Algorithmes de tris

Compétences

- Implémentation de tris
- Évaluer la complexité expérimentale de différents tris

Introduction

Nous venons d'étudier dans le TD6 plusieurs algorithmes de tri :

- le premier un tri récursif nommé XSort
- le deuxième basé sur une amélioration du tri insertion,
- le dernier une variante du tri fusion.

Les deux derniers tris partent de l'hypothèse que l'on peut exploiter les sous-séquences triées d'un tableau.

Étude expérimentale du XSort

Une première intuition est que le tri XSort a une complexité dans le pire cas en $O(2^n)$ avec n la taille du tableau. Vérifiez empiriquement cette intuition et expliquez l'expérience que vous avez réalisée et en quoi elle confirme ou infirme une complexité $O(2^n)$ dans le pire cas.

Temps d'exécution sur des tableaux d'entiers avec tirage pseudo-aléatoires

Pour effectuer un tirage d'entiers pseudo-aléatoire on peut utiliser l'instruction suivante :

```
1 # import de la fonction randint avec l'alias rd
2 from random import randint as rd
3 # un tableau T de 5 valeurs entières dans l'intervalle [0, 10]
4 T = [rd(0, 10) for i in range(5)]
```

À l'aide d'une méthode empirique, remplir le tableau ci-dessous avec le temps moyen d'exécution de chaque algorithme de tris pour un tableau de taille n. Les calculs doivent être effectués pour un jeu de 10 expériences sur chaque tri.

n	Tri Insertion	newSort1	newSort2	Tri Fusion	Tri Rapide
1					
10					
100					
1000					
10000					
100000					
1000000					

Comparer les résultats obtenus avec les complexités théoriques.

Temps d'exécution sur des tableaux contenant des sous-séquences pré-triées

Pour une taille de tableau $n = 5000$, remplir le tableau ci-dessous avec le temps moyen d'exécution de chaque algorithme de tris. Les calculs doivent être effectués pour un jeu de 10 expériences sur chaque tri. Le code permettant d'obtenir des sous-séquences triées est fourni dans le fichier `sous_sequences_triees.py`

k	Tri Insertion	newSort1	newSort2	Tri Fusion	Tri Rapide
1					
10					
100					
500					
1000					
3000					
5000					

Comparer les résultats obtenus avec les complexités théoriques.