

Activité : Diviser pour régner

Introduction

L'expression **diviser pour régner** est très ancienne et se retrouve dans les manuels d'histoire. C'est une stratégie d'affaiblissement d'un adversaire qui consiste à diviser ses membres et à les opposer pour semer la discorde et ainsi rendre la victoire plus facile.

En informatique, le concept est plus pacifique, mais la stratégie demeure. L'algorithme consiste à :

- Diviser, c'est à dire découper un problème initial en sous-problèmes ;
- Régner en résolvant les sous problèmes plus petits donc plus simples.
- Combiner les solutions des sous-problèmes pour donner une solution au problème initial.

Le **tri fusion** d'un tableau de nombres est une application du principe **diviser pour régner**.

- L'algorithme consiste à diviser ce tableau de nombres en 2 tableaux, puis chaque tableau en 2 nouveaux tableaux, etc.
- Une fois les tableaux divisés jusqu'à avoir qu'une seule valeur, on recompose les tableaux par assemblage en triant les éléments.

Vous trouverez des informations sur la page wikipedia suivante : https://fr.wikipedia.org/wiki/Tri_fusion

Diviser des tableaux

On souhaite créer un tableau de valeurs à partir d'un autre tableau plus grand. Les valeurs sélectionnées se suivent dans le grand tableau ou sont séparées par un même nombre de valeurs.

Le **slice** est une technique en Python qui permet de sélectionner des éléments d'un tableau et se note avec les deux points.

Par exemple, pour extraire les valeurs d'un tableau `t`, on utilise la syntaxe : `t[indice début : indice fin : écart]`.

Voici trois règles utiles.

- Si l'indice de début est 0, on peut ne pas l'écrire.
- Si l'indice de fin correspond à la dernière valeur, on peut laisser vide.
- Si les valeurs à sélectionner se suivent, l'écart n'est pas nécessaire.

Soit `t` un tableau de 8 valeurs. Créer à partir de ce tableau :

- 1) Un tableau sans la première et la dernière valeur ;
- 2) Un tableau composé de la première moitié de tableau ;
- 3) Un tableau composé de la seconde moitié de tableau ;
- 4) Un tableau avec les deux valeurs centrales ;
- 5) Un tableau avec les valeurs d'indices pairs ;
- 6) Un tableau avec les valeurs d'indices impairs ;

Fusionner des tableaux

La fusion de 2 tableaux peut se réaliser de plusieurs façons.

- 1) La concaténation est une opération consistant à assembler deux tableaux en un seul. Le signe d'addition + est utilisé pour la concaténation.
 - a) Concaténer les tableaux [3] et [5].
 - b) Concaténer les tableaux [5] et [3].
 - c) Concaténer les tableaux [3,7] et [5,8].
 - d) Les tableaux obtenus par concaténation ont-ils leurs valeurs triées ? Pourquoi ?
- 2) Il est possible de fusionner deux tableaux triés en un seul tableau trié. Pour y parvenir, il faut parcourir chaque tableau et insérer dans un nouveau tableau les valeurs en respectant l'ordre.
 - a) En réalisant des schémas, décrire la fusion des tableaux [1,6,8] et [3,5].
 - b) Écrire en Python une fonction `fusion_tableaux` qui prend en paramètre 2 tableaux triés et renvoie un tableau trié avec les valeurs des 2 tableaux passés en argument.

Le tri fusion

L'algorithme se décompose en 2 temps. Le premier temps consiste à **diviser** le tableau en tableaux vides ou à 1 valeur. Le second temps consiste à **fusionner** les tableaux triés jusqu'à obtenir le tableau initial trié.

- 1) Appliquer le tri fusion sur la liste $L = [7, 4, 9, 1, 3, 5, 8]$ en donnant précisément les différentes étapes.
- 2) On donne le script de la division :

```
def tri_fusion(liste):  
    if len(liste)<2:  
        return ...  
    else:  
        # on divise en 2 listes la liste reçue en paramètre  
        list1=tri_fusion(...)  
        liste2=tri_fusion(...)  
        # après division on fusionne les listes  
        return fusion(..., ...)
```

Saisir et compléter ce code dans votre script Python.

- 3) Écrire la fonction `fusion` qui effectue la fusion de 2 tableaux triés.
- 4) Créer des listes aléatoires de nombres entiers puis vérifier que le tri fusion est réalisé.
- 5) On va comparer l'efficacité du tri fusion par rapport au tri par sélection et tri par insertion.
 - a) Recréer la fonction de tri par sélection d'une liste de nombres.
 - b) Effectuer des mesures sur le temps de tri pour chacune des méthodes (fusion, sélection) en prenant des tailles de listes de 100 à 1000 nombres avec un pas de 100.
 - c) Représenter sur un même graphique les mesures.