

## Homework Assignment #2

5.1. What's the difference between a component-based architecture and a service-oriented architecture?

**The difference between a component-based architecture and a service-oriented architecture is that component-based sees pieces of the system as loosely grouped components that provide service for each other while service-oriented architectures have pieces that are implemented as services — usually running on separate computers which makes them more separated.**

5.2 Suppose you're building a phone application that lets you play tic-tac-toe against a simple computer opponent. It will display high scores stored on the phone, not in an external database. Which architectures would be most appropriate and why?

**A monolithic architecture would be most appropriate because this is not an application that requires a remote service, and is relatively small. Tic-tac-toe would work well on a single-tiered application that has a combination of both UI and data access. There is no need to go more complicated.**

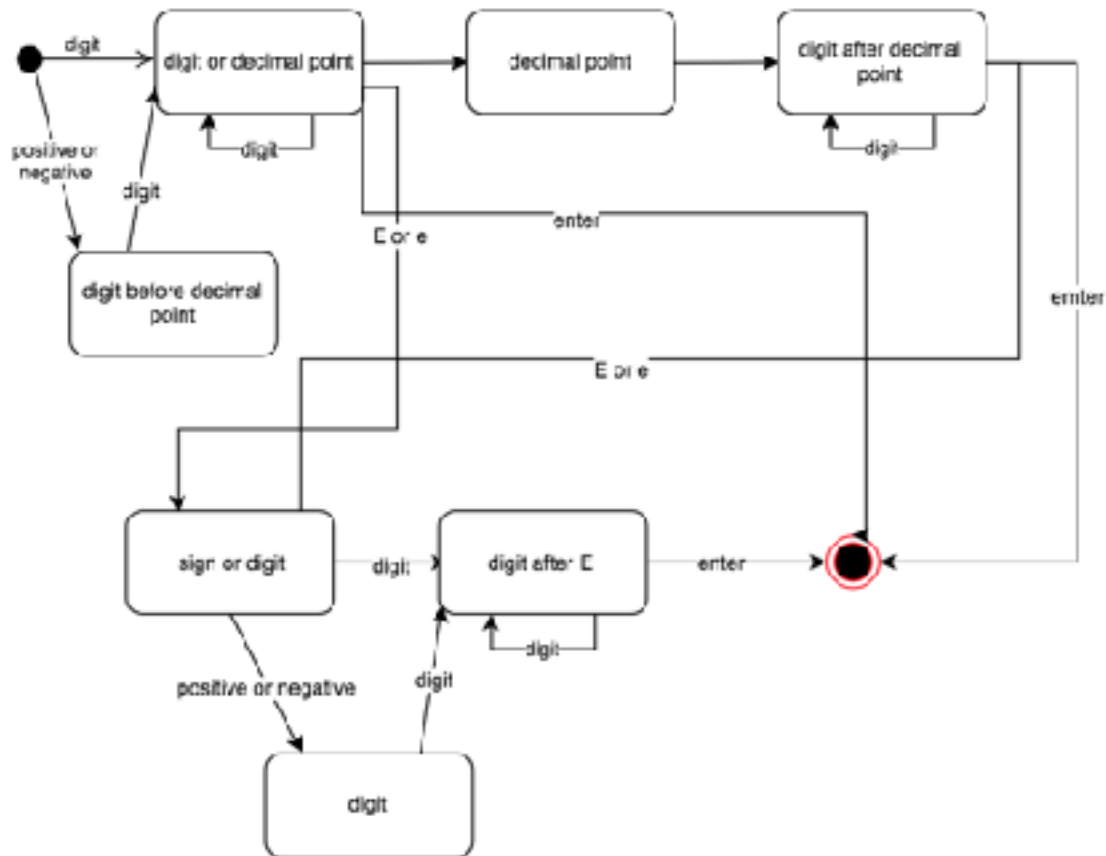
5.4 Repeat question 3 [after thinking about it; it repeats question 2 for a chess game] assuming the chess program lets two users play against each other over an Internet connection.

**A monolithic service-oriented application would be necessary because the UI should all be the same, but the only main difference would be using the Internet to exchange information with another player on a different system.**

5.6 What kind of database structure and maintenance should the **ClassyDraw** application use?

**ClassyDraw does not require a database structure because it can store the drawing in separate files that can be managed by OS tools. In order to restore drawings, ClassyDraw could create temporary files while a user is 'drawing' and restore the file after a crash were to happen.**

