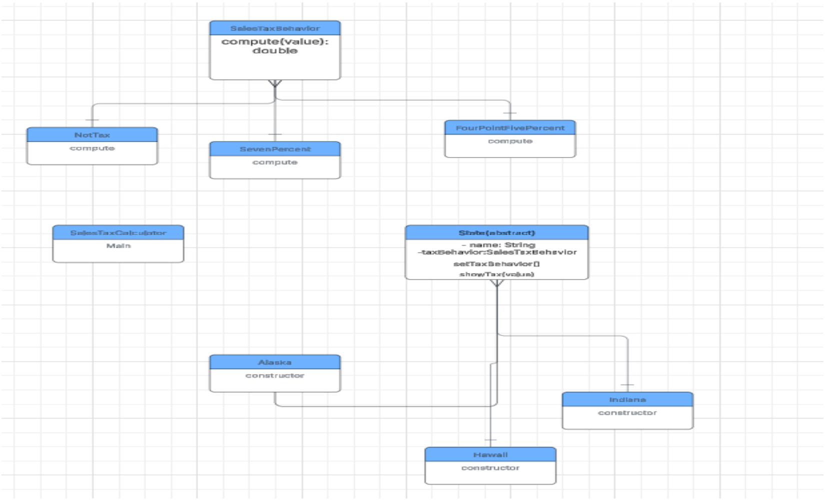
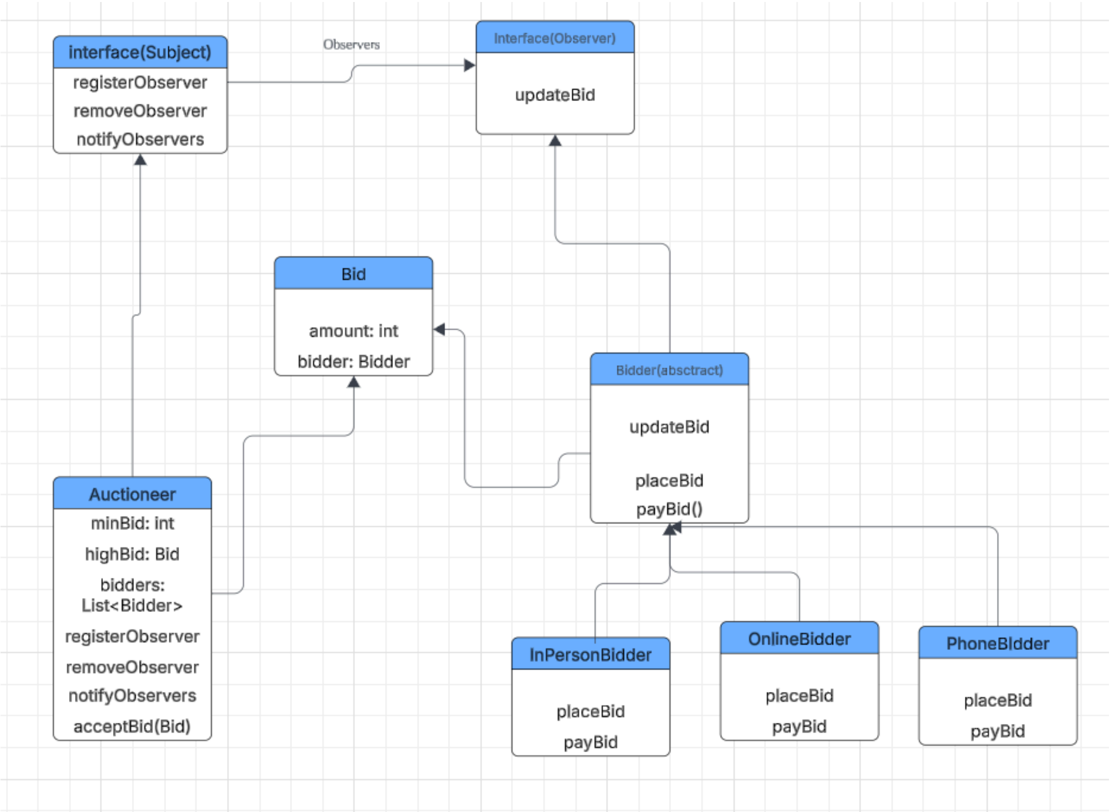
Ruohan You



Problem2:

1: MallardDuck → Duck : This is an IS-A relationship because MallardDuck inherits from Duck.
2: RedheadDuck → Duck : This is an IS-A relationship because RedheadDuck inherits from Duck.
3: RubberDuck → Duck : This is an IS-A relationship because RubberDuck inherits from Duck.
4: DecoyDuck → Duck : This is an IS-A relationship because DecoyDuck inherits from Duck.
5: Duck → FlyBehavior : This is a HAS-A relationship because Duck has a FlyBehavior instance, allowing it to change flying behavior dynamically.
6: Duck → QuackBehavior : This is a HAS-A relationship because Duck has a QuackBehavior instance, allowing it to change quacking behavior dynamically.
7: Quack → QuackBehavior : This is an IS-A relationship because Quack implements QuackBehavior.
8: Squeak → QuackBehavior : This is an IS-A relationship because Squeak implements QuackBehavior.
9: MuteQuack → QuackBehavior : This is an IS-A relationship because MuteQuack implements QuackBehavior.

Problem 3 ：



Comment： I designed this auction system following the Observer Pattern. The auctioneer acts as the subject, notifying all bidders (observers) whenever a new bid is placed. Each bidder can place bids and respond to updates accordingly.