

TD 4 - Listes triées

Pour toutes les fonctions du TD, on déterminera leur validité, leur terminaison et leur complexité après les avoir écrites.

Recherche d'une racine d'une fonction continue par dichotomie

L'approximation numérique de la racine d'une fonction à variable réelle par dichotomie a été vue en classe de Terminale. On va traiter cette situation continue dans un premier temps, avant de s'intéresser à la recherche par dichotomie dans une liste triée (situation discrète, plus délicate).

- 1 – Ecrire une fonction `racine_dichot` qui à deux réels a, b , à une fonction continue $f : [a, b] \rightarrow \mathbb{R}$ telle que $f(a)f(b) < 0$ ($f(a)$ et $f(b)$ de signes opposés) et à un réel ε renvoie une valeur approchée à ε près d'une racine de f comprise entre a et b .

Listes triées

Une fois que l'on a des listes triées, la manipulation de ces listes peut s'avérer facilitée, notamment les fonctions de recherche. On va à présent implémenter des fonctions pour les listes triées. Les fonctions seront écrites en supposant qu'elles seront appliquées à des listes triées.

- 2 – Ecrire une fonction `recherche_dichot` qui à une liste triée `l` et une valeur `a` renvoie `True` si `a` est un élément de `l`, et renvoie `False` sinon. La recherche devra être effectuée par dichotomie.
- 3 – Ecrire une fonction `insere_triee` qui à une liste triée `l` et une valeur `a` renvoie une liste triée contenant les éléments de `l` avec la même occurrence que dans `l` et `a`.
- 4 – Ecrire une fonction `concat_triee` qui à deux listes triées `l1` et `l2` renvoie la liste triée issue de la concaténation des listes triées `l1` et `l2`.