1. Program Statement

Program has different and specific functions for a driver depending on the type of vehicle involved; Car, Truck, Boat, or Plane. Driver will not only have the ability to see and store data of one or more of these types of vehicles, but also be able command the turns vehicles can take and trace the route of each vehicle.

2. Requirements

a. **Assumptions**
   i. User enters lowercase letters for menu selection
   ii. User enters integer value for age variable
   iii. User selects appropriate option in Menu Selections, without giving out of bound inputs.
   iv. Speed is in miles per hour
   v. Land Vehicle transmission is automatic
   vi. Truck involved is a big cargo truck
   vii. Altitude 0 means plane is at the airport on ground (not in air)
      1. Measured in feet
   viii. Lights can only be turned on / off while engine is running
   ix. Fuel is measured in gallons
   x. Boat propeller level can only be changed while engine is on
   xi. User can only go on at most of two trips per vehicle

b. **Specifications**
   i. Welcome message to driver
   ii. Vector to store type of vehicle driven each time changing it
   iii. Vector to store turns that user makes
   iv. Vector to store each place of arrival in order
   v. Menu
      1. Menu for Type of Vehicle
         a. Car
            i. 1. Turn on Car
            ii. 2. Accelerate
            iii. 3. Decelerate
            iv. 4. Turn off Car
            v. 5. Turn right
            vi. 6. Turn left
         b. Truck
            i. 1. Turn on Truck
            ii. 2. Accelerate
            iii. 3. Decelerate
            iv. 4. Turn off Truck
            v. 5. Turn right
            vi. 6. Turn left
         c. Boat

i. 1. Turn on Boat
ii. 2. Accelerate
iii. 3. Decelerate
iv. 4. Turn off Boat
v. 5. Launch Boat
vi. 6. Dock Boat
vii. 7. Turn right
viii. 8. Turn left

d. Plane
i. 1. Turn on Plane
ii. 2. Take off
iii. 3. Accelerate
iv. 4. Decelerate
v. 5. Land
vi. 6. Turn off Plane
vii. 7. Turn right
viii. 8. Turn left

e. Taxi (same as Car)

vi. Store data
1. Age of vehicle (default = 0)
2. Price of vehicle (default = 0.0)
3. True or false, is car a race car? (default = false)
4. True or false, is truck a diesel type? (default = false)
5. Number of passengers
   a. <= 4 in Car
   b. <= 1 in Truck
   c. <= 8 in Boat
   d. <= 200 in Plane
6. Fuel Capacity
   a. 45 in Car
   b. 5,000 in Truck
   c. 30 in Boat
   d. 50,000 in Plane
7. Cargo capacity of truck is 100
8. Propeller level of boat is between 0 and 5
9. Starting mileage of land vehicle is 1000 miles
10. Highest altitude of plane: 35,000 ft
    a. 0 = Not in water
    b. 5 = Maximum depth
11. Cost
    a. Plane = $1,000/trip
       i. Gas price = $4/gal
    b. Car = $100/trip if taxi car

i. Gas price = $3/gal
c. Boat = $500/trip for renting
i. Gas price = $3/gal
d. Truck = -$5/trip for delivering cargo
i. Gas price = $3/gal
12. Coefficient of friction of both car and truck is between 0 and 1

vii. Manipulate data
1. Get and set age of each vehicle
2. Get and set price of each vehicle
3. Get and set true or false whether or not car is a race car
4. Get and set true of false whether or not truck is a diesel type

