Laboratorium 2

1. Create a class named `MyString` which publicly inherits from the `string`. The class should contain:
    a. a constructor that takes `const char*` argument and prints on the screen the string "*MyString constructor called*". Assign a default value to the constructor argument.
    b. a destructor which prints on the screen the string "*MyString destructor called*"
    c. overloaded output operator
2. Create a class named `Figure`. The class should consist of:
    a. private floating point fields (e. g. `m_x, m_y`) that represent the figure location on a plane and the field named `m_label` of `MyString` type,
    b. a constructor that takes two double arguments `x` and `y` and `label` argument of type `const char*` and prints on the screen the string "*Figure constructor called*". Assign default values to all constructor arguments.
    c. destructor which prints on the screen the string "*Figure destructor called*",
    d. getter methods for the private fields.
    e. `void print(void)` method that prints on the screen the string "*I'm a Figure*" as well as the location and the label of the `Figure` object.
3. Create a class named `Rectangle` that publicly inherits from the `Figure` class. The class should have the following fields:
    a. private floating point fields (e. g. `m_w, m_h`) that represent rectangle width and height,
    b. a constructor that takes four double arguments `x`, `y`, `w`, `h` and `label` of type `const char*` and prints on the screen the string "*Rectangle constructor called*". Assign default values to all constructor arguments.
    c. a destructor that prints on the screen the string "*Rectangle destructor called*",
    d. getter methods for the private fields.
4. Create a class named `Square` which publically inherits from the `Rectangle` class. The class should implement:
    a. a constructor that takes three double arguments `x`, `y`, `w`, and `label` of type `const char*` and prints on the screen the string "*Square constructor called*". Assign default values to all constructor arguments.
    b. destructor that prints on the screen the string "*Square destructor called*"
5. In the `main` function do:
    a. Create `Figure` object using default constructor, run the program and notice the order each constructor and destructor is being called.
    b. Create `Rectangle` object using default constructor, run the program and notice the order each constructor and destructor is being called.
    c. Create `Square` object using default constructor, run the program and notice the order each constructor and destructor is being called.
    d. Create `Square` object with any arguments (e. g. x=1.0, y=2.0, w=10.0 and label "*square 1*")
    e. Call the `print` method from the `Square` object. Notice the function being called.

f. Add the `void print(void)` method to the `Rectangle` class that prints on the screen the string "*I'm a Rectangle*" as well as the location and the label of the `Rectangle` object and run the program. Notice the function being called.
g. Add the `void print(void)` method to the `Square` class that prints on the screen the string "*I'm a Square*" as well as the location and the label of the `Square` object and run the program. Notice the function being called.
h. Add the `MyString& to_upper()` method to the `MyString` class which converts all characters in the `MyString` object to uppercase. Call the method in all above `print` functions.

6. Suppose we want to create a class names `Circle`. Which class (`Figure`, `Rectangle` or `Square`) should `Circle` inherit from?