



## Épreuve de Rattrapage

### Exercice 1 :

On considère la procédure ci-dessous :

Procédure TestCompteur (n : entier)

Début

Si  $n < 13$  Alors

afficher (n+1)

TestCompteur (n+3)

afficher (n-1)

FinSi

FinProcédure

Quel est l'affichage produit par l'exécution de l'algorithme ci-dessous : (2pts)

Algorithme Test

Début

afficher ("Début")

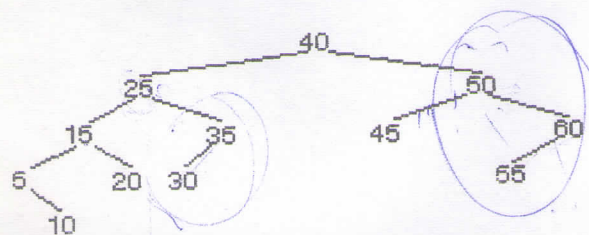
TestCompteur (5)

afficher ("FIN !")

FinAlgorithme

### Exercice 2 :

1. Écrire un algorithme permettant d'afficher les éléments triés par ordre croissant d'un arbre binaire de recherche de racine a . (2pts)
2. On souhaite vérifier un arbre binaire s'il est de recherche ? Écrire une fonction qui teste si un arbre binaire donné en entrée est un ABR. (3pts)
3. On se donne l'arbre AVL ci-dessous : (3pts)



- Donner l'arbre obtenu par suppression de 45 dans cet arbre, puis celui obtenu par suppression de 25 dans le nouvel arbre.

### Exercice 3 :

1. Dites quelle est la complexité de l'algorithme calcul pour  $m < n$ . (2pts )

```
Algo calcul(n,m :entier) : entier
  x ← 1
  pour i ← 1 à n faire :
    pour j ← 1 à m faire :
      x ← x + j
  retourner x
FinAlgo.
```

2. En déduire le résultat. (1pt )

### Exercice 4 :

1. Proposer une suite d'instructions pour réaliser un tri décroissant d'une liste chaînée d'entiers. (3pts )
2. Écrire un algorithme permettant de créer une liste doublement chaînée à partir d'un tableau contigu d'entiers de taille n. (4 pts )