

Projet C++ : Matrices

Avant de commencer : la qualité des commentaires, avec notamment la présence des antécédents, des conséquents, des invariants de boucle, les rôles de chacune des fonctions, ainsi que les noms donnés aux variables, l'emploi à bon escient des majuscules et la bonne indentation rentreront pour une part importante dans l'appréciation du travail. Ce projet doit permettre de montrer votre autonomie et votre compréhension tant dans la conception du programme que dans sa réalisation. Enfin, si les codes de plusieurs projets se trouvent être identiques, ou être copiés depuis le web, tous les projets concernés seront immédiatement sanctionnés par un zéro.

1 Matrice et MatriceDouble

Programmez tous les exercices du DS du 13 décembre 2021. Le sujet est en ligne pour mon site web. Vous modifierez l'en-tête de la méthode `get` de la classe `Matrice<T>` de la façon suivante :

```
virtual T get(const int i, const int j) const =0;
```

2 MatriceCreuse

On appellera *matrice creuse* une matrice de réels **double** qui contient au moins 90% de 0.0. Vous écrirez la classe `MatriceCreuse` pour représenter ces matrices. Vous choisirez une représentation idoine pour ne mémoriser que les éléments non nuls.

Vous implémenterez les méthodes de la classe `Matrice<T>` auxquelles vous ajouterez :

- le produit de produit et la somme de matrices ;
- la méthode `estCreuse` qui teste si une matrice est creuse ou non ;
- la conversion d'une `MatriceCreuse` en `MatriceDouble` et réciproquement ;

Ces méthodes sont à mettre dans toutes les classes.

3 Programme de test

Vous écrirez un programme qui teste toutes vos méthodes précédentes pour mettre en évidence leur bonne programmation.

4 Remise du projet

Votre projet est à faire en *binôme*. Il est à rendre au plus tard :

Le 16 janvier 2022, 23h59

Vous déposerez votre projet dans l'espace *Projet n. 1* du cours *EIEL721 - ECUE Programmation objet* sur le moodle de l'université sous forme une archive (tar) comprimée (gz) : `matrices-nom1-nom2.tgz` avec `nom1` et `nom2` les noms de chaque étudiant du binôme.

4.1 L'archive devra contenir :

- vos fichiers source `.cpp` et `.hpp` correctement documenté (chaque fonction doit avoir un commentaire, les invariants de boucle doivent être marqués), indenté, et codé (les noms de variables explicites, éviter les trop longues fonctions) , et un fichier `Makefile`.
- un fichier `Documentation` au **uniquement** format `pdf` et décrivant l'organisation de vos classes et vos choix de programmation ;
- la compilation avec les options `-Wall` ne doit pas donner de *warning*.

Bon travail et bon courage