
Feuille de TP - 1 (1 séance)
Implémentation en Java

Exercice 1. Soit S un tableau d'entiers. Nous supposons dans un premier temps que le nombre n d'éléments de S est une puissance de 2. Le tri par fusion de S consiste à :

- (1) diviser S en deux parties égales (la première moitié notée S_1 , et la deuxième moitié notée S_2),
- (2) trier récursivement chaque partie, et
- (3) fusionner les tableaux triés résultant de ces deux tris, afin d'obtenir un tableau trié contenant tous les éléments de S .

Ces étapes sont à faire tant que le nombre d'éléments dans le tableau à trier est d'au moins 2. Les tableaux de longueur 1 sont déjà triés.

Notez que la division de S en deux parties égales peut être faite de deux manières : **avec copie**, lorsque les éléments de S sont copiés dans deux tableaux contenant chacun la moitié des éléments, et le travail à effectuer se fait sur chacun de ces tableaux ; ou **sans copie**, lorsque les éléments de S restent dans le tableau S , et le travail se fait sur la première moitié de ce tableau (définie par deux indices deb_1 et fin_1), respectivement sur la deuxième moitié de ce tableau (définie par deux indices deb_2 et fin_2).

En ce qui concerne la fusion des tableaux triés, il est nécessaire de définir un nouveau tableau, recevant le résultat de la fusion.

On demande :

- (a) Programmez le tri par fusion, en utilisant de préférence la variante sans copie.
- (b) Modifiez votre algorithme pour qu'il traite des tableaux dont la longueur n'est pas forcément une puissance de 2.
- (b) Quelle est la complexité de votre algorithme ?

Pour aller plus loin

Exercice 2. Programmez l'algorithme de recherche dichotomique sur une séquence ordonnée d'entiers.

Exercice 3. Programmez le calcul de la **suite de Syracuse** pour un entier positif n .

La suite de Syracuse est définie comme suit : le premier élément est un nombre entier strictement positif n ; si n est pair, on le divise par 2 ; si n est impair, on le multiplie par 3 et

on ajoute 1. Ceci fournit le 2ème élément de la suite. Le 3ème élément se calcule de la même manière, à partir du 2ème élément. En répétant l'opération, on obtient une suite d'entiers strictement positifs, appelée suite de Syracuse.

Par exemple, la suite de Syracuse pour $n = 6$ est : 6, 3, 10, 5, 16, 8, 4, 2, 1, 4, 2, 1, ...

Une conjecture dit que, quel que soit n , le nombre 1 est obtenu tôt ou tard (après quoi, les éléments 1, 4, 2 apparaissent à l'infini dans cet ordre). Dans votre programme, vous pouvez arrêter la génération d'éléments une fois que vous avez obtenu 1. Si votre programme boucle, vous avez peut-être résolu la conjecture en lui apportant un contre-exemple (mais il y a aussi une infime chance que vous ayez fait une erreur dans votre programme).

Plus d'informations sur cette suite : https://fr.wikipedia.org/wiki/Conjecture_de_Syracuse