

# Algorithmique et Programmation 2: Contrôle 2

## 1 Arbres binaires de recherche

### 1.1

Seul (b). (Note pour (c), 3, 2 et 1 sont à droite de 4.)

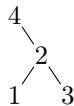
### 1.2

Prefixe: 4, 2, 1, 3, 6, 5, 7.

Infixe: 1, 2, 3, 4, 5, 6, 7 (tous les arbres binaires de recherche produisent un infixe trié).

Postfixe: 1, 3, 2, 5, 7, 6, 4.

### 1.3



## 2 Arbres équilibrés

### 2.1

```
1 || def hauteur(A):
2 ||     if A == None:
3 ||         return -1
4 ||     else:
5 ||         return 1 + max(hauteur(A.gauche), hauteur(A.droit))

6 || def equilibre(A):
7 ||     if A == None:
8 ||         return 0
9 ||     else:
10 ||        return hauteur(A.droit) - hauteur(A.gauche)
```

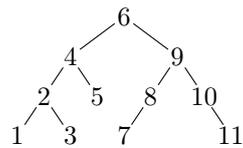
### 2.2

5, 11 et 13.

### 2.3

- (a) 6
- (b) 2
- (c) 3

### 2.4



Complexité  $O(1)$ .

### 2.5

```
11 | def rotate(A) :  
12 |     gg = A.gauche  
13 |     gd = A.droit.gauche  
14 |     d = A.droit.droit  
15 |     A.gauche = Noeud(A.entier)  
16 |     A.entier = A.droit.entier  
17 |     A.gauche.gauche = gg  
18 |     A.gauche.droit = gd  
19 |     A.droit = d
```