

TD2

C++ - ENSISA 2A

Ali El Hadi ISMAIL FAWAZ

November 14, 2024



Une école d'ingénieurs de l'Université de Haute-Alsace



Exercice I: Algorithmes de Recherche

1. Définir une classe 'SearchingAlgorithm' qui contient les éléments suivants:
 - 'numberComparisons' qui compte le nombre de comparaison fait pour chercher un élément
 - 'totalComparisons', une variable statique, qui compte le nombre total de comparaison pour toutes les instances de cette classe
 - 'totalSearch', une variable statique, qui compte le nombre total de fois que le code a cherché un élément pour toutes les instances de la classe
 - 'averageComparisons', une variable statique, qui contient le nombre moyen des comparaisons sur toutes les instances de la classe.
2. Définir le constructeur par défaut.
3. Définir la fonction virtuelle pure 'search' qui prend comme argument un vecteur de la librairie standard ('int' éléments) et un 'int' pour chercher dans ce vecteur. La fonction doit rendre un 'int' qui représente l'indice de l'élément, cette fonction rend '-1' si l'élément ne se trouve pas dans le vecteur.
4. Une fonction 'displaySearchResults' qui prend comme argument un 'ostream&' et 2 'int' (results et target) qui montre le nombre de comparaisons total pour trouver 'target' sur l'indice 'results'. Cette fonction calcule le 'totalComparisons' et calcule le 'averageComparisons'.
5. Implémenter les 3 algorithmes de recherches: 'LinearSearch', 'JumpSearch' et 'BinarySearch'

Exercice II: Matrices

1. Définir la classe 'MatrixBase' qui contient les éléments:
 - data, un vecteur template en 2D
 - rows et cols, des variables 'size_t'
2. Définir les constructeurs
3. Définir les fonctions 'addElement' et 'getElement'
4. Définir les fonctions 'getRows' et 'getCols'
5. Définir la fonction 'Display'
6. Définir la classe 'MatrixNumerical' qui hérite la classe 'MatrixBase' spécifique pour les valeurs numériques ('int', 'double', 'float' etc.)
7. Définir la fonction 'getDeterminant'
8. Définir les constructeurs
9. Définir les opérateurs '+', '-' et '*'
10. Définir la fonction 'getInverse' qui utilise la méthode de CoFactor (hint: définir une fonction hors la classe 'getCoFactor')
11. Définir l'opérateur '/'
12. Définir la fonction statique 'getIdentity' qui rend une matrice carrée identité, la taille de la matrice est donnée comme argument 'int'.