

ALGORITHME POUR LA RESOLUTION DES PROBLEMES

Représentation du Problème

■ Description du Problème

Le tri est un problème fondamental en informatique qui consiste à organiser des éléments dans un ordre particulier, généralement croissant ou décroissant. Des algorithmes de tri efficaces sont essentiels pour optimiser les performances dans diverses applications, telles que l'indexation et la recherche dans les bases de données.

■ Représentation Diagrammatique

Tableau non trié : [5, 2, 9, 1, 6]

Tableau trié (ordre décroissant) : [9, 6, 5, 2, 1]

Solution

■ Algorithme Utilisé : Tri par Fusion

Le tri par fusion est un algorithme diviser-pour-régner qui divise récursivement le tableau en sous-tableaux plus petits, les trie, puis fusionne les sous-tableaux triés.

■ Implémentation du Code (Python)

```
TD00 > Report structures.py > ...
1  def tri_par_fusion(arr):
2      if len(arr) > 1:
3          milieu = len(arr) // 2
4          partie_gauche = arr[:milieu]
5          partie_droite = arr[milieu:]
6
7          tri_par_fusion(partie_gauche)
8          tri_par_fusion(partie_droite)
9
10         i = j = k = 0
11         while i < len(partie_gauche) and j < len(partie_droite):
12             if partie_gauche[i] < partie_droite[j]:
13                 arr[k] = partie_gauche[i]
14                 i += 1
15             else:
16                 arr[k] = partie_droite[j]
17                 j += 1
18             k += 1
19
20         while i < len(partie_gauche):
21             arr[k] = partie_gauche[i]
22             i += 1
23             k += 1
24
25         while j < len(partie_droite):
26             arr[k] = partie_droite[j]
27             j += 1
28             k += 1
29
30     # Exécution de l'exemple
31     arr = [5, 2, 9, 1, 6]
32     tri_par_fusion(arr)
33     print("Tableau Trié :", arr)
```

Résultats

Exemple d'Entrée et de Sortie

Entrée : [5, 2, 9, 1, 6]

Sortie : [1, 2, 5, 6, 9]

Le tri par fusion trie efficacement le tableau en **$O(n \log n)$** , ce qui le rend très performant pour les grands ensembles de données.

Conclusion

Le tri par fusion est un algorithme de tri efficace et stable qui garantit une complexité temporelle dans le pire des cas de **$O(n \log n)$** . Son approche diviser-pour-régner assure une performance optimale par rapport aux méthodes de tri simples comme le tri à bulles (**$O(n^2)$**). Cependant, il nécessite une mémoire supplémentaire pour la fusion, ce qui le rend moins efficace en termes d'espace par rapport aux algorithmes de tri en place comme le tri rapide.

Dépôt GitHub

Le code complet et le rapport sont disponibles dans le dépôt GitHub suivant :