Tutorial n°3

Preliminaries : Create and add to your project two files called **Labs3.h** and **Labs3.cpp**.

1 Pointers as arrays

1. Declare then implement a function $DisplayPointerInfo(\dots)$ which displays on screen the address of the first element of an array represented by a pointer, then all the values, if multiple, of this array.
2. To test this function, declare two integer pointers *a* and *b* to dynamically allocate arrays of integers for *n* elements (*n* should be asked to the user). Array *a* will be filled with even numbers, and array *b* will be filled with odd numbers.

2 Swapping values of arrays represented by pointers

1. Declare the implement a function $swap1(\dots)$ that swaps two variables represented by pointers.
2. Declare then implement a function $SwapArray(\dots)$ that swaps all the values of two arrays represented by pointers. Use Exercice 1 to display the results.

3 Allocation and deallocation of monodimensional and bidimensional arrays represented by pointers

1. Declare then implement a function $CreateArray(\dots)$ that returns a pointer to an array of *n* integers.
2. Declare then implement a function $DeleteArray(\dots)$ that takes and deletes an array of integers.
3. Declare then implement a function $CreateMatrix(\dots)$ that returns a pointer to an array of arrays of $n \times m$ floats.
4. Declare then implement a function $Deletematrix(\dots)$ that takes and deletes this kind of matrix.
5. Declare then implement a function $DisplayMatrix(\dots)$ that displays the address of the matrix of floats in the memory and all its elements.

In the main procedure, try the following: create an array or a matrix then display it immediatly. What do you remark?

4 A little bit of geometry

Declare and implement functions to compute the dot product (in 3*D* at least, *nD* suggested) and the inner product (in 3D).

5 Matrix multiplication in the general case

Declare and implement a function $MatrixProduct(\dots)$ that returns the matricial product of two matrices of arbitraty dimensions. What are the differences with the exercice with static arrays?

6 Pointers arithmetic

You may have used indexes and the offset operator in the previous exercices. Redo them, **WITHOUT** using indexes and offset operator [] (when possible).