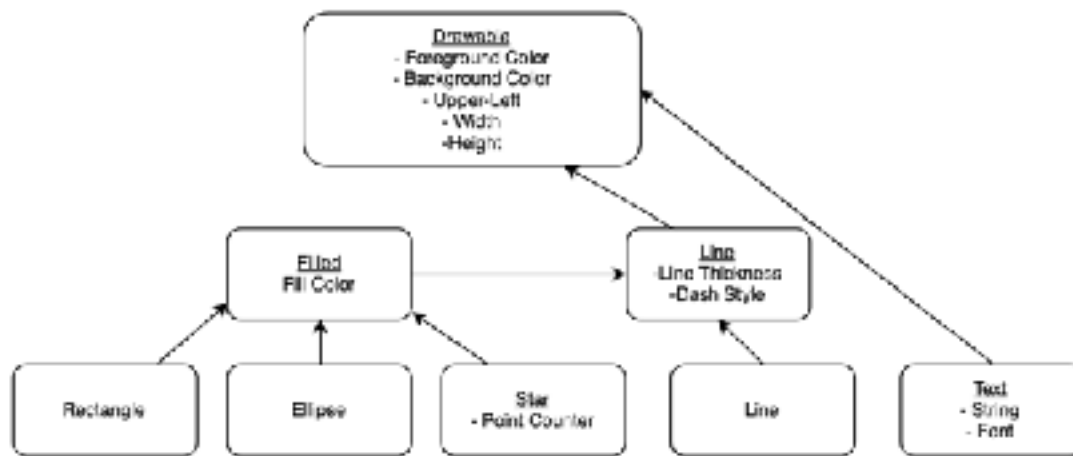
5.8 Draw a state machine diagram to let a program read floating point numbers in scientific notation as in +37 or -12.3e+17 (which means  $-12.3 \times 10^{17}$ ). Allow both E and e for the exponent symbol. [Jeez, is this like Dr. Dorin's DFAs, or *what???*]



6.1 Consider the **ClassyDraw** classes **Line**, **Rectangle**, **Ellipse**, **Star**, and **Text**. What properties do these classes all share? What properties do they not share? Are there any properties shared by some classes and not others? Where should the shared and nonshared properties be implemented?

- These classes all share properties for color (foreground and background), drawing position (upper-left, width, height) — properties needed for drawing.
- Properties that do not share include Text classes that need font information for the string and Star classes that need a point counter to keep track of points.
- Properties shared by some classes and not others include Rectangle, Ellipse, and Star that share a color to be filled AND Rectangle, Ellipse, Star, and Line that include the thickness of the line and dash style.

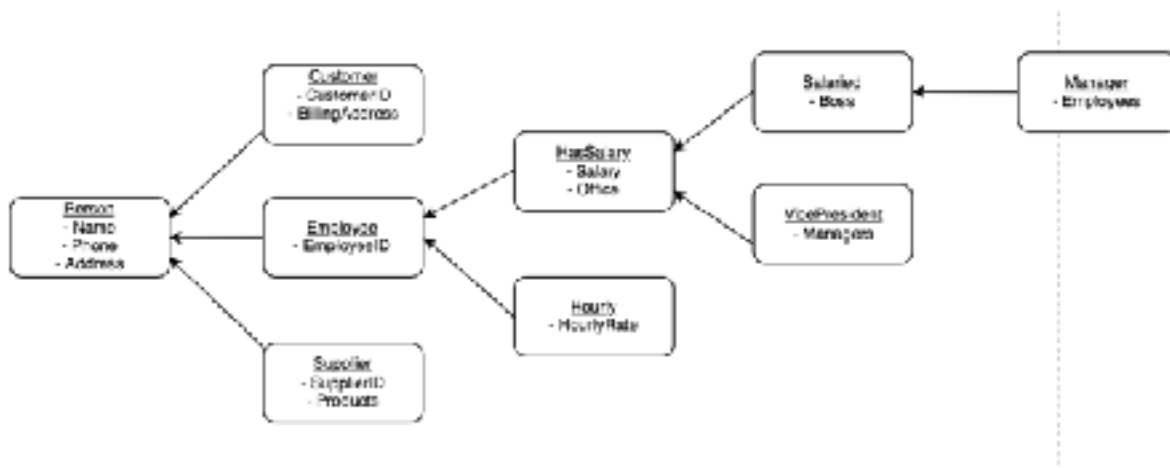
6.2 Draw an inheritance diagram showing the properties you identified for Exercise 1. (Create parent classes as needed, and don't forget the **Drawable** class at the top.)



6.3 The following list gives the properties of several business-oriented classes.

- Customer — Name, Phone, Address, BillingAddress, CustomerID
- Hourly — Name, Phone, Address, EmployeeID, HourlyRate
- Manager — Name, Phone, Address, EmployeeID, Office, Salary, Boss, Employees
- Salaried — Name, Phone, Address, EmployeeID, Office, Salary, Boss
- Supplier — Name, Phone, Address, Products, SupplierID
- VicePresident — Name, Phone, Address, EmployeeID, Office, Salary, Managers

Assuming a **Supplier** is someone who supplies products for your business, draw an inheritance diagram showing the relationships among these classes. (Hint: Add extra classes if necessary.)



6.6 Suppose your company has many managerial types such as department manager, project manager, and division manager. You also have multiple levels of vice president, some of whom report to other manager types. How could you combine the **Salaried, Manager,** and **VicePresident** types you used in Exercise 3? Draw the new inheritance hierarchy.

