

Exercice 1

- 1) hauteur  $h$  : au maximum  $2^h$  nœuds / [donc  $h < \log_2(n)$ ]  
 2) arbre à  $n$  nœuds  $\rightarrow n-1$  arêtes /

Exercice 2 [syntaxe : new Nœud(valeur, filsGauche, filsDroit)]

Nœud a = new Nœud(9, null, null); /

Nœud b = new Nœud(5, a, null); /

Nœud c = new Nœud(7, null, null); /

Nœud racine = new Nœud(3, b, c); /

Exercice 3

```
taille (Arbre a) {
    return tailleAux(a.racine);
}
```

```
tailleAux (Nœud v) {
    if (v == null)
        return 0;
    else
        return 1 + tailleAux(v.gauche)
            + tailleAux(v.droit);
}
```

```
profondeur (Arbre a) {
    return pAux(a.racine);
}
```

```
pAux (Nœud v) {
    if (v == null)
        return -1; // arbre vide
    else
        return 1 + max(pAux(v.gauche),
            pAux(v.droit));
}
```

```
valMax (Arbre a) { // on suppose a != null
    return vAux(a.racine);
}
```

```
auxValMax (Nœud v) {
    if (v.gauche == null) {
        if (v.droit == null) {
            return v.val;
        } else {
            return max(v.val, auxValMax(v.droit));
        }
    } else {
        if (v.droit == null) {
            return max(v.val, auxValMax(v.gauche));
        } else {
            return max(v.val, auxValMax(v.gauche),
                auxValMax(v.droit));
        }
    }
}
```

```
nbNœuds (Arbre a) {
    return aux(a.racine);
}
```

```
aux (Nœud v) {
    if (v == null) {
        return 0;
    } else if (v.gauche == null & v.droit == null) {
        return 1;
    } else {
        return aux(v.gauche) + aux(v.droit);
    }
}
```

2) On suppose  $a \neq \text{null}$

```
estPaire (Arbre a) {  
    return auxP(a.racine);  
}
```

```
auxP (Noeud v) {  
    if (estFeuille(v.droite)) {  
        if (estFeuille(v.gauche)) {  
            return true; // fin de l'arbre  
        } else {  
            return auxP(v.gauche);  
        }  
    } else {  
        return false;  
    }  
}
```

#### Exercice 4

```
infixeAux (Noeud n) {  
    if (n != null) {  
        infixeAux(n.gauche);  
        print(n.v);  
        infixeAux(n.droit);  
    }  
}
```