02393 C++ Programming Exercises

Week 8, April 4, 2016

Hand-in via `https://dtu.codejudge.net/02393-f16/`, before April 11, 5pm

Fun with shapes The goal of the exercise is to implement a family of classes of two-dimensional shapes. You should implement a file `shapes.h` and a file `shapes.cpp` and upload them into CodeJudge. Our test cases will use those files (see also the example in class).

The following classes must be implemented: `Shape`, `Rectangle`, `Square` and `Circle`. There are several possibilities for the design of the classes. Try to think what are the possibilities for inheritance relations among the classes and how to reduce the amount of code. Further hints are provided in the test programs and in the related live code example.

The classes should provide some common methods:

- Constructors: `Rectangle` objects are constructed from two doubles, that specify height and width, respectively; `Square` objects are constructed from one double, that specifies the length of the sides; `Circle` objects are constructed from one double, that specifies the radius.
- `area` should return the area of the shape;
- `perimeter` should return the perimeter of the shape;
- `height` should return the height of the shape;
- `width` should return the width of the shape;
- `rotate` should have the effect of rotating a shape by 90 degrees. This may have no effect on some shapes;

Further hints can be deduced from the test programs and from the example used in class.

Challenge. Putting two rectangles of the same height side-by-side yields a new rectangle (possibly a square one). How would you implement such a functionality? Also, you could consider the exam example in CampusNet, which contains a related exercise in which the goal is to implement a class of regular polygons.