

### Sujet 4 : Les objets et les pointeurs

**Buts :** Tableaux dynamiques, partage de la mémoire, allocation dynamique, désallocation dynamique, objet.

## Les objets et les pointeurs

Dans cet exercice nous serons amenés à réaliser une classe `VecteurND` permettant de manipuler un vecteur en  $N$  dimensions. Ses coordonnées sont réelles.

VecteurND
<ul style="list-style-type: none"> <li>- unsigned int dim</li> <li>- double * coord</li> </ul>

Il vous est demandé de travailler dans les trois fichiers suivants :

- *VecteurND.h* (définition de la classe),
- *VecteurND.cc* (définition des méthodes de la classe),
- *Principal.cc* (fonctions externes à la classe et fonction `main`).

**Q1.** Définir la classe `VecteurND`. Ecrire les constructeurs et destructeur de cette classe si nécessaire. Définir les méthodes suivantes :

- un accesseur à la dimension du `VecteurND`,
- une méthode permettant d'afficher un `VecteurND`,
- une méthode permettant de calculer la norme d'un `VecteurND`,
- une méthode permettant d'obtenir le produit scalaire de 2 `VecteurND`.

Proposer un jeu de tests **pertinent** permettant de tester l'ensemble des fonctionnalités des classes développées.

**Q2.** Afin de permettre une plus grande flexibilité dans l'utilisation des `VecteurND` il serait intéressant de surcharger les opérateurs communément utilisés sur des vecteurs.

Ajouter à la définition de la classe `VecteurND` l'ensemble des opérateurs suivants : `+`, `-` unaire, `*` (produit par un nombre réel), `<` (qui compare les normes de deux vecteurs), `=` (si nécessaire), `<<`, `>>` et `[]`. En cas d'impossibilité d'obtention d'un résultat une exception sera lancée.

**Q3.** On souhaiterait maintenant revoir l'implémentation de la classe `VecteurND`. Le tableau contenant les coordonnées du vecteur sera stocké dans un `vector<double>`.

VecteurND
- vector<double> coord

Reprendre les questions 1 et 2 avec cette nouvelle représentation, l'interface utilisateur ne devra pas changer (on proposera strictement les mêmes constructeurs, méthodes et opérateurs que dans la version précédente, aucun type de paramètre ne devra être changé).