

J1MI2013: Algorithmes et Programmes: feuille 8**Tris(1)****Travaux pratiques****Exercice 1. Expérimentation**

Lancer l'application <http://inriamecsci.github.io/#!/grains/methodes-tri>

1. Testez le tri sélection en visuel (sur cette application l'algorithme cherche le maximum au lieu du minimum). A l'aide de l'onglet `tri temporel` faites des tests sur le temps de calcul pour trier par exemple :
 - 100 tableaux de taille 100
 - 100 tableaux de taille 200
 - 100 tableaux de taille 500
 - 500 tableaux de taille 500
 - 1000 tableaux de taille 500
 - 1 tableau de taille 3000
2. Même chose avec le tri insertion.
3. Comparez les résultats stockés dans le journal avec les résultats théoriques vus en cours et en TD

Récupérer depuis le site de l'UE l'archive `tp08.tar.gz` et désarchivez-le. Placez vous ensuite dans le répertoire `tp08`.

Exercice 2. Tri par sélection

Dans le fichier `testTri.c` :

1. Écrire une fonction `indiceMin`, qui renvoie l'indice d'un élément minimal d'un tableau, entre les indices `i` et `n-1` inclus. Quels seront les paramètres de cette fonction ?
2. Écrire une fonction `echanger` qui échange le contenu de deux cases d'un tableau d'entiers.
3. Écrire un algorithme itératif `triSelection` qui appelle `indiceMin` et `echanger` pour effectuer le tri.
4. Tester votre programme pour différentes séquences d'entiers donnés en arguments de la ligne de programme.

Exercice 3. Tri par insertion

1. En supposant que les éléments d'indice inférieur à `i` d'un tableau `t` sont déjà triés, quelles instructions permettent d'insérer `t[i]` à sa place parmi les premiers `i + 1` éléments de `t` ?
2. Dans le fichier `testTri.c`, écrire une fonction itérative `triInsertion` qui mets en œuvre l'algorithme de tri par insertion.
3. Tester votre programme pour différentes séquences d'entiers donnés en arguments de la ligne de programme.