## Lab work #6

**Topics**

* classes
* attributes / member functions
* constructors / destructors
* encapsulation / composition
* const and static

**Exercises**

1. Create a class called Time that has separate int member data for hours, minutes, and seconds. One constructor should initialize this data to 0, and another should initialize it to fixed values. You should also implement the copy constructor.

Implement the insertion operator in order to "display" the time in the 11:59:59 format. Implement also the sum and subtraction operators to work with this type of objects. Include also member functions to act as setters and getters.

You should then implement a program to test your class. You should create at least two initialized Time objects and one that isn’t initialized. Then it should add the two initialized values together, leaving the result in the third time variable. Finally it should display the value of all the objects. Make appropriate member functions and arguments const.

1. Implement a program that allows you to manage geometric figures: circle, square and rectangle. Each figure is characterized by two mandatory attributes: color (string), centre in two-dimensional coordinate system (2D) and additional attributes that specify the dimensions of the figure (radius to circle, two sides to rectangle, one side to square). Take into account the concepts of encapsulation and visibility. Consider the following specification:
   1. Create a class to describe each of the figures;
   2. Use the class Point as an attribute of the classes that implement the various geometric figures (composition);
   3. Take into account the concepts of encapsulation and visibility (ex. For the class Point)

class Point {

private:

double x;

double y;

// .....

};

* 1. Implement several constructors:

Circle(double x, double y, double r) {…}

Circle(Point centre, double r) {…}

// The same for the other figures…

Point(double x, double y) {…}

* 1. add all the important special methods (get…(), set…(), ...);
  2. implement a method to determine the area of each kind of figure;
  3. implement a method to determine the perimeter of each kind of figure;
  4. implement a method to verify if two circles intersect;
  5. implement a program to test all the created classes.

1. It is intended to implement a simplified information system for the management of a university library. The library contains a book catalog and a set of users (students only). All users are identified by their university number (nMec), name and course. The books are characterized by an ID (numeric and sequential, starting at 100), title and type of loan (CONDITIONAL or NORMAL). Start with the following settings:

class User {

private:

String name;

String course;

int nMec;

// .....}

class Book{

private:

int id;

String title;

String type;

// .....}

Make use of access modifiers to ensure that all class attributes are not accessible from outside. If necessary, define new attributes to meet the requirements of the specification.

Using the developed classes, implement a program that, in an interactive way (using a menu), allows managing the library, users and loans. Each student can only request a maximum of 3 books simultaneously. You must modify the user class to be able to store the IDs of the requested books, as well as the class book to indicate their availability.

The program must allow the enrollment and removal of users. For books, the program should allow the introduction of books, make a loan and allow the return of books. For simplicity, consider that there is only one copy of each book and that books with a CONDITIONAL loan type can not be ordered.

Implement a program that interactively (using a menu) allows you to manage the library, users and loans. The program must allow the enrollment and removal of users. For books, the program should allow the introduction of books in the system, check availability of a certain item, loan and return of books. It is recommended that loan and return operations be performed based on the book ID and the student's number.

OPTIONS:

1 – enrol student

2 - remove student

3 - print list of users

4 - register new book

5 - print list of books

6 - loan

7 - return

8 - exit