




# TP 22 – Exceptions

 Une version interactive de cet énoncé est disponible dans le fichier **VEtudiant.zip**. Décompressez le fichier et naviguez vers le fichier **doc-Test/html/index.html**. Ce fichier est également le point d'entrée vers l'aide en ligne du code déjà réalisé.




## Objectifs du TP

L'objectif du TP est de reprendre la classe matrice du TP précédent et de gérer les erreurs potentielles.

## Exercice 1 Génération d'une erreur

-  Dans le programme principal, créez une matrice **matA** de taille 2x3 et une matrice **matB** de taille 3x2. Initialisez leurs valeurs.
-  Créez une troisième matrice **matC** et copiez le code suivant :

```
matC = matA + matB;
std::cout << "Affichage des matrices :\n"
           << "matA = \n"
           << matA << "\n"
           << "matB = \n"
           << matB << "\n"
           << "matC = \n"
           << matC << "\n";
```








-  Compilez et exécutez votre programme.
  -  Que se passe-t-il ?
  -  Déterminez pourquoi le programme se termine sans afficher les matrices.

## Exercice 2 Création d'une classe d'erreur



Les erreurs possibles dans la classe matrice sont les suivantes :

Fonction membre	Erreur
<code>operator () (size_t, size_t) const</code>	Les indices sont hors de la plage
<code>operator () (size_t, size_t)</code>	Les indices sont hors de la plage
<code>operator*(const CMatrice&amp; mat) const</code>	Les tailles ne sont pas compatibles
<code>operator+=(const CMatrice&amp; mat)</code>	Les tailles ne sont pas compatibles
<code>operator-=(const CMatrice&amp; mat)</code>	Les tailles ne sont pas compatibles


Pour ces deux types d'erreur, nous pouvons leur attribuer un code d'erreur. Ce code d'erreur peut être stocké dans une classe puis interprété par une méthode de cette classe pour afficher le message d'erreur.

-  Créez une classe **CErreurMatrice** qui a pour attributs :
  -  Un entier qui sera le code d'erreur
  -  Une chaîne de caractères qui identifiera la fonction qui a généré l'erreur.
  -  Un constructeur avec deux paramètres qui initialisera les deux attributs.
  -  Un accesseur public pour chacun des attributs.
-  Définissez l'opérateur de sortie vers flux permettant d'afficher le message d'erreur correspondant au code d'erreur d'un objet **CErreurMatrice**.
-  Testez cette classe dans un programme principal.

### Exercice 3 Utilisation de la classe d'erreur

-  Par le mécanisme des exceptions, utilisez la classe **CErreurMatrice** dans la classe matrice pour signaler les erreurs au programme principal.
-  Modifiez le programme principal de l'exercice 1 en conséquence pour que la gestion soit complète. Le programme doit dans tous les cas afficher les matrices **matA**, **matB** et **matC**. Si une erreur survient, elle doit simplement être affichée à l'utilisateur via le flux d'erreur.

### Exercice 4 Test complet

-  Ecrivez un programme principal permettant de tester toutes les erreurs possibles de la classe **CMatrice**.