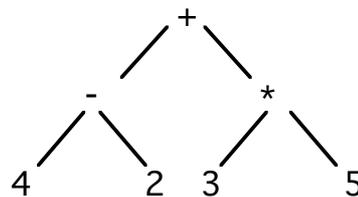


TP numéro 5

1 Arbres binaires pour représenter des expressions arithmétiques

On considère des arbres qui représentent des expressions arithmétiques. Les valeurs des nœuds peuvent donc être soit des nombres, soit les symboles +, -, * et /.

L'expression arithmétique $(+ (- 4 2) (* 3 5))$ sera représentée par l'arbre binaire suivant :



- Écrire une fonction qui prend un tel arbre et qui renvoie la valeur numérique de l'expression correspondante. On supposera que chaque nœud possède soit 0 fils soit 2 fils.

2 Modification d'arbres binaires

- Définir une fonction qui prend en paramètre un arbre binaire de nombres et qui retourne une liste composée de cet arbre dont toutes les valeurs sont désormais positives et du nombre de changements qui ont été effectués pour obtenir ce nouvel arbre.

```

(arbre-positif '(12(-4(2())())(-14(-6(8())(12())(18()))))
→ ((12(4(2())())(14(6(8())(12())(18())))) 3)
    
```

3 Arbres binaires de recherche

- Écrire une fonction qui calcule le minimum d'un arbre de nombres ordonné. Attention à bien tenir compte du fait qu'il s'agit d'un arbre binaire de recherche.
- Écrire une fonction qui construit la liste ordonnée des valeurs d'un ABR comprises entre deux valeurs données. Attention à ne pas parcourir inutilement certaines branches de l'arbre.

4 Parcours selon un chemin

On définit un chemin comme une liste de symboles g ou d (pour gauche et droite). Il s'agit d'une liste des directions à suivre à partir de la racine de l'arbre.

- Écrire une fonction booléenne qui, à partir d'un arbre et d'un chemin, dit si le chemin spécifié sort de l'arbre ou non.
- Écrire une fonction qui calcule la somme des valeurs des nœuds qui se trouvent le long d'un chemin donné. On supposera que le chemin ne sort pas de l'arbre.