Question 1B:

Part 1B

State
- name : String
+ getName() : String
+ setName(String name) : void
+ showTax(Double value) : void

<< Interface >> Sales Tax Behavior
+ compute(Double value) : Double

Hawaii
showTax {
// 4.5% Sales Tax
}

Alaska
showTax {
// 0% Sales Tax
}

Indiana
showTax {
// 7% Sales Tax
}

No Tax
compute {
// No Sales Tax
}

FourPointFivePercent
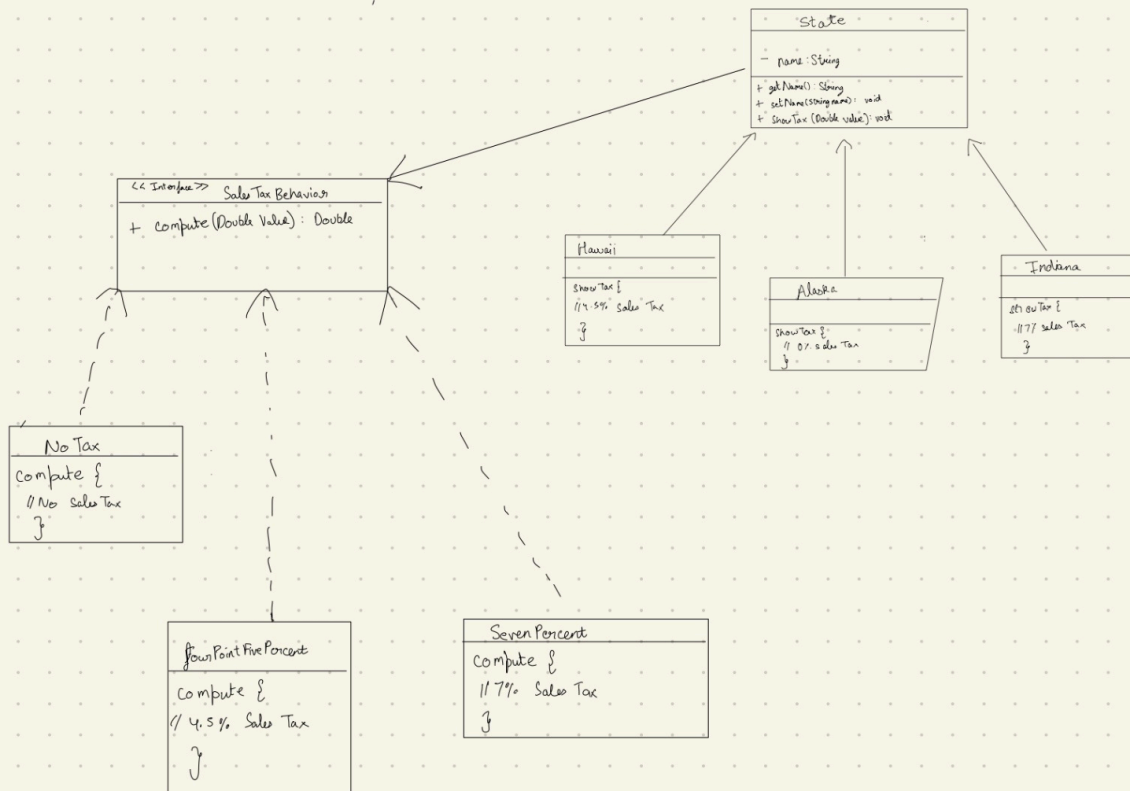compute {
// 4.5% Sales Tax
}

Seven Percent
compute {
// 7% Sales Tax
}

Question 2:
1. MallardDuck IS-A Duck: MallardDuck is a subclass of Duck, inheriting the properties of Duck
2. RedHeadDuck IS-A DUck: RedHeadDuck extends Duck, inheriting the behaviors of Duck
3. RubberDuck IS-A Duck Duck: inherits Duck but can override behaviors (only some)
4. DecoyDuck IS-A Duck: extends Duck, can modify few behaviors of Duck
5. Duck HAS-A FlyBehavior: Duck class has a reference to FlyBehavior
6. Duck HAS-A QuackBehavior: Duck class has a reference to QuackBehavior, dynamically assigning of quacking behaviors
7. Quack IS-A QuackBehavior: Implements QuackBehavior interface
8. Squeak IS-A QuackBehavior: Squeak implements QuackBehavior
9. MuteQuack IS-A QuackBehavior: MuteQuack implements QuackBehavior.

Question 3:

Part 3

<<Interface>>
Bidder

+ Update Bid (Double newBid): void

+ payBid (): void

InPerson Bidder

– name : String
– latestBid : Double

Update Bid (Double newBid) {
// latest bid is NewBid
}
payBid () {
// Print the bid payable
}

Online Bidder

– name : String
– latestBid : Double

Update Bid (Double newBid) {
// latest bid is NewBid
}
payBid () {
// Print the bid payable
}

Phone Bidder

– name : String
– latestBid : Double

Update Bid (Double newBid) {
// latest bid is NewBid
}
payBid () {
// Print the bid payable
}

Auctioneer

bidder : ArrayList< Bidder>
highestBid : Double
minPrice : Double

+ register Bidder (Bidder bidder): void
+ remove Bidder (Bidder bidder): void
+ notify Bidder (): void
+ newBid (Double Bid): void
+ accept Bid (Bidder winBid): void

Live Auctioneer

+ start Auction () {
// announce start
}
+ HighestBid() {
// announce Highest bid
}

+ end Auction {
// announce end
}