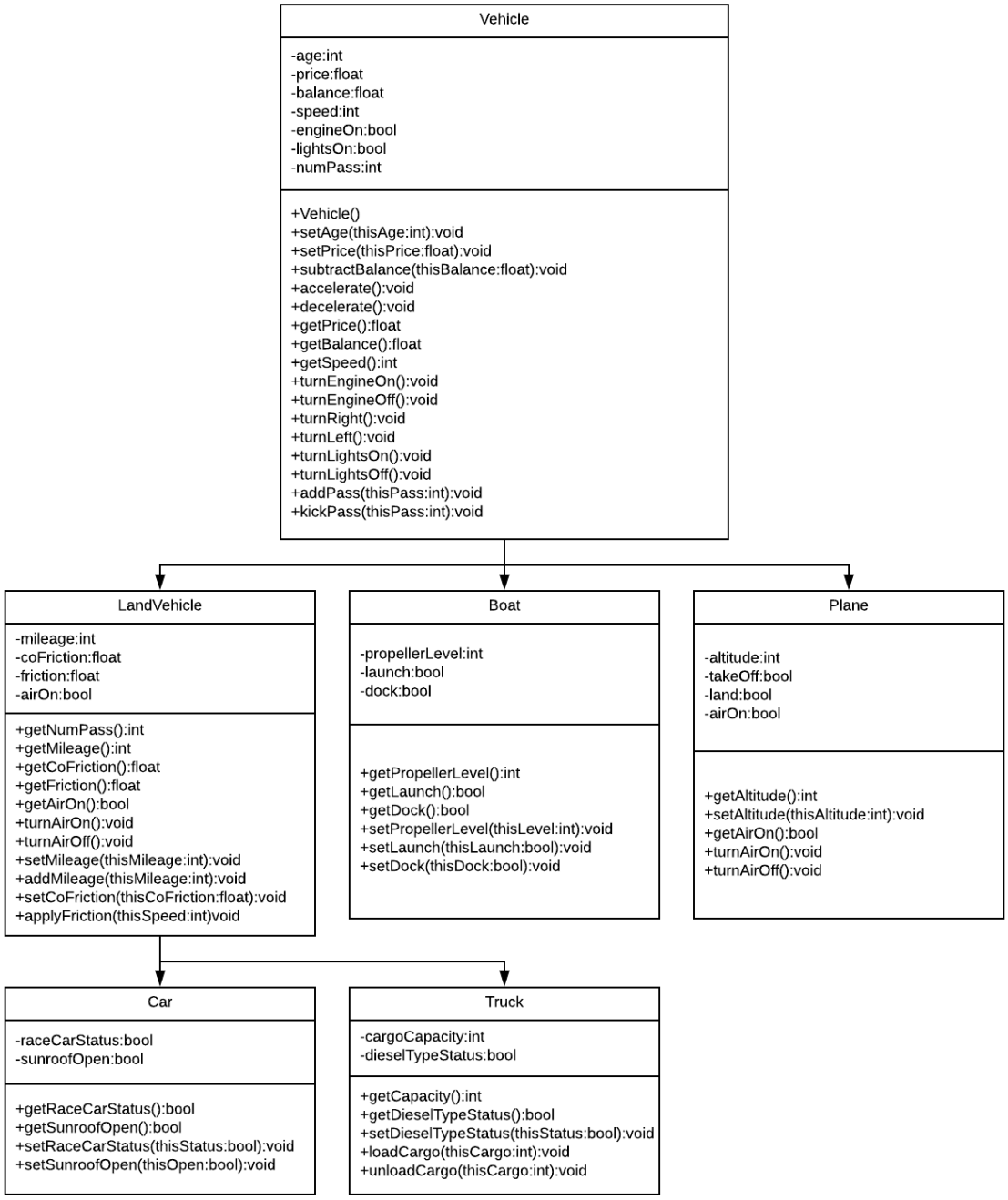
viii. Inheritance
1. Class *Car* and class *Truck* inherit class *LandVehicle* which inherits class *Vehicle*
2. Class *Boat* and class *Plane* inherit class *Vehicle*

ix. Thank you message to the driver

x. Perform checks
1. Only accept values for age more than what was previously stored and should always be positive
2. Only accept values for price less than what was previously stored and should always be positive

xi. Store everything in an output file

3. UML Diagram



4. Decomposition Diagram

Main		
Input	**Process**	**Output**
Age	Check if it is more than previously stored value. Store it in the variable age	"Age of vehicle stored"
Price	Check if it is less than previously stored value. Store it in variable price	"Price of vehicle stored"
Menu selection to get age	Make an appropriate print statement with variable price	Print the statement with age
Menu selection to get price	Make an appropriate print statement with variable print	Print the statement with price
Race car status	Set the passed in status to the variable raceCarStatus	"Race Car Status stored"
Menu selections to get race car status	Make an appropriate print statement with variable raceCarStatus	Print the statement with race car status
Diesel type status	Set the passed in status to the variable dieselTypeStatus	"Diesel Type Status stored"
Menu Selection of owning which type of vehicle	Make an object and call functions of that particular class selected by user	Confirmation saying user selected the particular vehicle
Menu selection of turn lights on / off	Set value of turn lights on to true / false	"Lights are on"
Menu selection of turn engine on	Set value of engine on to true if it is previously false	"Lights are off"
Menu selection of turn engine off	Set value of engine on to false if it is previously true	"Vehicle turned off"
Add this # of passengers	Only add if total # of passengers do not exceed maximum capacity	"Passengers added"
Subtract this # of passengers	Only subtract if final # of passengers do not go below 0	"Passengers kicked out"
Balance	Subtract this amount of balance from stored balance	"Balance deducted"
Menu selection to accelerate	Set speed to a greater number	"Vehicle is at full speed"
Menu selection to decelerate	Set speed to a smaller number	"Vehicle is at a lower speed"
Menu selection to turn right	Assign value 2 in a vector at correct location	"You have turned right"
Menu selection to turn left	Assign value 1 in a vector at correct location	"You have turned left"
Menu selection to turn air on	Set value of air on to true if it is previously false	"Air is on"
Menu selection to turn air off	Set value of air on to false if it is previously true	"Air is off"

