Chapter 3.7 Polymorphism Problem Set

# Problem 3.7.1: Chess Board

Consider the following problem:

A chess board consists of a collection of chess pieces. Each chess piece has a value corresponding to its worth (10 points for a Queen, 5 points for a rook, etc.), a letter indicating is type (‘Q’ for queen, ‘R’ for rook, etc.), and rules governing how it moves. In other words, you can ask a piece for a set of moves are possible from its position on the board and the other pieces that reside there.

Create a class diagram describing this chess board class with all the related classes necessary to complete it. For the sake of simplicity, only include three piece types: a pawn, a bishop, and a queen.

# Problem 3.7.2: Shapes

Consider the following problem:

A drawing program contains a collection of drawings, each of which is a type of shape. A shape can be many things: a circle, line, polygon, curve, square, oval, triangle, or diamond. Each shape has an outline color and width. Each shape with a volume can also has a shading color. Of course, each shape has a position and size, though the notion of size will depend on the shape.

Create a class diagram describing the notion of drawings and of shape. Note: only include attributes and methods mentioned or implied in the problem description.

A screenshot of a computer

AI-generated content may be incorrect.

# Problem 3.7.3: Vehicles

Consider the following problem:

A list program allows a user to maintain a collection of vehicles. Each item can be one of the following: a bicycle, car, skateboard, skis, motorcycle, and inline skates. There are several flavors of bicycles (road, mountain, cyclocross, BMX, unicycle, and gravel). There are several flavors of cars (truck, sportscar, passenger, compact, SUV, and convertible). The user will be able to edit each vehicle, add a new instance, and display the contents. Each vehicle will have a variety of attributes specific to the type of vehicle.

Create a class diagram describing this program. Note: only include attributes and methods mentioned or implied in the problem description.