L’aumonier Timothée Groupe A2

Walter Malcolm

Viguoine Claire

Bonjour, voici un document texte pour justifier nos choix pour la fonction recherche et celle du tri.

Nous avons utilisé le tri par insertion car il a une meilleure complexité temporelle que le tri par sélection, le tri par insertion a en moyenne une complexité temporelle de $O(n^2)$ et dans le meilleur cas de $O(n)$ alors que le tri par sélection est toujours de $O(n^2)$ avec tous les deux une complexité spatiale de $O(1)$ dans le pire cas.

Nous avons effectué une recherche séquentielle car le tableau contenant les noms et les prénoms n'est pas trié constamment, alors nous ne pouvons pas effectuer une recherche dichotomique car si l'on veut trier puis rechercher alors la complexité temporelle serait de $O(n^2) + O(\log(n))$ alors que la recherche séquentielle est en moyenne de $O(n)$.

Nous avons finalement opté sur un tri fusion pour notre fonction, ce tri repose sur un principe : « diviser pour régner ». C’est avec ce principe qui lui permet de diviser une grosse opération en plein de petites qu’il arrive à se démarquer de par sa vitesse et son efficacité. Bien que plus complexe et plus élaboré, sa complexité algorithmique en est remarquable avec une complexité de $O(n*\log(n))$ ce qui en fait un tri réellement puissant et compétitif qui permettrai sur un long fichier de données d'éviter une complexité comme avec le tri par sélection de $O(n^2)$ qui est immensément plus conséquente. Le tri fusion étant l'un des plus rapides il est sans aucun doute un très bon choix qu'importe l'usage et c'est donc une valeur sûre qui nous permet un tri rapide et efficace là où notre idée de départ aurait montré quelques lenteurs et réticences sur un énorme jeu de données. C’est aussi le seul algorithme de tri avec une telle complexité en restant stable et nous assure donc un tri toujours comme escompté

Sources :

Complexités algorithmiques : https://fr.wikipedia.org/wiki/Algorithme_de_tri

Section : "Comparaison des algorithmes"