

TP 7 : Tableaux à deux dimensions

Ce TP est consacré à la programmation sur les tableaux à deux dimensions (Tableaux 2D). Mis à part le premier exercice qui sera fait avec Jupyter, tout le reste du TP sera programmé avec de la compilation dans un terminal.

N'oubliez pas de commencer par soumettre votre TP de la semaine précédente.

Puis comme d'habitude, utilisez la commande `info-111` pour télécharger les fichiers du TP puis lancer Jupyter.

Exercice 1 (Déclaration, Allocation, Initialisation et affichage de tableau 2D).
Voir la feuille jupyter `feuille_tableau2D.ipynb`.

Par la suite, pour chaque exercice, on vous fournit un fichier squelette qu'il faut compléter, en **remplaçant** les lignes de la forme

```
// Remplacer la ligne suivante par le code adéquat
throw runtime_error("Fonction nombreMines non implanté ligne 33");
```

et les remplaçant par votre code. Comme on verra en amphi, `throw` signale une erreur : ici le fait que la fonction n'est pas encore implanté ; il faut donc bien supprimer cette ligne.

Une fois complété, il faut le compiler dans un terminal avec la commande :

```
info-111 g++ nomFichier.cpp -o nomFichier
```

Pour l'exécuter il faut taper :

```
./nomFichier
```

Attention : Il est recommandé de compiler et exécuter vos fichiers à chaque fois que vous avez écrit une fonction, afin de détecter les erreurs au fur et à mesure.

Pourquoi compiler avec `info-111 g++` ? Vous pouvez tout aussi bien, comme la semaine dernière, compiler directement avec `g++` :

```
g++ -std=c++11 nomFichier.cpp -o nomFichier
```

Simplement, `info-111 g++` rajoute automatiquement `-std=c++11` et quelques autres options pratiques pour déboguer ou pour mieux détecter les erreurs. Regardez les messages affichés et notez lesquelles ! Ainsi, si vous travaillez chez vous directement sur votre portable et n'avez pas la commande `info-111`, vous pourrez continuer à bénéficier de ces options.

Exercice 2 (Déclaration, Allocation et Initialisation de tableau 2D).

Éditer le fichier `tableau2D-initialisation.cpp`, le compléter avec l'éditeur de texte `jedit` (ou autre), le compiler et l'exécuter pour le vérifier.

Exercice 3 (Opérations sur Tableaux à deux dimensions).

Faire la même chose (éditer, compléter, compiler, exécuter, tester) avec les fichiers :

- `tableaux2D.cpp`
- `matrices.cpp`

— `occupation-salle.cpp`

Exercice ♣ 4 (Le jeu du démineur).

Faites la même chose (éditer, compléter, compiler, exécuter, tester) avec le fichier `demineur.cpp`