

**MS-3803**  
**Program #3**  
**Inheritance & Polymorphism**

**Date: Wednesday, January 10, 2007**

**Due: Wednesday, January 17, 2007**

The assignment is to implement a text-based graphics program using inheritance relationships and polymorphic behaviors (similar to the one we did in class together, or similar to the one the author does [in chapter 9 of the 5<sup>th</sup> edition of the textbook, or in chapter 8, section 21 of the 6<sup>th</sup> edition]).

We wish to produce a “shapes” taxonomy that would allow for a hierarchy of graphical objects; such as Points, Lines, Circles, Rectangles, Cylinders and Cubes to be “drawn”. In order to keep things simple, our drawing for this assignment will only amount to sending text out to the console (hence the term “Text Graphics” for our assignment).

Arrange these as classes in a hierarchy and associate drawing behavior with each class that is sensible to draw. (The Shapes class, for instance, would be an abstract class since there’s no reasonable way to draw a generic shape.) The drawing behavior for this assignment need only consist of text output to the console that states where the shape is and how it would draw itself (just as we did in the example from class).

In your implementation be sure to use both inheritance and composition where appropriate. Remember that inheritance is called an “IS-A” relationship and composition is called a “HAS-A” relationship. This rule-of-thumb design principle works for most simple taxonomies (such as this).

You’ll also need to use polymorphism in your implementation (i.e. the `draw()` method) so that each shape draws itself appropriately when sent the `draw()` message.

Extra Credit

If you’re able to get this working without much fuss, then to further improve your program, create a container of shapes (choose from an array, ArrayList, Vector, or some other simple container) and iterate through the container, sending each object the `draw()` message to invoke the drawing behavior. This too will be demonstrated in class.