

## IFT1169 – TRAVAIL PRATIQUE #1 – 26 janvier 2012

## « Accordons nos violons »

Mohamed Lokbani

**Équipes :** le travail est à faire en monôme (une seule personne).

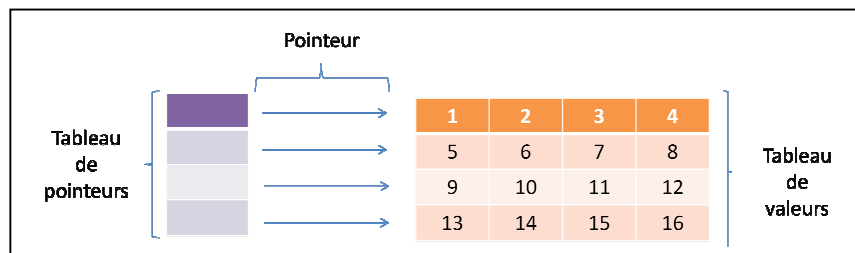
**Remise :** une seule remise est à effectuer par voie électronique le **lundi 06 février 2012, 11h59 au plus tard, sans possibilités de prorogation**.

**Conseils :** n'attendez pas le dernier jour avant la remise pour engager votre travail. Vous n'aurez pas le temps nécessaire pour le réaliser.

**But :** ce TP a pour but de vous faire pratiquer les différents outils de travail utilisés dans l'enseignement du cours.

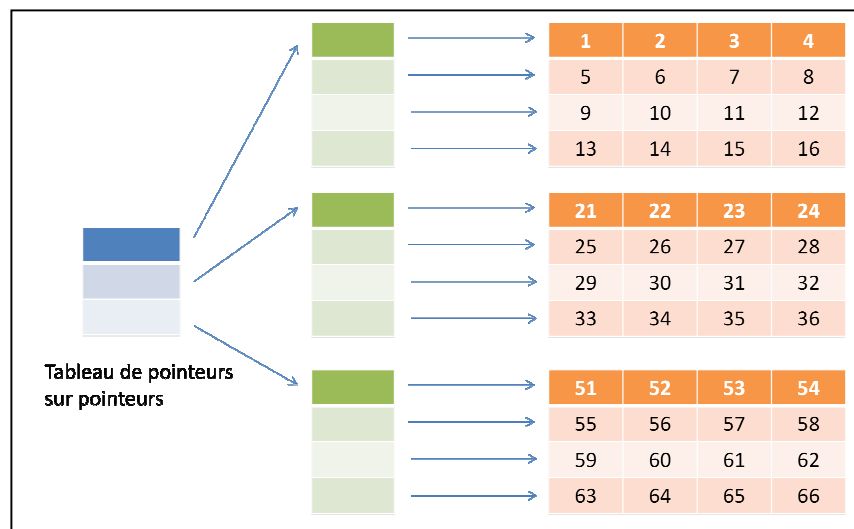
**Énoncé :** soit le tableau d'entiers de taille N. Si « XTab » est le nom associé à ce tableau, sa représentation statique est sous la forme « int XTab[N] ». La valeur de N doit être connue dès le départ afin que le compilateur puisse réserver l'espace mémoire nécessaire. Ce même tableau peut-être aussi alloué de manière dynamique comme suit « int\* ptr = new int[N] ; ». Ainsi, nous avons défini la variable ptr, un pointeur sur un tableau d'entiers de taille N. Ces éléments sont stockés dynamiquement dans une zone mémoire donnée. Cet exercice a pour but de généraliser cette notion à des tableaux 2D et 3D.

Un tableau bidimensionnel est en réalité un tableau de pointeurs à une dimension. Chacun de ces pointeurs va pointer un autre tableau, lui aussi, en a une dimension. Un exemple d'un tel tableau est le 2<sup>e</sup> argument de la fonction « main », « char\*\* argv » ou « char\* argv[] ». La représentation d'un tel tableau bidimensionnel est comme suit :



Vous devez donc allouer d'abord le premier tableau, celui devant contenir les pointeurs. Il représente en réalité le nombre de lignes de votre tableau 2D. Par la suite, vous allez allouer le second tableau, celui qui va contenir les valeurs. Il représente un multiple du nombre de colonne.

Le même principe peut-être généralisé pour un tableau multidimensionnel. Nous allons examiner le cas d'un tableau 3D :



Vous allez d'abord allouer un premier tableau afin de stocker des pointeurs de pointeurs. Chacun de ces pointeurs va pointer un tableau bidimensionnel.

