

TD – 01

Exercices d'applications

Savoirs et compétences :

- Alg – C17 : tris d'un tableau à une dimension de valeurs numériques (tri par insertion, tri rapide, tri fusion).

Exercice 1 – Représentation du coût temporel des tris

Objectif Représenter pour chacun des tris les courbes indiquant le temps d'exécution en fonction du nombre d'éléments à trier.

On donne la bibliothèque de tri `tris.py` dans laquelle différents tris ont été implémentés. On dispose ainsi des fonctions :

- `tri_insertion`;
- `tri_rapide`;
- `tri_fusion`.

On dispose aussi de la méthode `sort` disponible en Python.

On pourra importer le module `**` de la bibliothèque et le module `randint` de la bibliothèque `random`.

Question 1 Tracer, dans chacun des 4 cas, le temps de tri d'une liste en fonction du nombre d'éléments de la liste. Le nombre d'éléments variera de 0 à 100 000. Une liste de n éléments sera composée de nombres choisis aléatoirement entre 0 et n .

Question 2 Donner, pour chacun des cas, la complexité dans le meilleur des cas, le pire des cas et le cas moyen.

Exercice 2 – Tri à bulles

Question 1 Donner l'algorithme permettant de trier un algorithme selon la méthode du tri à bulles.

Question 2 Dans quel cas est-on dans le meilleur des cas ? Quelle alors la complexité de l'algorithme ?

Question 3 Dans quel cas est-on dans le pire des cas ? Quelle alors la complexité de l'algorithme ?

Exercice 3 – Tri cocktail

Question Donner l'algorithme permettant de trier un algorithme selon la méthode du tri cocktail.

Question 1 Dans quel cas est-on dans le meilleur des cas ? Quelle alors la complexité de l'algorithme ?

Question 2 Dans quel cas est-on dans le pire des cas ? Quelle alors la complexité de l'algorithme ?