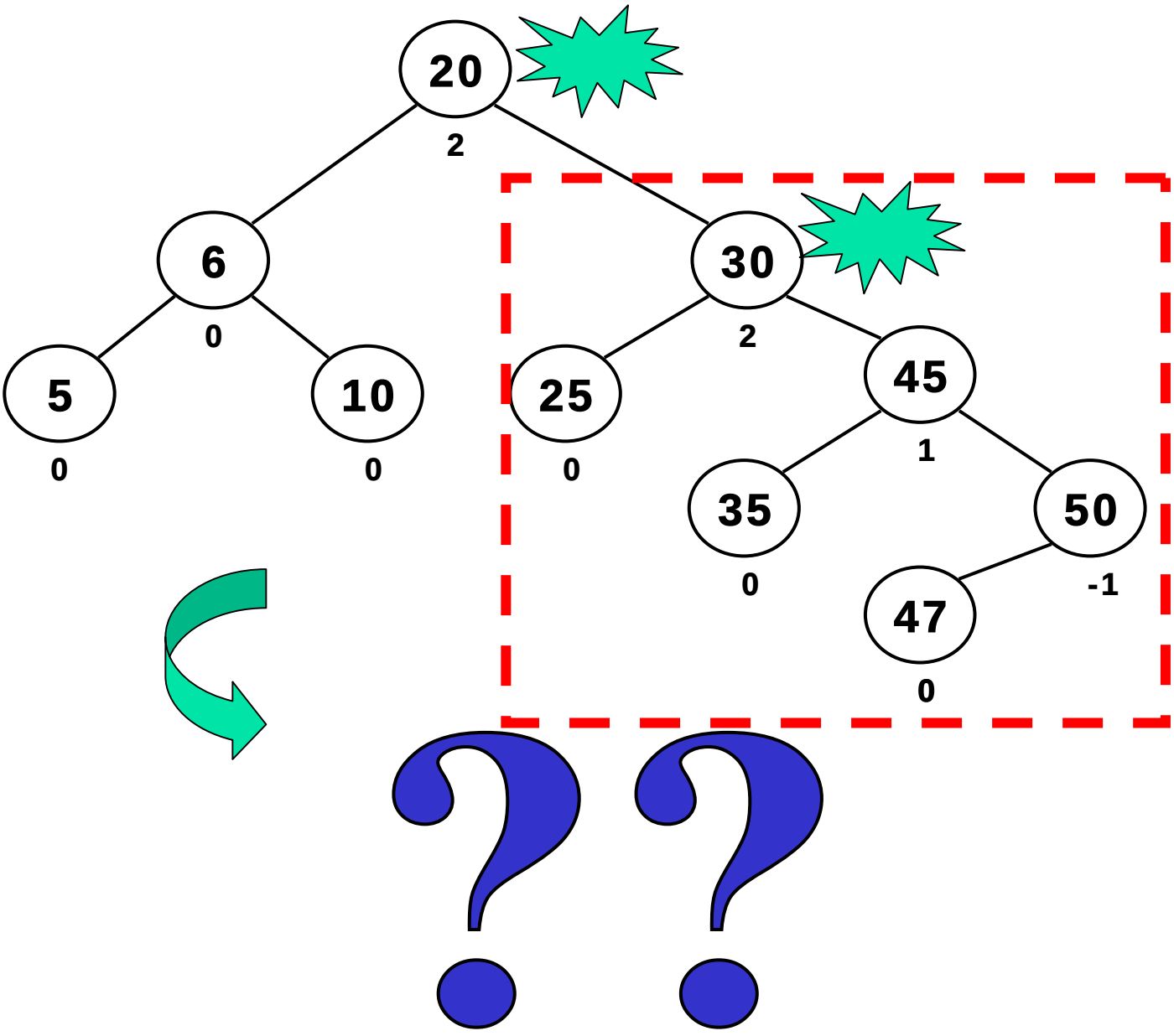


droite



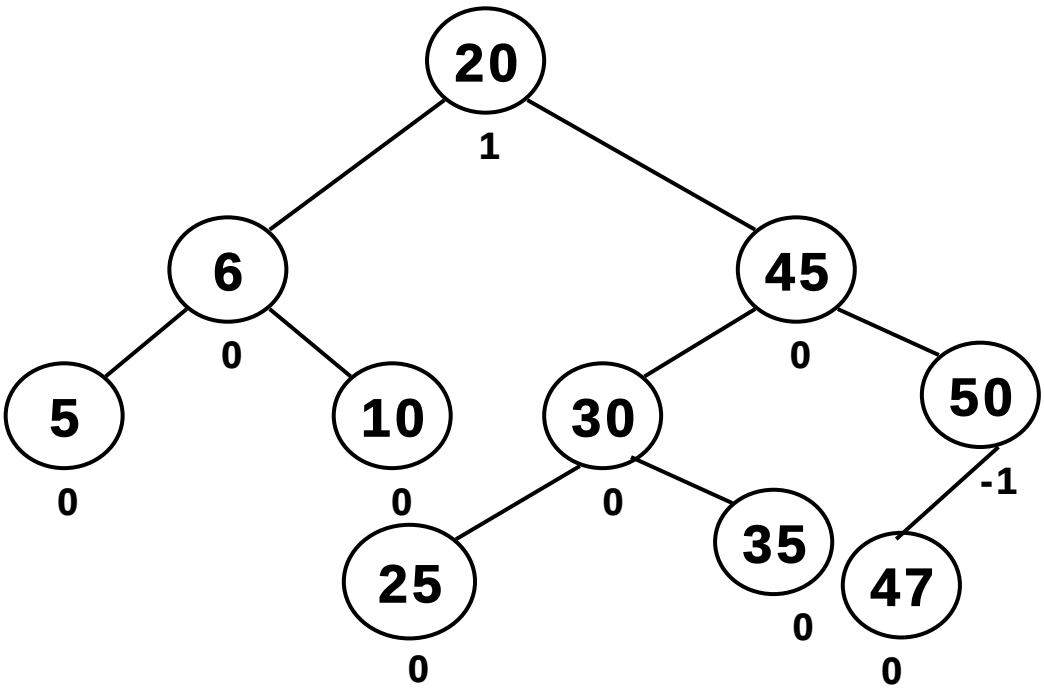
Création d'un arbre AVL

Insérer 25, 45, 35, 50 et 47



Création d'un arbre AVL

Rotation gauche



L'arbre résultat est AVL

Arbre AVL : Suppression

- **Première étape** : supprimer le nœud dans l'arbre comme s'il s'agit d'une suppression dans **ABR**. Le chemin de la racine au nouveau nœud est appelé **chemin de recherche**

- **Deuxième étape** : MAJ les champs BAL de tous les nœuds du **chemin de recherche**
- **Troisième étape** : Vérifier si l'arbre est AVL. Si l'arbre est déséquilibré → le rendre AVL (pareil que dans l'insertion).