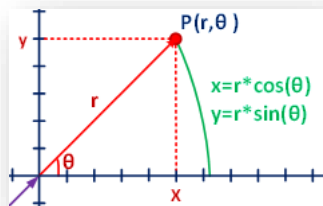


ASSIGNMENT #3 AND #4

SUBJECT & BASIC INFORMATION

➡ WRITE A C# PROGRAM WITH FOLLOWING REQUIREMENTS

- Define a **Point2D** class:
 - ✚ Add data members of cartesian coordinates (**x** and **y**) and related properties for these fields
 - ✚ Define a default constructor with no parameters
 - ✚ Define a constructor setting initial 2D coordinates with random **x** and **y** values
 - ✚ Define a **printCoordinates()** method that prints the coordinates of the 2D point
 - ✚ Define a **calculatePolarCoordinates()** method that calculates polar coordinates (**P(r,θ)**) of this 2D point according to the figure below:



$$x^2 + y^2 = r^2$$

$$r = \text{sqrt}(x^2 + y^2)$$

$$\theta = \tan^{-1}(y/x)$$

- ✚ Define a **calculateCartesianCoordinates()** method that calculates cartesian coordinates (**P(x,y)**) of the 2D point (vice verse of converting to polar coordinates)
 - ✚ Define a **printSphericalCoordinates()** method that prints the pre-calculated spherical coordinates of this 2D point.
- Define a **Polygon** class
 - ✚ Add **center** data member composed of **Point2D** class
 - ✚ Add **length** data member and related property for this field
 - ✚ Add **numberOfEdges** data member and related property for this field
 - ✚ Define a default constructor with no parameters
 - ✚ Define a constructor setting initial center and radius with random values
 - ✚ Define a **calculateEdgeCoordinates()** method that calculates the vertex points of the polygon.
 - First vertex should start with a random point calculated depending on the **center** and **length** values.
 - ✚ Define a **rotatePolygon()** method that recalculates the vertex points of the polygon (rotation is done clockwise)
- Create a form interface including these form elements
 - ✚ Two **textBoxes** to enter the **center** of the polygon
 - ✚ A **textBox** to enter the **length** of the polygon
 - ✚ A **textBox** to enter **numberOfEdges** of the polygon (at least 3)
 - ✚ A **pictureBox** to draw the graphics depending on the textboxes

- ✚ A textBox to enter the angle of rotation (the initial value should be zero)
- ✚ A **button** that start drawing graphics
 - create a polygon object depending on the input values
 - call the required functions to (re)calculate the edge coordinates
 - draw the polygon on the pictureBox

RULES & EVALUATION

- ➡ Name of the project should be the student number (without dot)
- ➡ To optimize the size of the assignment folder, the project should be cleaned (to clean your Solution/Project, use **Build-> Clean Solution**)
- ➡ The beginning of all .cs files should include this comment lines below

```
//*****
//**
//**          STUDENT NAME.....:
//**          STUDENT NUMBER.....:
//*****
```

- ➡ There should be comment lines for some operations (methods, specific calculations, etc.)
- ➡ **Deadline:** Control SABIS system
- ➡ A **honor-code page** should be prepared
 - ✚ It should include a cover including student information (name, surname, number, lecturer, course name, ...)
 - ✚ At the end of the page, there should be an '**honor code**' signed by the student.
- ➡ You should upload **your project file and honor-code page (in pdf format)** together before deadline.