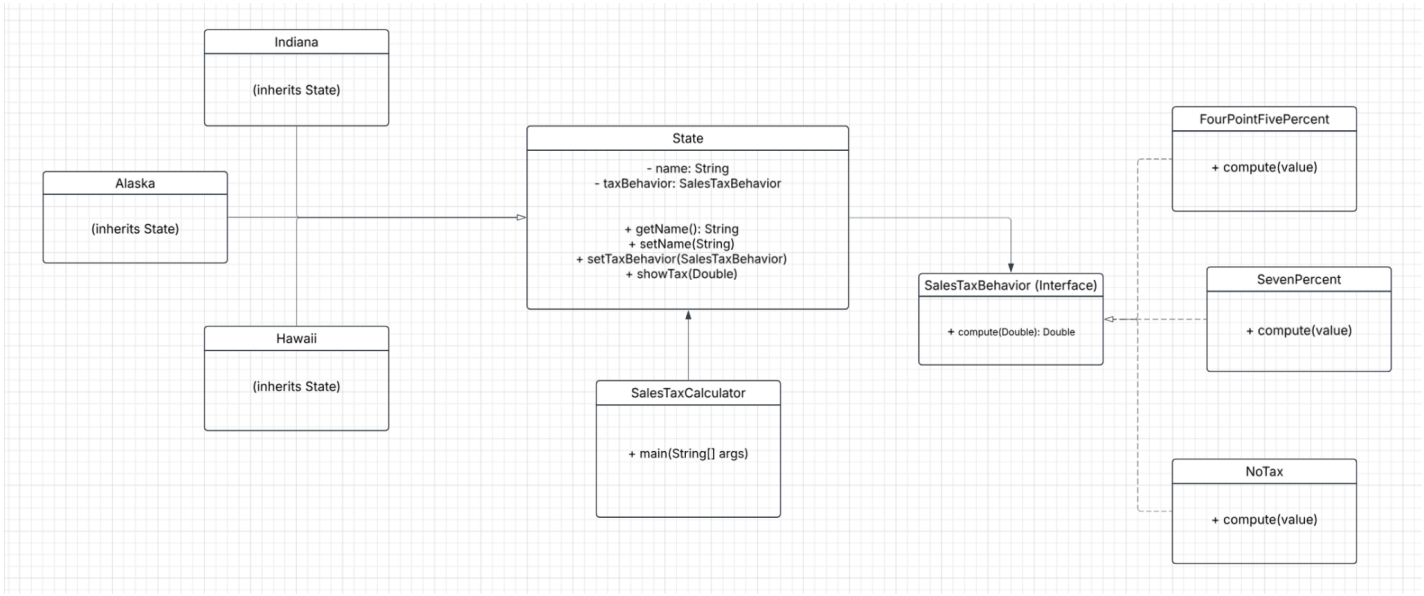
Problem 1 Part B:



Problem 2:

1. MallardDuck to Duck = IS-A, MallardDuck inherits from Duck, meaning it is a type of Duck.

2. RedheadDuck to Duck = IS-A, RedheadDuck extends Duck, inheriting its properties and behaviors.

3. RubberDuck to Duck = IS-A, RubberDuck is a subclass of Duck, following the inheritance hierarchy.

4. DecoyDuck to Duck = IS-A, DecoyDuck inherits from Duck, meaning it is a type of Duck.

5. Duck to FlyBehavior = HAS-A, The Duck class has a FlyBehavior, allowing it to assign different flying behaviors.

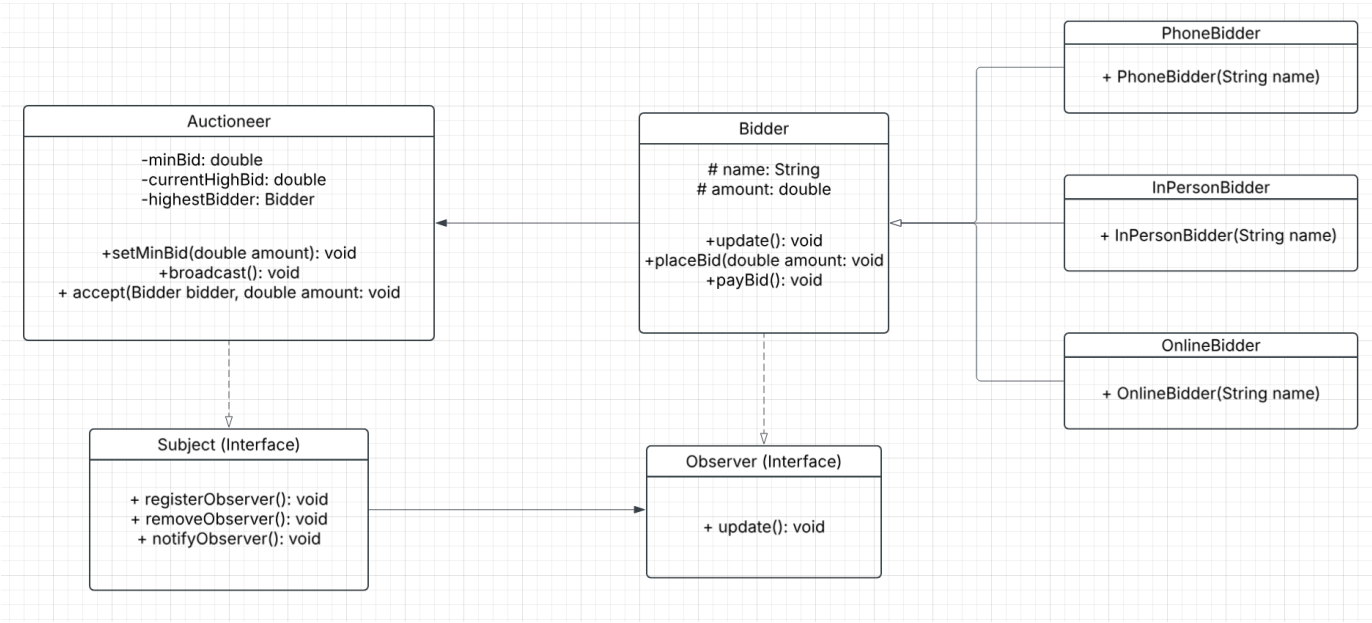
6. Duck to QuackBehavior = HAS-A, Duck has a QuackBehavior, meaning it moves quacking to an external object.

7. Quack to QuackBehavior = IS-A, Quack implements QuackBehavior, meaning it is a type of quack behavior.

8. Squeak to QuackBehavior = IS-A, Squeak implements QuackBehavior, meaning it is a type of quack behavior.

9. MuteQuack to QuackBehavior = IS-A, MuteQuack implements QuackBehavior, meaning it is a type of quack behavior.

Problem 3:


PhoneBidder
+ PhoneBidder(String name)
Auctioneer
-minBid: double
-currentHighBid: double
-highestBidder: Bidder
+setMinBid(double amount): void
+broadcast(): void
+ accept(Bidder bidder, double amount: void
Bidder
name: String
amount: double
+update(): void
+placeBid(double amount: void
+payBid(): void
InPersonBidder
+ InPersonBidder(String name)
OnlineBidder
+ OnlineBidder(String name)
Subject (Interface)
+ registerObserver(): void
+ removeObserver(): void
+ notifyObserver(): void
Observer (Interface)
+ update(): void