Written Homework Assignment #2

Problems from Ch 5 and 6

Problem 5.1, Stephens page 116

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

- A component-based architecture builds an application from reusable software components that are combined within a single system or application.A service-oriented architecture (SOA) builds an application from independent services that communicate over a network, often using web services or APIs.

The key difference is that components are usually internal building blocks of one application, while services are independent systems that can be accessed remotely and reused across multiple applications.

Problem 5.2, Stephens page 116

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?

The most appropriate architecture would be the two tier architecture because the application runs entirely on the phone, high scores are stored locally and there is no need for distributed services or external communication.

Problem 5.4, Stephens page 116

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.

Either a three-tier architecture or a service-oriented architecture because two players must communicate over the internet, game state must be synchronized between devices, and likely requires a server to coordinate moves.

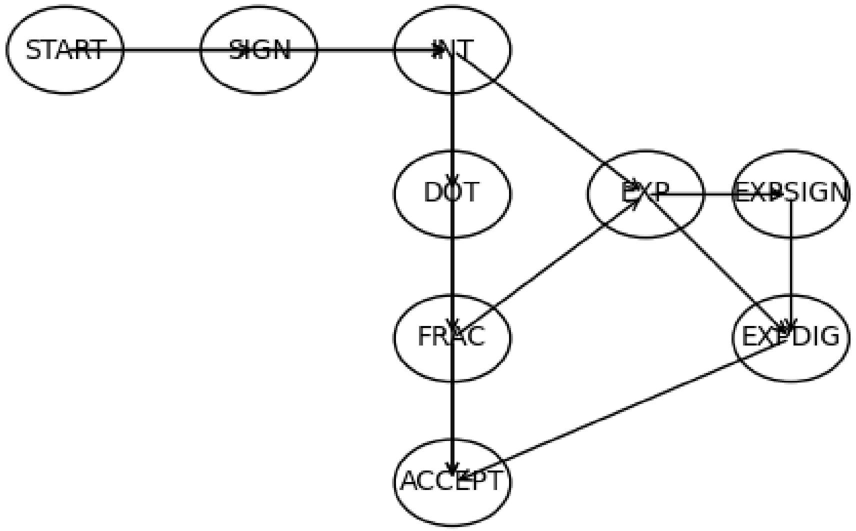
Problem 5.6, Stephens page 116

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

A relational database structure

Problem 5.8, Stephens page 116

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 x 10^{17}). Allow both E and e for the exponent symbol. [Jeez, is this like Dr. Dorin's DFAs, or *what*???]



Problem 6.1, Stephens page 138

Consider the **ClassyDraw** classes **Line**, **Rectangle**, **Ellipse**, **Star**, and **Text**.

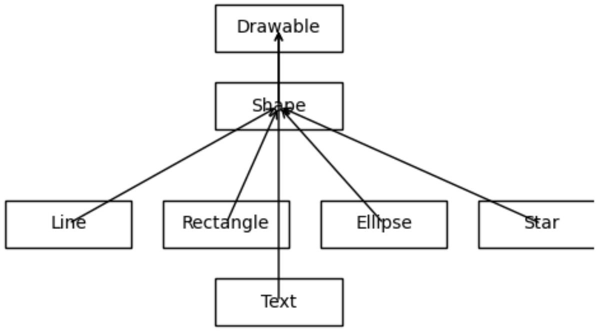
1. What properties do these classes all share?
   a. Position, color, possible the thickness, draw(), move()
2. What properties do they **NOT** share?
   a. Text has font and string content, line has end point, rectangle/Ellipse
3. Are there any properties shared by some classes and not others?
   a. Width/Height → Rectangle, Ellipse, StarFill color → Rectangle, Ellipse, Star (maybe not Line), Text-specific properties → only Text

4. Where should the shared and nonshared properties be implemented?

- Shared properties → in a common parent class (Drawable), Shape-specific shared properties → in an intermediate class (Shape), Unique properties → in individual subclasses

Problem 6.2, Stephens page 138

Draw an inheritance diagram showing the properties you identified for Exercise 6.1. [Create parent classes as needed, and don't forget the `Drawable` class at the top.]


Drawable
Shape
Line
Rectangle
Ellipse
Star
Text