

Sorting en C++

Prérequis : Pour mener à bien ce travail, il est nécessaire d'avoir installé les outils classiques pour le C++ (compilateur, CMake, etc.).

Contexte du travail : On s'intéresse ici à l'implémentation en C++ de méthodes de tri de tableau. En particulier, deux méthodes sont considérées dans ce TP :

- tri à bulle : https://fr.wikipedia.org/wiki/Tri_à_bulles
- tri rapide : https://fr.wikipedia.org/wiki/Tri_rapide

Travail demandé : Le but de ce travail est d'implémenter en C++ les deux algorithmes :

- L'implémentation devra utiliser au maximum la librairie standard `std` et les possibilités du langage offertes par les normes C++11/14/17.
- Il conviendra d'implémenter une vérification du tri, par exemple en parcourant l'ensemble du tableau trié et en vérifiant la position de chaque élément.
- Les données devront être générées de manières aléatoires.
- Le temps d'exécution de chaque méthode devra être mesuré pour le tri de petits et grands vecteurs.

Quelques pistes utiles : Voir <https://en.cppreference.com/w/cpp> pour la documentation officielle du C++.

- Ne pas hésiter à utiliser et abuser du mot-clé `auto`.
- En utilisant `<random>`, il est possible de générer des nombres suivant les distributions classiques.
- En utilisant `<chrono>`, il est possible d'instrumenter les implémentations.

Liens utiles : Voici quelques ressources qui devraient vous inspirer :

- <https://en.cppreference.com/w/cpp/language/templates>
- <https://en.cppreference.com/w/cpp/container/vector>
- <https://en.cppreference.com/w/cpp/language/auto>
- <https://en.cppreference.com/w/cpp/language/range-for>
- <https://en.cppreference.com/w/cpp/numeric/random>
- <https://en.cppreference.com/w/cpp/chrono>

Bonus : Vous pourrez comparer vos implémentations avec la fonction de tri fournie par la librairie standard `std::sort`.