CSCI-C322 Assignment 1
Problem 1 Part B
State sales tax computation using the strategy pattern
Donnie Parent

State

- name:String

+ getName():String

+ setName(String name):void

+ showTax(double value):void

<<interface>>
SalesTaxBehavior

+ compute(Double value):Double

Alaska

showTax {
// No sales tax
}

Indiana

showTax {
// 7% sales tax
}

Hawaii

showTax {
// 4.5% sales tax
}

NoTax

compute {
// No sales tax
}

Seven Percent

compute {
// 7% sales tax
}

FourPointFivePercent

compute {
// 4.5% sales tax
}
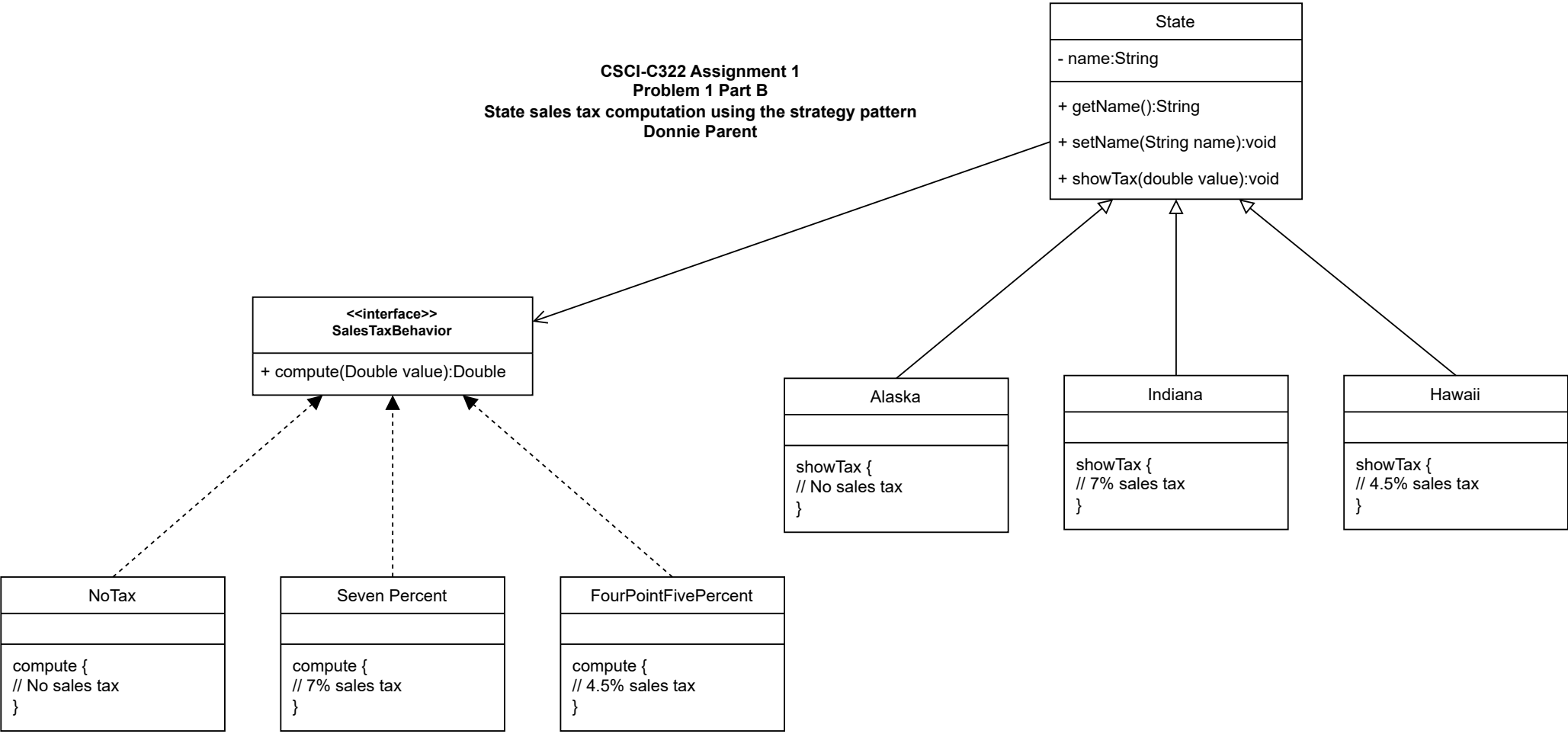
Problem 2

1.   A MallardDuck <mark>IS-A</mark> Duck because MallardDuck inherits behaviors such as swim() or display() from Duck.

2. A RedheadDuck <mark>IS-A</mark> Duck because RedheadDuck inherits behaviors such as swim() or display() from Duck.

3. A RubberDuck <mark>IS-A</mark> Duck because RubberDuck inherits behaviors such as swim() or display() from Duck.

4. A DecoyDuck <mark>IS-A</mark> Duck because DecoyDuck inherits behaviors such as swim() or display() from Duck.

5. A Duck <mark>HAS-A</mark> FlyBehavior because Duck and all of its subclasses contains a FlyBehavior object as a variable (flyBehavior).

