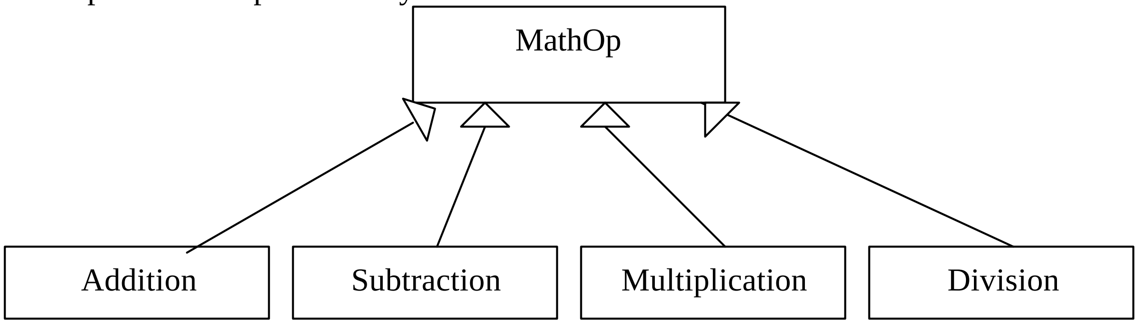
KSU/CCIS CSC113

LAB No. 5 - Abstract Classes and Polymorphism

Date: 24 – 28 /11/2007

Exercise 1:

The following diagram presents the UML hierarchy of mathematical operations where each operation is represented by a class.



Class	Result()	toString()
MathOp: - opName: String + mathOp(String) + toString() + result(): double		opName
Addition: - Op1: double - Op2: double + Addition (double, double) + result(): double	Op1 + Op2	opName “: “ Op1 “ + ” Op2 “ = “ result()
Subtraction: - Op1: double - Op2: double + Subtraction (double, double) + result(): double	Op1 + Op2	opName “: “ Op1 “ - ” Op2 “ = “ result()
Multiplication: - Op1: double - Op2: double + Multiplication (double, double) + result(): double	Op1 * Op2	opName “: “ Op1 “ * ” Op2 “ = “ result()
Division: - Op1: double - Op2: double + Division (double, double) + result(): double	If Op2 != 0 result = Op1 / Op2 else result = -99999.99999	If Op2 == 0 “Error: Divide by zero” Else opName “: “ Op1 “ / ” Op2 “ = “ result()

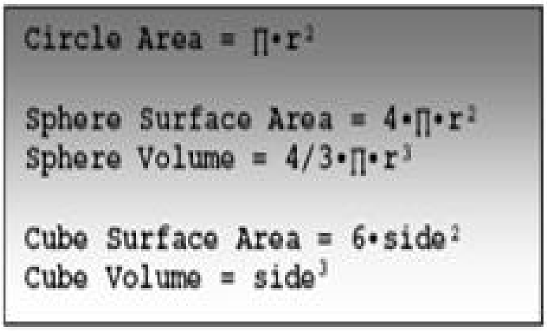
Implement the **mathOp** classes and then write the test class that has **main()** method as follow:

1. Define four objects (one for each non-abstract class).
2. Define an array of **mathOp** consists of all 4 types of objects.
3. Store the objects defined in step 1 into the array defined in step 2.
4. Print the array data **except** the objects of type **Subtraction**.

Exercise 2:

Implement the **Shape** hierarchy shown below. Each **TwoDimensionalShape** should contain method **getArea()** to calculate the area of the two dimensional shape. Each **ThreeDimensionalShape** should have methods **getArea()** and **getVolume()** to calculate the surface area and volume, respectively.

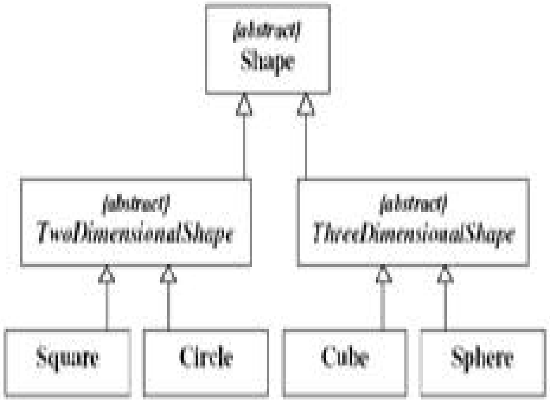
Create a program that uses an **array of Shape** references to object of each concrete class in the hierarchy. The program should **print** a text description of the object to which each array element refers. Also, in the loop that processes all the shapes in the array, **determine** whether each shape is a TwoDimensionalShape or ThreeDimensionalShape. If the shape is TwoDimensionalShape, **display its area.** If a shape is a ThreeDimensionalShape, **display its area and volume.**



Circle Area = ∏•r^{2}

Sphere Surface Area = 4•∏•r^{2}
Sphere Volume = 4/3•∏•r^{3}

Cube Surface Area = 6•side^{2}
Cube Volume = side^{3}



Output

```
part of the main…
shapes[0]= new Circle(4);
shapes[1]= new Cube(2);
shapes[2]= new Sphere(3);
shapes[3]= new Square(7);
shapes[4]= new Circle(1);
shapes[5]= new Sphere(4);
```

Output should be like this:

```
This is a 2D shape.
Its area = 50.26548245743669
-------------------
This is a 3D shape.
Its area = 24.0
Its volume = 8.0
-------------------
This is a 3D shape.
Its area = 113.09733552923255
Its volume = 84.82300164692441
-------------------
This is a 2D shape.
Its area = 49.0
-------------------
This is a 2D shape.
Its area = 3.141592653589793
-------------------
This is a 3D shape.
Its area = 201.06192982974676
Its volume = 201.06192982974676
```