TD 7 : Tableaux à deux dimensions (tableaux 2D)

Notes pour les enseignants : A l'approche des vacances, il est fort possible que l'on constate un certaine démotivation/fatigue/desintérêt lors des séances de cette semaine. Si c'est le cas, leur dire pour les motiver que les tableaux2D sont la base essentielle des sujets de projets qui vont arriver. Ne pas hésiter aussi à leur dire dès le départ que les tableaux2D peuvent représenter beaucoup d'objets (images, photos, écrans d'ordi, écran de smartphone....), ça les motive un peu ! Pour ceux qui ont vraiment du mal, ne pas hésiter à aller les voir et les encourager à bien reprendre chez eux les TD/TP des semaines précédentes. Leur dire que tout n'est pas perdu mais il va falloir bosser.

Exercice 1 (Échauffement)**.**
On considère le tableau d'entiers à deux dimensions suivant :

(1) Quelles sont les valeurs `tab2d[0][0]`, `tab2d[0][1]`, `tab2d[2][3]` ?

(2) Donner le nombre de lignes et de colonnes du tableau `tab2d`.

Correction : Les valeurs sont : 7, -1, 25. Le tableau a trois lignes et quatre colonnes.

Exercice 2 (Déclaration, allocation et initialisation de tableau 2D)**.**
Écrire un programme qui crée un tableau 2D de L lignes et C colonnes et qui l'initialise avec un entier v dans chaque case.

Correction :

```
#include <iostream>
#include <vector>
using namespace std;
/** Infrastructure minimale de test **/
#define CHECK(test) if ( !(test)) cerr << "Test failed in file " << __FILE__ << " li

/** Construit un tableau 2D L x C dont les valeurs sont initialisées à v
 * @param L un entier: le nombre de lignes
 * @param C un entier: le nombre de colonnes
 * @param v un entier pour initialiser les valeurs de chaque case
 * @return le tableau 2D d'entiers
 **/
vector<vector<int>> tableau2DInitialise(int L, int C, int v) {
    vector<vector<int>> resultat;              //Définition
    resultat = vector<vector<int>>(L);         //Allocation
    for ( int i = 0; i < resultat.size(); i++ ) {
        resultat[i] = vector<int>(C);    //Allocation des sous-tableaux
    }
    for ( int i = 0; i < resultat.size(); i++ ) { //initialisation
        for ( int j = 0; j < resultat[i].size(); j++ ) {
            resultat[i][j] = v;
        }
    }
    return resultat;
}
int main() {
    cout << "Lancement des tests. Si tout marche bien," ;
    cout << " rien ne devrait s'afficher ci dessous." << endl;
```