TP 2 : Tri

Objectif du TP

L'objectif de cette séance est de pratiquer la programmation récursive de tris

EXERCICES SUR MACHINES !

Exercice 1 Tris (60 min)

a) À l'aide de la classe de test `TestTriEtudiant` fournie, développez le tri par fusion vu en cours. *Astuce : il est possible de recopier les éléments d'un tableau dans un autre tableau à l'aide de la méthode `Arrays.copyOfRange`.*

b) Dans le cours, nous avons développé une méthode itérative `fusion(t1,t2,t)` qui fusionne deux tableaux triés t1 et t2 et stocke le résultat dans t. Écrivez une méthode récursive qui fait exactement la même chose.

c) Pour le tri rapide, écrivez une méthode `int[] partitionPivot(int[], int min, int max, int pivot)` qui se base sur le principe de `partitionV2OK` vue en cours en considérant le prédicat `"< Pivot"`.

d) Écrivez une nouvelle méthode de partitionnement qui considère le prédicat `"est pair"`.

e) Ajouter des variables à l'algorithme de tris qui permettent de mesurer le nombre de copies de tableaux effectuées, le nombre d'échanges effectués, le nombre d'affectations effectuées et affichez les à la fin du test.

f) Lancez plusieurs tris sur des tableaux de tailles différentes (5, 10, 100, 500, 1000, 5000, 10000) et avec à chaque fois les 4 méthodes d'initialisation. Retenez pour chacun de ces tris, les valeurs des variables décrites en c) et tracez une courbe pour chaque type d'initialisation en fonction de la taille du tableau.