Vous devez définir les fonctionnalités nécessaires pour que la fonction « main » définie en annexe -1- puisse s'exécuter correctement.

**Attention :** si votre classe n'est pas canonique, vous devez prévoir toutes les méthodes nécessaires à son bon fonctionnement de la classe. Nous allons vérifier cela lors de la correction.

**Fichiers à remettre :** vous devez remettre les fichiers nécessaires pour une compilation et une exécution correctes de vos programmes. Vous devez inclure aussi un fichier « makefile » permettant d'automatiser la compilation de vos fichiers. Par ailleurs, vous devez inclure dans votre remise un rapport décrivant le travail effectué. Ce rapport doit porter le nom « tp1H12.pdf ».

**Remise :** il est important de noter que votre TP sera compilé avec « gcc4.6 ». Si par choix vous décidez d'utiliser un autre compilateur, vérifiez que le code que vous avez produit (devant normalement fonctionner correctement chez vous) fonctionne aussi bien sur les ordinateurs de la DESI. Pour avoir la version du compilateur, utilisez la commande "gcc -v" qui devra donner le numéro de version « 4.6 ».

Par ailleurs, assurez-vous de la présence de l'option « pedantic » sur la ligne de compilation.

Pour éviter d'omettre l'un des fichiers associés à ce travail lors de la remise, nous vous conseillons de regrouper l'ensemble des fichiers dans un seul fichier compressé **tp2h11.zip** et de remettre ce dernier comme suit :

1. commencez d'abord par vous connecter sur la machine « **remise** » comme cela a été pratiqué dans la démo #01.
2. envoyez l'ensemble de vos fichiers par la procédure de remise électronique habituelle (Pour obtenir de l'aide sur cette commande, tapez dans un Xterm : man remise). Respectez les noms des fichiers.

**remise ift1169 1169tp1 tp1h12.zip**

3. vérifiez que la remise s'est effectuée correctement.

**remise -v ift1169 1169tp1**

**Barème :** ce TP1 est noté sur **5 points**. Les points sont répartis comme suit :

**En plus du précédent barème, vous risquez de perdre des points dans les cas suivants ....**

- La non remise électronique (volontaire ou par erreur) est sanctionnée par la note 0.
- Les programmes ne contenant pas d'en-tête, -1 point.
- Un programme qui ne compile pas : 0.
- Un programme qui compile mais ne réalise pas les choses prévues dans la spécification : 0.
- Les avertissements (warnings) non corrigés : cela dépend de la quantité! À partir de -0.25 et plus.
- Le non respect du nom du fichier va générer une erreur de compilation donc un des points de la spécification n'a pas été respecté : 0.
- Aberration dans le codage : même si tous les chemins mènent à Rome, faites l'effort nécessaire pour éviter de prendre le plus long!

### **Des questions à propos de ce TP?**

Une seule adresse : [dift1169@iro.umontreal.ca](mailto:dift1169@iro.umontreal.ca)

Pour faciliter le traitement de votre requête, inclure dans le sujet de votre email, au moins la chaîne: [IFT1169] et une référence au **tp01**.

### **Mise à jour**

26-01-2012 diffusion de l'énoncé.

```
int main(){

    Tab2D x1(4,6);    // Tableau bidimensionnel de 4 lignes par 6 colonnes
                     // Il est initialisé avec une valeur entière égale à 0.

    Tab3D y1(2,3,5); // Tableau tridimensionnel de 2lignes, 3 colonnes et une profondeur de 5
                     // Il est initialisé avec une valeur entière égale à 0

    x1.affiche();     // format ligne par ligne comme dans le dessin sur l'énoncé
    x1.set(3,5,10);   // x1[3][5]=10
    x1.get(3,5);       // On affiche l'élément dont l'index est (3,5)
    x1.affiche();     // format ligne par ligne comme dans le dessin sur l'énoncé

    y1.affiche();     // format ligne par ligne comme dans le dessin sur l'énoncé
    y1.set(1,2,3,10); // y1[1][2][3]=10
    y1.get(1,2,3);     // On affiche l'élément dont l'index est (3,5)
    y1.affiche();     // format ligne par ligne comme dans le dessin sur l'énoncé

    return 0;
}
```

**Annexe -1- la méthode « main »**