Mileage	Add this mileage to car's overall mileage	"Mileage added"
Coefficient of friction	Set this value to the variable of coefficient of friction and call function to apply friction	"Friction applied"
Menu selection to open sunroof	Set value of sun roof open to true if it is previously false	"Sunroof is open"
Menu selection to close sunroof	Set value of sun roof open to false if it is previously true	"Sunroof is closed"
# of cargo to be loaded	Add this # of cargo to current # of cargo only if total # of cargo does not exceed maximum capacity	"Cargo loaded"
# of cargo to be unloaded	Subtract this # of cargo from current # of cargo only if it does not end up with less than 0 cargo	"Cargo unloaded"
Propeller level	If the input is between 1 and 5, then assign the input to the variable	"Propeller level set"
Menu selection to launch boat	Set value of launch to true and value of dock to false if launch is previously false	"Boat launched"
Menu selection to dock boat	Set value of dock to true and value of launch to false if dock is previously false	"Boat docked"

5. Test Strategy

a. **Valid Data**

b. **Invalid Data**

6. Test Plan Version 1

Test Strategy	Test Number	Description	Input	Expected Output	Actual Output	Pass/Fail
Valid	1	Age of vehicle is greater than previously stored value				
Valid	2	Price of vehicle is less than previously stored value				

		unless storing it 1st time				
Valid	3	Price value is always positive				
Valid	4	User enters corresponding number for choosing either true or false for race car status				
Valid	5	User enters corresponding number for choosing either true or false for diesel type status				
Valid	6	User enters corresponding number for choosing from all vehicles on list				
Valid	7	Speed is between 0 mph and 45 mph				
Valid	8	Speed is changed while engine is on, speed is not the same as previous, propeller is under water for boat, and boat is not docked				
Valid	9	User turns on engine while it was previously off				

Valid	10	User turns off engine while speed is 0 mph, engine is previously on, air is off, sun roof is closed, propeller level for boat is 0, boat is docked, and plane altitude is 0 ft				
Valid	11	User turns on the lights while they are previously off, and while engine is on				
Valid	12	User turns off the lights while they are previously on, and while engine is on				
Valid	13	Passengers are added to have total of between 1 and 200, while speed = 0				
Valid	14	Passengers are kicked out to have total of between 0 and 200				
Valid	15	User enters original mileage 1000				
Valid	16	Coefficient of friction is between 0 and 1				

Valid	17	User turns air off while it is previously on and while engine is on				
Valid	18	User turns air on while it is previously off, while engine is on, and while sunroof is closed				
Valid	19	User opens sunroof while it is previously closed, while air is off, and while engine is on				
Valid	20	User closes sunroof while it is previously open and while engine is on				
Valid	21	User sets cargo capacity in beginning to be between 1 and 100				
Valid	22	User adds cargo to have total of 100 or less cargo if number of cargo added is not 0 while truck speed = 0				
Valid	23	User enters propeller level to be				

		between 1 and 5 while boat while boat engine is on				
Valid	24	User selects to launch boat if it is not previously launched				
Valid	25	User selects to dock boat if it is not previously docked				
Valid	26	User sets altitude of plane to be between 15,000 ft to 35,000 ft while speed 100 – 140 mph and while altitude is previously 0				
Valid	27	User sets altitude of plane to be 0 while speed 100 – 140 mph and while altitude is previously 15,000 ft to 35,000				
Invalid	1	Age of vehicle is more than previously stored value				
Invalid	2	Price of vehicle is more than previously				

		stored value unless storing it 1st time				
Invalid	3	Price value is negative				
Invalid	4	User enters number not corresponding to choosing either true or false for race car status				
Invalid	5	User enters number not corresponding to choosing either true or false for race diesel type status				
Invalid	6	User enters number not corresponding to choosing either Car or Truck				
Invalid	7	Speed of car / truck is less than 0				
Invalid	8	Speed of car / truck is more than 45				
Invalid	9	User selects to change speed while engine is off				
Invalid	10	User selects to change speed to the previously stored speed value				
Invalid	11	User selects to change speed while				

