# Report Lab 4 OOP

Marc Sirvent and Norbert Tomàs

**CircularRegion (extends EllipsoidalRegion)**

The class CircularRegion has the following methods:

**CircularRegion()** which, given a point and a radius, calls the constructor of the super class (EllipsoidalRegion ) with that point and radius.

**double getArea()** which calls getArea of EllipsoidalRegion and returns that value.

**Color**

The class color stores 3 doubles (x, y, z) in which it will store the values of the color depending on the color representation we are using. It also stores a boolean to check if the representation of the color is RGB.

The class Color has the following methods:

**void toHSV()** which does the conversion from the values of a color in RGB to the HSV system.

**void toRGB()** which does the same as toHSV but in the opposite way.

**DrawApp**

The class DrawApp stores 2 lists of entities, one with all the entities and the other one just with the selected ones.

The class DrawApp has the following methods:

**void addEntity()** which adds an entity to the list of entities.

**void draw()** which iterates through all the entities and draws them.

**List<Entity> selected()** which, given a point, selects the entities that have that point inside and returns them in a list.

**boolean isSelected()** which checks if an entity is selected. If so it returns true, else it returns false.

**void selection()** which iterates through all the entities to select them and store them in the list of selected entities (attribute).

**void move()** which iterates through all the entities and calls the function move in each of them with the vector given as an input.

**EllipsoidalRegion (extends Region)**

The class is similar to the one in the previous lab but it implements some new functions such as:

**void move()** which calls the method move in the point class with the vector given as an input.

**boolean isSelected()** which given a point it checks if the class is selected and returns true, else it returns false

**abstract Entity**

This class stores the color of the lines which is of type java.awt.Color. Since this class is abstract it only implements the following methods:

**void setColor()** which stores the color of the lines.

**void setColor()** which returns the attribute lineColor.

**void rotate()** which given an angle it rotates the entities.

**Line (extends Entity)**

The class stores two points.

The class DrawApp has the following methods:

**void draw()** which it calls the method drawLine of the graphics class with the points of the line.

**boolean isPointClose()** which checks if a given point is close to the line.

**boolean isSelected()** which given a point it checks if the class is selected and returns true, else it returns false

**MyWindow**

This class is the main class. Here we create the window, the entityDrawer and all the entities and it's colors.

**Point**

This class is very similar to the one on the previous lab but it implements two new methods:

**void move()** which, given two doubles, it sums them to the coordinates of the point.

**Vector difference()** which returns a vector which is the difference of two given points.

**PolygonalRegion (extends Region)**

This class is very similar to the one on the previous lab but it implements this two new methods:

**boolean isPointInside()** which, given a point, checks if it's inside the polygonal region by iterating over all the points of the polygon and doing the cross product of that point, the next one and the point given as an input. It will return true if the point is inside and false otherwise.

**void move()** which iterates over all the points of the PolygonalRegion and sums to each one the number of units we want to move the polygon.

**boolean isSelected()** which given a point it checks if the class is selected and returns true, else it returns false.

**RectangularRegion (extends PolygonalRegion)**

This class calls the constructor of the super class giving as an attribute a new ArrayList composed by the points of the rectangle.

**double getArea()** which calls the method getArea of the super class.

**Abstract Region**

This class stores the fillColor of the region and, since it's abstract, it only implements the following methods:

**void setFillColor()** which given a color it stores it to the attribute of the fillColor of the class.

**Text (extends Entity)**

This class stores a String with the text we want to print and the point where we will print it. It has the following methods:

**void draw()** which draws the string by calling the method drawString of the graphics class.

**void move()** which adds a given units to the coordinates of the point.

**boolean isPointClose()** which checks if a given point is close to the text.

**boolean isSelected()** which given a point it checks if the class is selected and returns true, else it returns false

**TriangularRegion (extends PolygonalRegion)**

This class works similar to the RectangularRegion. It creates a new ArrayList with the 3 points of the triangle and also has a function called getArea which returns the area of the triangle.

**Vector**

This class is similar to the class point and is used to represent two points as a vector.

**double crossProduct()** which given two vectors it computes the cross product between them.