Computer Science – Object Programming in C++

Laboratory #11

Your task for today is to build some hierarchical classes, that will represent mathematical expressions and geometric figures.

After implementation of each stage uncomment appropriate section in Main.cpp file to test your code. You will get 0.5 points for each class implementation.

STAGE_1 (3 Points) – In this stage you have to implement class hierarchy representing mathematical expression. The abstract class `Expression` is a base class for other expressions. It contains virtual `Evaluate` method to evaluate the expression and `Info` to display info about the expression. Please take a look at the implementation. There are two types of expression, first one is `SingleExpression`, that represents a single value expression like constants or square roots, the other one is `MultiExpression`, which represents multi expressions that contains left and right value such as addition or multiplication. Take a look at `Constant` class implementation, that derives from `SingleExpression` and represents constant value. Based on this, implement this stage.

Your task is to implement other classes, that you can find in the file (*Expression.h*), accordingly to their naming and usage in Main.cpp file. You have to implement appropriate constructor and pure virtual method derived from base class.

For `MultiExpression` classes constructor should take two pointers to `Expression` class, then initialize internal class values with those arguments. `SingleExpression` should only take float values, you can create fields in the class if needed. Take a look at already implemented classes.

`Evaluate` method should evaluate expression, this mean the `SquareRoot` class will return square root value from given in constructor value or `Multiplication` class will return result of multiplication of two expressions. `Info` should write information about expression. Use parenthesis to indicate the order of the evaluation. To display all expressions till the end use `Info` method on left and right values also.

STAGE_2 (1.5 Points) – This stage consists of implementation of `GeometricFormula`. It is a base class for Area and Volume calculation. Implement all classes that you can see in the file (*GeometricFormula.h*). The only task is to implement their constructors (see usage in Main.cpp file), which should initialize expression (field from a base class) with appropriate calculation of what they should compute (for example the `AreaFormulaCircle` should calculate the area of a circle with given radius). Take a look at sample implementation of `AreaFormulaParallelogram` and `VolumeFormulaPrism` classes. For initialization use expressions implemented in previous stage.

STAGE_3 (3 Points) – In this stage you have to implement `GeometricFigure` hierarchy classes. There are two types of figures, first one `FlatGeometricFigure`, that represents flat figures like rectangle and `SpatialGeometricFigure`, which represents spatial figure such as cubes. Only constructors (see usage in Main.cpp file) and `Info` methods should be implemented. Implement all classes that you can find in file (*GeometricFigure.h*). Constructor should initialize the area (for first type) or area and volume (for second type) fields (using implemented in previous stage formulas) with appropriate values. The `Info`

method should write the name of the class and then call base's class `Info` method. Take a look at example of `Parallelogram` and `Cube` classes implementation.

Take under consideration that `Rectangle` and `Square` classes are not inherited directly from `FlatGeometricFigure` class.

STAGE_4 (0.5 Points) – In this stage you have to implement function `float GetVolumeSumFromSpatialFigures(FlatGeometricFigure** geometricFigures, unsigned short size)` that you can find in *Main.cpp* file. It should return a sum of volumes from all given geometric figures. Remember that only spatial figures has volume. Use RTTI mechanism to recognize which of those figures are spatial figures and then increase a total volume value appropriately.

The sample output from the Application:

```
 ---------------------- STAGE_1 (3Pts) ----------------------
Constant(17) : 17 = 17
Constant(29.5) : 29.5 = 29.5
Addition(Constant (29.5), Constant(8)) : (29.5+8) = 37.5
Subtraction(Constant (51), Constant(15)) : (29.5-8) = 21.5
Multiplication(Constant(11), Constant(4)) : (11*4) = 44
Subtraction(Constant (18), Constant(4)) : (18/4) = 4.5
SquareRoot(144) : sqrt(144) = 12
Power(12) : 12^2 = 144
Multiplication(Addition(Constant(9.5),Constant(0.5)), Subtraction(Constant
(29.5),Constant(28))) : ((9.5+0.5)*(29.5-28)) = 15

 ---------------------- STAGE_2 (1.5Pts) ----------------------

AreaFormulaParallelogram (a = 1; h = 1) : (1*1) = 1
AreaFormulaParallelogram (a = 1; h = 5) : (1*5) = 5
AreaFormulaTriangle (a = 1; h = 5) : ((1/2)*(1*5)) = 2.5
AreaFormulaTriangle (a = 2; h = 1) : ((1/2)*(2*1)) = 1
AreaFormulaCircle (r = 1) : (3.14*1^2) = 3.14
AreaFormulaCircle (r = 5) : (3.14*5^2) = 78.5
VolumeFormulaPrism (Ba = 1; H = 1) : (1*1) = 1
VolumeFormulaPrism (Ba = 78.5; H = 2) : (78.5*2) = 157
VolumeFormulaPyramid (Ba = 2.5; H = 2) = ((1/3)*(2.5*2)) = 1.66667
VolumeFormulaPyramid (Ba = 1; H = 2) = ((1/3)*(1*2)) = 0.666667

 ---------------------- STAGE_3 (3Pts) ----------------------

Parallelogram: Area: (1*5) = 5
Rectangle - Area: (3*5) = 15
Square - Area: (3*3) = 9
Triangle - Area: ((1/2)*(2*2)) = 2
Circle - Area: (3.14*2^2) = 12.56

Cube - Area: (2*2) = 4 Volume: (4*2) = 8
Tetrahedron - Area: ((1/2)*(1*2)) = 1 Volume: ((1/3)*(1*6)) = 2
Cone - Area: (3.14*1^2) = 3.14 Volume: ((1/3)*(3.14*3)) = 3.14
```

```
 ---------------------- STAGE_4 (0.5Pts) ----------------------

Cube - Area: (5.15*5.15) = 26.5225 Volume: (26.5225*5.15) = 136.591
Circle - Area: (3.14*3^2) = 28.26
Square - Area: (4*4) = 16
Cone - Area: (3.14*3^2) = 28.26 Volume: ((1/3)*(28.26*1)) = 9.42
Cone - Area: (3.14*3^2) = 28.26 Volume: ((1/3)*(28.26*7.13)) = 67.1646
Rectangle - Area: (1*4.5) = 4.5
Parallelogram: Area: (1*2.5) = 2.5
Tetrahedron - Area: ((1/2)*(1.5*2)) = 1.5 Volume: ((1/3)*(1.5*3)) = 1.5
Tetrahedron - Area: ((1/2)*(1.5*1)) = 0.75 Volume: ((1/3)*(0.75*5)) = 1.25


Sum Of Volumes From Spatial Figures: 215.925
```