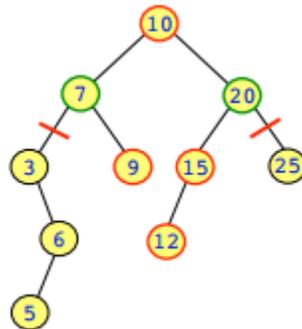


Partie 2 : Programmation fonctionnel pour Recherche en dimension (Emacs)

Supposons par exemple qu'on veuille constituer et manipuler les arbres de recherche binaires (ARB) pour résoudre le problème suivant : Compter les éléments d'un ensemble d'entiers qui sont dans un intervalle donné [min, max] par exemple si on prend que $E=\{3,5,6,7,9,10,12,15,20,25\}$. L'ARB associé sera comme suit :



Si on cherche à compter les éléments entre 9 et 18 alors on trouve 4.

- Dans votre cas l'ensemble à étudier est le suivant :
 $E=\{7,9,10,12,15,20,30,40,55,60,65,70,72,79,80,90\}$
- On cherche de compter les éléments entre 25 et 75 par une fonction comme le suit :
`Count_Interval(Ens :T,min :integer,max :integer) :integer`
- Et si on fait l'appelle de cette fonction alors :
`Count_Interval(E,25,75)=7`

Travail : Vous devez faire dans Emacs les taches suivantes :

1. Construire l'arbre binaire qui correspondant à l'ensemble E et l'afficher.
2. Construire le sous-programme qui fait l'ajout un élément à E.
3. Construire le sous-programme qui supprime un élément de E en respectant tous les cas possibles (patern Matching).
4. Construire le sous-programme qui faite la recherche en dimension de la fonction `Count_Interval(E,min,max)` en faisant l'appelle `Count_Interval(E,25,75)`.
5. Si on ajoute l'élément 53 à E alors :
 - Afficher l'ARB associé à E.
 - Recalculer `Count_Interval(E,25,75)`.
6. Si on supprime l'élément 50 de E alors:
 - Afficher l'ARB associé à E
 - Recalculer `Count_Interval(E,25,75)`.