		boat propeller level is 0				
Invalid	12	User selects to change speed while boat is not docked				
Invalid	13	User selects to turn engine on while it's on				
Invalid	14	User selects to turn engine off while it's off				
Invalid	15	User selects to turn engine off while speed is more than 0				
Invalid	16	User selects to turn engine off while air is on				
Invalid	17	User selects to turn engine off while sun roof is open				
Invalid	18	User selects to turn engine off while boat propeller level is more than 0				
Invalid	19	User selects to turn engine off while boat is not docked				
Invalid	20	User selects to turn engine off while plane altitude is more than 0				
Invalid	21	User selects to turn lights				

		on while they are on				
Invalid	22	User selects to turn lights on while engine is off				
Invalid	23	User selects to turn lights off while are off				
Invalid	24	User selects to add passengers to have total of 0				
Invalid	25	User selects to add passengers to have total of more than 4 passengers in car				
Invalid	26	User selects to add passengers to have total of more than 1 passengers in truck				
Invalid	27	User selects to add passengers to have total of more than 8 passengers in boat				
Invalid	28	User selects to add passengers to have total of more than 200 passengers in plane				
Invalid	29	User selects to kick 0				

		passengers out				
Invalid	30	User selects to kick passengers out to have total of negative passengers				
Invalid	31	User enters a negative number for mileage				
Invalid	32	Coefficient of friction is less than 0				
Invalid	33	Coefficient of friction is more than 1				
Invalid	34	User selects to turn air off while it is off				
Invalid	35	User selects to turn air on while it is on				
Invalid	36	User selects to turn air on while engine is off				
Invalid	37	User selects to turn air on while sunroof is open				
Invalid	38	User selects to open sunroof while it is on				
Invalid	39	User selects to open sunroof while engine is off				
Invalid	40	User selects to open sunroof while air is on				

Invalid	41	User selects to close sunroof while it is closed				
Invalid	42	User sets cargo capacity in the beginning to be negative				
Invalid	43	User sets cargo capacity in the beginning to be greater than 100				
Invalid	44	User adds cargo to have total of more than 100 cargo				
Invalid	45	User adds cargo to have total of the same amount as before				
Invalid	46	User selects to add cargo while speed of vehicle is greater than 0				
Invalid	47	User enters propeller level to be the same level as before				
Invalid	48	User enters propeller level to be negative				
Invalid	49	User enters propeller level to be more than 5				
Invalid	50	User selects to change propeller				

		level while engine is off				
Invalid	51	User selects to launch boat while it is already launched				
Invalid	52	User selects to dock boat while it is already docked				
Invalid	53	User selects to set altitude of plane to be less than 15,000 ft while speed is 100 – 140 mph				
Invalid	54	User selects to set altitude of plane to be more than 35,000 ft while speed is 100 – 140 mph				
Invalid	55	User selects to set altitude of plane to be between 15,000 ft to 35,000 ft while speed is less than 100				
Invalid	56	User selects to set altitude of plane to be between 15,000 ft to 35,000 ft while speed is more than 140				

Invalid	57	User selects to increase speed of plane while it is previously 500 mph (max)				
Invalid	58	User selects to decrease speed of plane while it is previously 120 mph (min)				
Invalid	59	User selects to set altitude of plane to be 0 while speed is more than 140 mph				
Invalid	60	User selects to set altitude of plane to be 0 while plane is already on land				
Invalid	61	User selects to change altitude while engine is off				

7. Initial Algorithm

a. In class Vehicle

i. Make private variables

1. Age (int)
2. Price (float)
3. Balance (float)
4. Speed (int)
5. engineOn (Boolean)
6. lightsOn (Boolean)
7. numPass (int)

ii. Make a vector called turns to store turns that users take

iii. Constructor

1. Set age to 0

2. Set price to 0.0
3. Set balance to $4,000
4. Set speed = 0
5. Set engineOn to false
6. Set lightsOn to false
7. Set numPass to 0

iv. In *setAge()* function
1. Set vehicle's age to be the value passed in
a. Validate the vehicle's age passed in is greater than previously stored value if not setting the age for 1st time
b. If setting the age for 1st time, validate that value passed in is positive

v. In *setPrice()* function
1. Set vehicle's price to be the value passed in
a. Validate the vehicle's price passed in is smaller than previously stored value if not setting the price for 1st time
b. If setting the price for 1st time, validate that value passed in is positive

vi. In *getAge()* function
1. Return vehicle's age

vii. In *getPrice()* function
1. Return vehicle's price

viii. In *subtractBalance()* function
1. Subtract passed in value from current balance