6. A Duck <mark>HAS-A</mark> QuackBehavior because Duck and all of its subclasses contain a QuackBehavior object as a variable (quackBehavior).

7. A Quack <mark>IS-A</mark> QuackBehavior because Quack implements the quack() method.

8. A Squeak <mark>IS-A</mark> QuackBehavior because Squeak implements the quack() method.

9. A MuteQuack <mark>IS-A</mark> QuackBehavior because MuteQuack implements the quack() method

Auctioneer

+ registerBidder():void
+ removeBidder(Bidder bidder):Double
+ broadcastHighBid(): void
+ setMinPrice(Double price):void
+ acceptBid(Bidder bidder):void

observer

<<interface>>
Bidder

+ ID:int
- bid:Double

+ getBid():Double
+ setBid(Double amount): void
+ seeUpdatedAmount():Double
+ payBid():void

subject
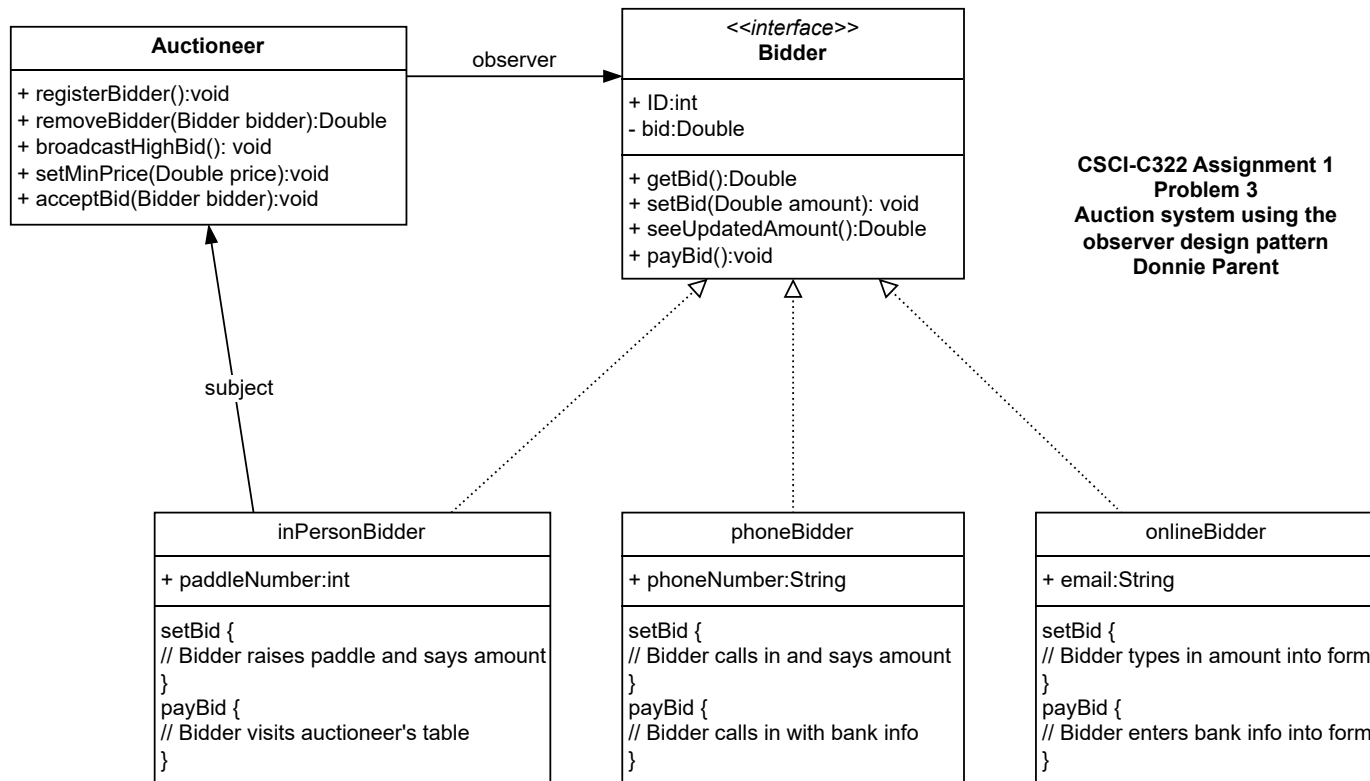
CSCI-C322 Assignment 1
Problem 3
Auction system using the
observer design pattern
Donnie Parent

inPersonBidder

+ paddleNumber:int

setBid {
// Bidder raises paddle and says amount
}
payBid {
// Bidder visits auctioneer's table
}

phoneBidder

+ phoneNumber:String

setBid {
// Bidder calls in and says amount
}
payBid {
// Bidder calls in with bank info
}

onlineBidder

+ email:String

setBid {
// Bidder types in amount into form
}
payBid {
// Bidder enters bank info into form
}