Question 2: Design an app for calling taxis (e.g. Uber);

Objects and Behavior:

Passenger

Data: name, phone,destionation,pick\_up

Behaviors: inputDestination, choose\_pick\_up, schedule,cancel,pay

Driver

Data: name, car\_number,phone

Behaviors: receive, finish,cancel

App

Behaviors: make\_route,searchDriver, start\_order,cal\_price

Order

Data: open\_status

Behaviors: complete

Car

Data: model, car\_number, color, owner

Card:

Data: card\_number, name,address

Behavior

Bank

Data: name

Behavior: authorize

Sequence of invoking Objects with Behavior

Passenger Tom

Driver driver

App uber

Order order

Car car

Card visa

Bank boa

Tom.inputDestionation && Tom.choose\_pick\_up -> uber.make\_route

Tom.schedule-> uber.searchDriver: collection of nearby drivers

Loop

if (order.open\_status is true)

if (driver.receive)

order.open\_status = false

uber.makeOrder

break

end

if (Tom.cancel)

break

end

end

uber.start\_order -> uber.cal\_price

if (driver.finish):

uber.finishOrder -> Tom.pay, card,boa:respond

If (boa.authorize is false)

Tom.failPay -> order.inComplete

else

Tom.finishPay -> order.complete

end

end