ix. In *accelerate()* function
1. Add 5 to speed

x. In *decelerate()* function
1. Subtract 5 from speed

xi. In *getBalance()* function
1. Return balance

xii. In *getSpeed()* function
1. Return speed

xiii. In *turnEngineOn()* function
1. Set engineOn variable to true

xiv. In *turnEngineOff()* function
1. Set entingOn variable o false

xv. In *turnRight()* function
1. Store number 3 in first empty space in the vector

xvi. In *turnLeft()* function
1. Store number 1 in the first empty space in the vector

xvii. In *turnLightsOn()* function
1. Set lightsOn variable to true

xviii. In *turnLightsOff()* function

1. Set lightsOn variable to false

xix. In *addPass()* function
1. Add passed in value to current number of passengers

xx. In *kickPass()* function
1. Subtract passed in value from current number of passengers

b. In class LandVehicle (inherits class Vehicle)

i. Private variables
1. Mileage (int) to add mileage of each vehicle
2. Coefficient of friction (int) to calculate friction
3. Friction (float) to be calculated by multiplying coefficient of friction by speed
4. Air on (Boolean) to hold status of A/C on or off

ii. Constructor
1. Set mileage to 1000
2. Set coefficient of friction to 0.5
3. Call function to calculate friction
4. Set air on status to false

iii. In *getNumPass()* function
1. Return number of passengers currently in the car / truck

iv. In *getMileage* function
1. Return current mileage of car / truck

v. In *getCoFriction* function
1. Return coefficient of friction

vi. In *getFriction()* function
1. Return calculated friction

vii. In *getAirOn()* function
1. Return air on status

viii. In *turnAirOn()* function
1. Set air on status to true

ix. In *turnAirOff()* function
1. Set air on status to false

x. In *addMileage()* function
1. Add passed in value to current mileage of vehicle

xi. In *setCoFriction()* function
1. Set the passed in value to the variable of coefficient of friction

xii. In *applyFriction()* function
1. Friction = coefficient of friction * speed

c. In class Car (inherits class LandVehicle)

i. Private variable for race car status (bool)

ii. Private variable for sun roof open (bool)

iii. Constructor
1. Set race car status to false
2. Sets sun roof open status to false

iv. In *setRaceCarStatus()* function
    1. Set race car status to be the value passed in
v. In *getRaceCarStatus()* function
    1. Return race car status
vi. In *getSunroofOpen()* function
    1. Return sunroof open status
vii. In *setSunroofOpen()* function
    1. Set sunroof status to be the value passed in

d. In class Truck (inherits class LandVehicle)
    i. Private variable for diesel type status (bool)
    ii. Private variable for cargo capacity (int)
    iii. Public variable for maximum cargo capacity (int)
    iv. Constructor
        1. Set diesel type status to false
        2. Set cargo capacity to 0
        3. Set maximum cargo capacity to 100
    v. In *setDieselTypeStatus()* function
        1. Set diesel type status to be the value passed in
    vi. In *getDieselTypeStatus()* function
        1. Return diesel type status
    vii. In *getCapacity()* function
        1. Return the current number of cargo in the truck
    viii. In *loadCargo()* function
        1. Add the value passed in to the current number of cargo stored
    ix. In *unloadCargo()* function
        1. Subtract the value passed in from the current number of cargo stored

e. In class Boat (inherits class Vehicle)
    i. Private variables
        1. Propeller level (int)
            a. 0 = Outside of water
            b. 1 to 5 = under water
        2. Launch (Boolean)
        3. Dock (Boolean)
    ii. Constructor
        1. Set propeller level to 0
        2. Set launch to false
        3. Set dock to true
    iii. In *getPropellerLevel()* function
        1. Return current propeller level
    iv. In *getLaunch()* function
        1. Return current launch status
    v. In *getDock()* function

1. Return current dock status

vi. In *setPropellerLevel()* function
1. Set the current propeller level to be the passed in value

vii. In *setLaunch()* function
1. Set the current launch status to be the status passed in

viii. In *setDock()* function
1. Set the current dock status to be the status passed in

f. In class Plane (inherits class Vehicle)
i. Private variables
1. Altitude (int)
2. Take off status (bool) to store that plane is in air
3. Land status (bool) to store that plane is in air

ii. Constructor
1. Set altitude to 0
2. Set take off status to false
3. Set land status to true

iii. In *getAltitude()* function
1. Return current altitude of plane

iv. In *setAltitude()* function
1. Set current altitude to the value passed in

v. In *getAirOn()* function
1. