Aix*Marseille université

Domaine Sciences et Technologies
LICENCE INFORMATIQUE

Algorithmique 1 : TP tris
Code UE : S04IN4A2

Année 2022-2023

TP tris

L'objectif de ce TP est de comparer expérimentalement différents tris. Vous allez implémenter les tris qui nous intéresse, puis vous comparerez les temps d'exécution de chacun sur des suites de longueurs diverses. Ces suites sont constituées d'éléments tous de même type et qui sont comparables. Ce pourra être par exemple des suites d'entiers, de chaînes de caractères,... ou tout autres objets dont la classe implémente l'interface `Comparable`. Ce TP s'étends sur 3 séances. Lors de la dernière séance vous montrerez votre projet au chargé de TP pour évaluation. Vous implémenterez les tris suivants :

- tri à bulle,
- tri par fusion,
- tri rapide *avec la partition dite du drapeau*,
- tri par tas
- (tri par base.)

Suivez l'ordre indiqué dans la liste pour l'implémentation. Tous ces tris seront présentés en cours ou en TD. Vous représenterez les suites à trier par des `ArrayList`, sauf pour le tri par fusion (voir les explications plus bas), et vous accèderez aux éléments de la suite à l'aide des méthodes `get` et `set`. Vous pourrez définir l'opération `permuter` qui permute deux éléments dans la liste à partir de leurs positions courantes. Certains algorithmes de tri présentés en cours utilisent une numérotation des éléments de la suite à partir de 1 $(e_1, e_2, \ldots, e_n)$. L'indexation dans `ArrayList` débute à l'indice 0 $(e_0, e_1, \ldots, e_{n-1})$, vous devrez donc réécrire l'algorithme du cours afin d'être cohérent avec cette indexation.
Suivez scrupuleusement les indications, et récapitulez dans un tableau que vous joindrez au projet les résultats que vous avez obtenus et les observations que vous pouvez faire.

Génération des suites

Afin de comparer les tris sur des suites de grandes tailles vous commencerez par écrire des fonctions de génération (dépendant du type des objets de la suite que vous voulez générer, entiers, caractères,...). La classe `java.util.Random` propose des méthodes pour générer des valeurs aléatoires de différents types. Vous écrirez au moins une fonction pour générer des suites de nombres entiers qui prendra en argument les bornes entre lesquelles sont choisies les valeurs et leur nombre. Vous pourrez ainsi comparer les différents tris sur des suites de longueurs différentes contenant des valeurs plus ou moins semblables.

Tri par fusion

Le tri par fusion est un tri qui a la particularité d'être bien adapté lorsque les suites sont représentées par des listes chainées. En effet, la fusion et la séparation sont faciles à implémenter et peu couteuses dans ce cas. Vous utiliserez des objets de type `LinkedList`. Cette classe propose toutes les méthodes dont vous aurez besoin, notamment `add` qui ajoute

un élément à la fin de la liste, ou `addAll` qui ajoute tous les éléments d'une collection, par exemple une `ArrayList`, à la liste chainée. Vous devrez écrire une fonction pour partitionner une liste chainée en deux listes de longueurs égales à un élément près, et une fonction qui fait le fusion de deux listes ordonnées de façon efficace.