

Les tris (séance 2). Exercices

1. Écrire deux versions, l'une récursive, l'autre itérative, d'une fonction booléenne nommée `appartient` qui reçoit comme argument un tableau d'entiers et des indices de début et de fin, ainsi qu'une valeur (entière), et répond `vrai` si la valeur figure dans le tableau. Pour la version récursive, on pourra raisonner de la manière suivante : *un élément appartient au tableau soit s'il se trouve dans la première case, soit s'il appartient au tableau formé en enlevant la première case.*
2. Proposer une version récursive du tri par insertion. La récursion s'arrête lorsqu'on a trié le plus petit tableau à trier.
3. **Recherche dichotomique** : ce type de recherche s'utilise uniquement sur des données triées. Principe :

Soit un tableau d'entiers triés, on compare l'élément recherché avec l'élément au centre du tableau. Trois possibilités :

- (a) c'est l'élément recherché
- (b) l'élément recherché est plus petit que l'élément comparé. On cherchera alors, de la même façon, l'élément dans la partie gauche du tableau.
- (c) l'élément recherché est plus grand que l'élément comparé. On cherchera alors, de la même façon, l'élément dans la partie droite du tableau.

Écrire deux versions, l'une récursive, l'autre itérative, d'une fonction booléenne nommée `appartient_dichotomique` qui prend en arguments un tableau d'entiers ordonnés et un entier à rechercher et répond `vrai` si la valeur figure dans le tableau.