

Homework #2

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

Component-based architecture regards a system as a collection of components that are loosely coupled that provide services for each other. On the other hand, service-oriented architecture differs in that the components are implemented as services which are self-contained programs that run on their own and provide clients with a service.

Problem 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?

Since this is a small application, a monolithic architecture would be most appropriate because the single program does everything from displaying the user interface for the tic-tac-toe game, accessing data, and processing the users moves and scores. Event-driven architecture could also be appropriate because different parts of the system respond to events as they take place while a user plays the tic-tac-toe game.

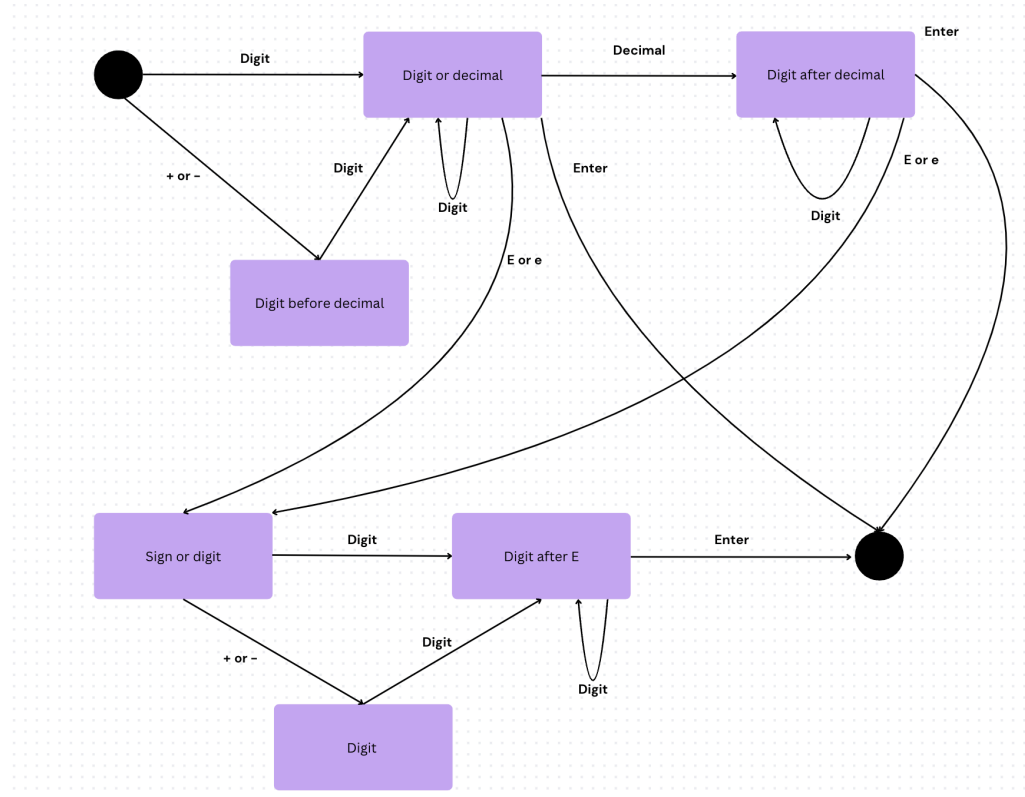
Problem 5.4: Repeat question 3 assuming the chess program lets two users play against each other over an Internet connection.

A client / server architecture would be the most appropriate for this chess program because it could support multiple clients (user interfaces for the two players) with the same server. Distributed architecture could also be useful for improving performance since different parts of the application run on distinct processors that may run simultaneously. This type of architecture allows for processors to be on different computers which may be the case for the two users playing the game of chess.

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

For database structure and maintenance, the ClassyDraw application is able to store drawings in separate files so not much of a database is needed. For this, there are operating system tools that allow a user to make copies of files, delete files, etc. To ensure database maintenance, the application could make a temporary file for when users are drawing that they could be able to restore if they leave the application.

Problem 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. Allow both E and e for the exponent symbol.



Problem 6.1: Consider the ClassyDraw classes Line, Rectangle, Ellipse, Star, and Text.

a. What properties do these classes all share?

All of the classes share the properties of ForeColor, BackColor, UpperLeft, Width, and Height.

b. What properties do they NOT share?

These classes do not share the properties of Font, String, NumPoints, FillColor, LineThickness, and DashStyle.

c. Are there any properties shared by some classes and not others?

The four classes of Rectangle, Ellipse, Star, and Line share the properties of LineThickness and DashStyle, but these properties are not shared by other classes. Rectangle, Ellipse, and Star also share the property of FillColor. Text has the properties of Font and String that are not shared by any other classes. The same is true for Star and

the property of NumPoints. The rest of the properties for the ClassyDraw application are shared by all five of these classes.

- d. Where should the shared and nonshared properties be implemented?

All of the shared properties could be implemented in one main class called 'Shape' since all of the Line, Rectangle, Ellipse, Star, and Text classes share these properties. Something similar could be done with the few classes that share some properties like how Rectangle, Ellipse, and Star share the properties of FillColor, LineThickness, and DashStyle.

Problem 6.2: Draw an inheritance diagram showing the properties you identified for Exercise 6.1.

