

Tutorial n°8 - STL

The goal of this tutorial is to use the STL vector class. **Please note that you should use iterators to manipulate these containers each time it is possible!** The length of this lab has been strongly shortened since some of the materials have been studied in class (Meshes and data structures using STL).

The key idea here is to implement a template class *CMatrix* to handle most basic linear algebra operations. It should be understood that **implementing your own matrix class is a very bad idea** since there exist very efficient libraries which will do the job much better than you in few hours of coding. Actually, creating your own library can be extremely time-consuming so you should always wonder if it would not be better to use an existing library.

However, through this exercise, you are given a good opportunity to apply several object-oriented concepts, such as operators overloads, work with template classes, and apply advanced C++ programming techniques.

Next follow the tasks for this lab:

1. Three constructors will be provided, one, without argument, that initializes an empty matrix, one that will generate an identity matrix of a given size and finally, one that will generate a rectangular matrix $m \times n$ with random coefficients.
2. Provide a function that allows the user to manually enter the coefficients of a matrix. Also overload `>>` operator.
3. Add a display function that displays the matrix coefficients. Also overload `<<` operator.
4. Provide a function to transpose a matrix.
5. Overload the standard operators `+`, `-`, `*` for *CMatrix*. Make sure to check if the dimensions of the two matrices match for a given operation before performing it.
6. Overload the operator `==` that will compare 2 matrices.
7. Provide a function that computes the trace of the matrix.