02393 C++ Programming Exercises

Week 7, October 10, 2016

Hand-in via `https://dtu.codejudge.net/`, before October 24, 5pm – **Note this is the day after the fall holidays.**

Two-dimensional Vector Space. While in C++ the term "vector" refers to an array-like data structure, in physics and maths it is usually a representation for an n-dimensional space. We consider here two-dimensional vectors in that sense and use the mathematical notation $\binom{x}{y}$ where in this exercise, x and y are numbers of type `double`. The goal of the exercise is to implement a class for such two dimensional vectors with the usual mathematical operations on vectors.

You are given you a header-file `vector2d.h` (see CampusNet), that contains the declaration of the class `v2d` for vectors in two-dimensional space. Moreover we provide a program `main.cpp` file that uses the `v2d` class that you can use to do some tests.

You should implement the class in a file `vector2d.cpp` as follows:

1. The constructor `v2d(double a, double b)` should build a vector $\binom{x}{y}$.
2. The constructor `v2d(const v2d & v)` is meant to build a vector that is exactly like vector `v`.
3. The destructor `~v2d()` does not need to do anything.
4. The assignment operator `v2d & operator=(const v2d &v)` updates a vector to make it exactly like vector `v`.
5. The vector addition method `v2d & operator+(const v2d &v)` updates a vector by adding another vector `v` to it. Recall that
$$\binom{x_1}{y_1} + \binom{x_2}{y_2} = \binom{x_1 + x_2}{y_1 + y_2}$$
6. The scalar multiplication method `v2d & operator*(double k)` updates a vector by multiplying it by a scalar factor `k`. Recall that scalar multiplication is defined by:
$$k \cdot \binom{x}{y} = \binom{k \cdot x}{k \cdot y}$$
7. The scalar product method `double operator*(const v2d &v)` multiplies a vector by another vector `v` and returns the result. Recall that the scalar product of vectors is defined by
$$\binom{x_1}{y_1} * \binom{x_2}{y_2} = x_1 \cdot x_2 + y_1 \cdot y_2$$
8. Method `double length()` computes the length of the vector. Recall that the length of a vector $\binom{u}{v}$ is $\sqrt{v * v}$.

Hints

- You need to upload your file `vector2d.cpp` only, and not the rest of the files.
- Most of the *operator* methods you need to implement (`=`, `*`, etc.) do not need to generate a new vector, but change the vector for which the method was called. For example, if `u` and `v` are vectors then `u + v` will update `u` (with the addition of `u` and `v`).
- Most of the *operator* methods you need to implement (`=`, `*`, etc.) need to return the very same vector (by reference). One of the consequences is that `(u + v) + w` will have the effect of updating `u` (with the addition of `u`, `v` and `v`). Recall that the object for which the object was invoked is accessed with `*this`. Many of your methods, hence, will need to finish with `return *this;` which returns the current object by reference (not a pointer!).

Challenge. Can you use templates to make class parametric with respect to the datatype of the elements (e.g. `double`, `float`, etc.)? Can you use templates to generalize the class to arbitrary dimentions (n-dimensional vectors)?