

# Exercices ABR Corrigé

## Exercice 1 : ABR et POO

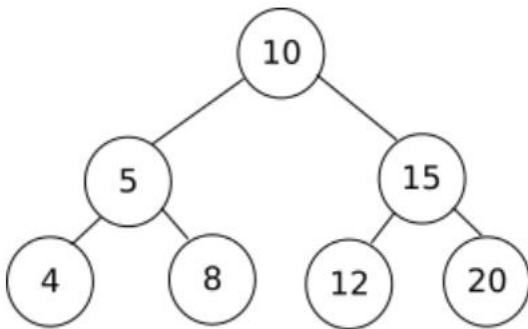
1a

taille de l'arbre = 7

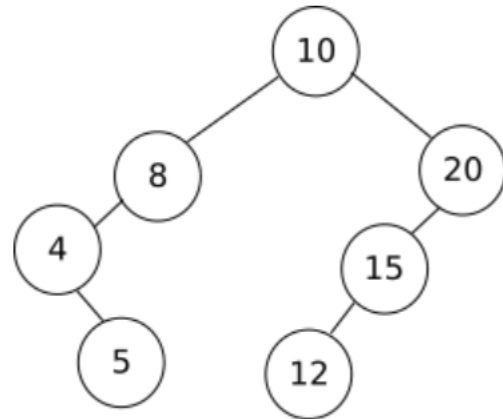
1b

hauteur de l'arbre = 4

2



3



4

```
def hauteur(self):  
    return self.racine.hauteur()
```

5

méthode taille de la classe Noeud :

```
def taille(self):  
    if self.gauche == None and self.droit == None :  
        return 1  
    if self.gauche == None :  
        return 1+self.droit.taille()  
    elif self.droit == None :  
        return 1+self.gauche.taille()  
    else :  
        return 1 + self.gauche.taille() + self.droit.taille()
```

méthode taille de la classe Arbre :

```
def taille(self):  
    return self.racine.taille()
```

6a

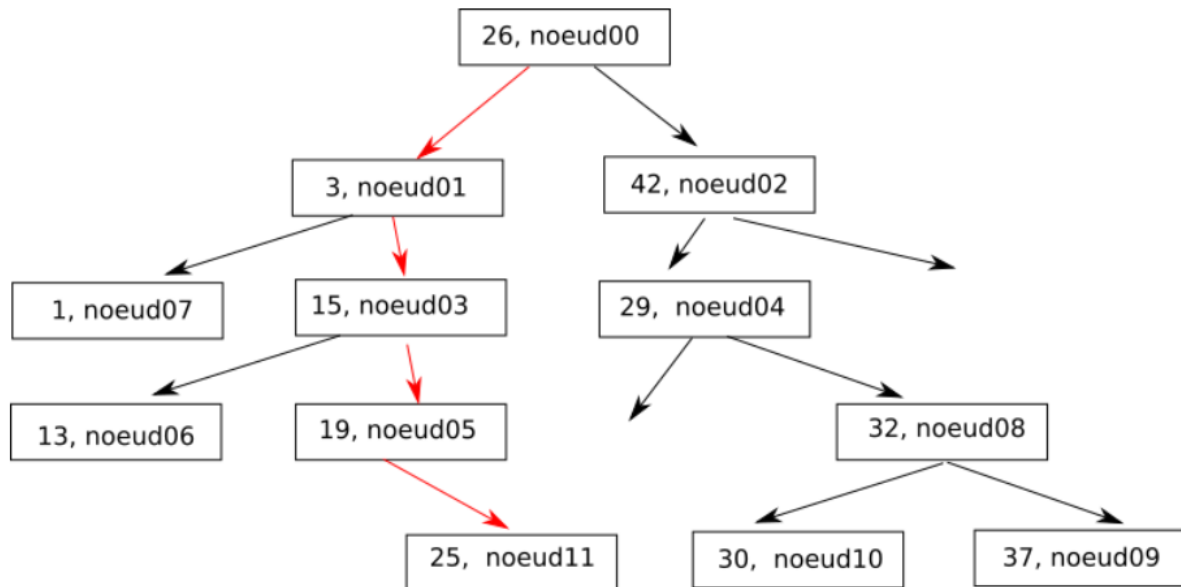
$t_{\min} = 2^{h-1}$

6b

```
def bien_construit(self):  
    t = self.taille()  
    h = self.hauteur()  
    return t >= 2**(h - 1)
```

## Exercice 2 : ABR

1



On désire insérer le noeud11 (valeur 25). On part de la racine (noeud00 de valeur 26), 25 est plus petit que 26, on considère donc le sous-arbre gauche et on se retrouve au niveau du noeud01 (valeur 3). 25 est plus grand que 3, on considère donc le sous-arbre droit au noeud01 et on se retrouve au niveau du noeud03 (valeur 15). 25 est plus grand que 15, on considère donc le sous-arbre droit au noeud03 et on se retrouve au niveau du noeud05 (valeur 19). 25 est plus grand que 19, on considère donc le sous-arbre droit du noeud05, ce sous-arbre droit est vide et on insère donc le noeud11 à cet emplacement. Le noeud11 est donc inséré sous le noeud5 en fils droit.

2

Il est possible de stocker toutes les valeurs comprises entre 26 et 29, c'est à dire : 26, 27 et 28 (on peut prendre 26 car il est précisé dans l'énoncé que "les valeurs de tous les nœuds du sous-arbre droit sont supérieures ou **égales** à la valeur du nœud X")

3a

26 - 3 - 1 - 15 - 13 - 19 - 25 - 42 - 29 - 32 - 30 - 37

3b

C'est un parcours préfixe

4

```
Parcours2(A)
  Parcours2(A.fils_gauche)
  Afficher(A.valeur)
  Parcours2(A.fils_droit)
```