ANKARA UNIVERSITY
COM102B
Fall 2017-18 (Spring)
Programming Assignment 3

Submission Deadline: 29/04/2018, 23:55

In this assignment, you will implement a Rectangle class with some overloaded operators.

Each rectangle is represented with x and y Cartesian Coordinates and with the width and height parameters (Figure 1), as follows:

<x-coordinate> <y-coordinate> <width> <height>

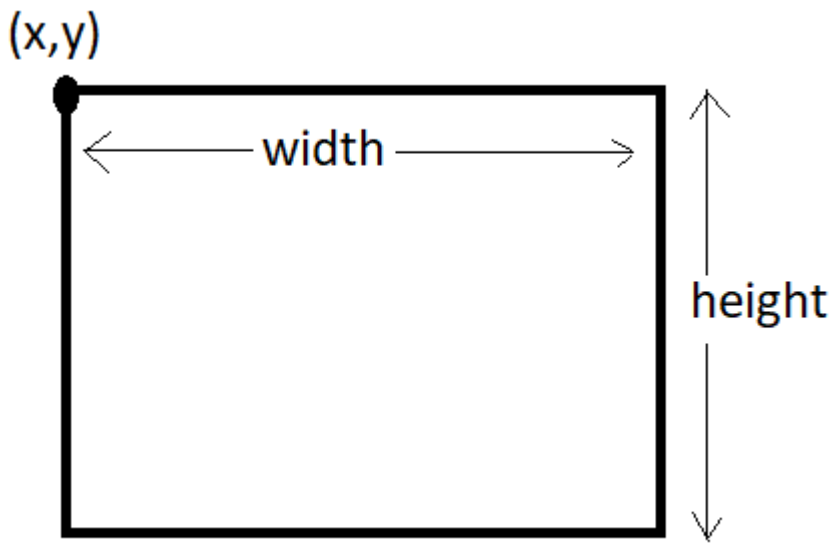

Figure 1: Rectangle object representation in the Cartesian Coordinates.

You will implement three operators:

- **operator&** : Returns the intersection area of two rectangles (Figure 2-left column)
- **operator|** : Returns the union area of two rectangles (Figure 2-middle column)
- **operator^**: Returns the bounding box area of two rectangles (Figure 2-right column)

Moreover, you need to overload **operator<<** and **operator>>** to read from the std. input and to write to std. output, respectively.

We provide you a sample Main.cpp file that will help you to test your Rectangle class implementation. There are 4 test scenarios. Please analyze the Main.cpp file carefully, and as suggested there, follow an incremental development approach.

Compilation:

Create Rectangle.h and Rectangle.cpp files that defines the necessary members and functions related to this class. Given Main.cpp file, compile your source code as follows:

>g++ Rectangle.cpp Main.cpp -o PA3

You can test this executable as explained in the next section.

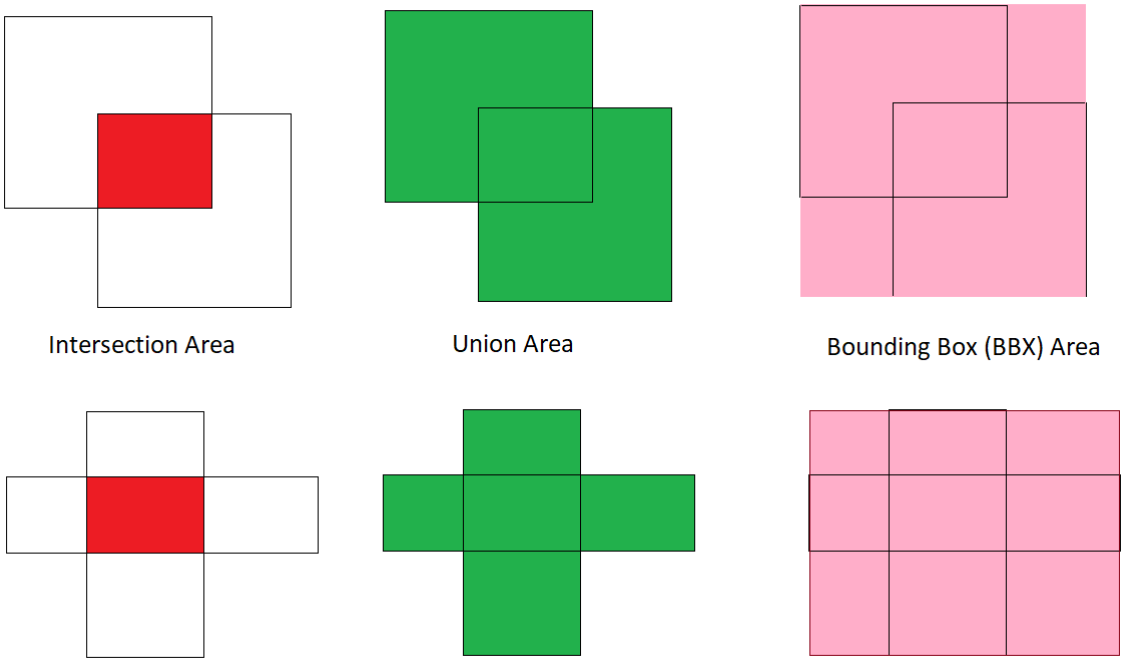

Figure 2: Intersection, Union and BBX areas are depicted for two exemplary cases. There can be many different alternatives.

Testing:

We provide sample input/output text files for you to test your codes at Ubuntu. These text files look in a different format in Windows. If you want to see and use them in Windows operating system, you need to change the line endings to Windows. You can do this easily on Windows by using Notepad++ program, using *Edit/EOF Conversion Menu*.

We recommend you to use *input redirection* mechanism of your operating system to test your programs. For example, if your executable is called as **PA3**, redirect the **input.txt** file to standard input using **<** operator and redirect your outputs to a file using **>** operator such as:

> ./PA3<input.txt>output.txt

This kind of execution enables your programs to read inputs from a file without writing any file related functions. In other words, **cin** reads data from the redirected files instead of the std. input in this way (e.g. keyboard).

Submission: Submit Rectangle.h and Rectangle.cpp files, archived as <student_id>.rar file. Ex: if your student id is 112600, you will send a rar file named as: 112600.rar

Warning: Any form of code copying, including the copies from the internet, is strictly prohibited. If we determine similarities between your codes with any other students in the class, it will be treated as cheating and will be punished. So, you would increase the risk of cheating when you see somebody else's code directly or see a solution in the internet (i.e. somebody else might have also copied the same code from the internet like you, so both of these codes will be evaluated as copies, since they both copy from an external source. Such attempts will always be considered as cheating). You are supposed to write the solution by yourselves.

Please test your programs with the given I/O files before submission.

Ask any questions related with the homework specs. to the course news forum in Moodle.
Follow further announcements about this homework from Moodle.

have fun ☺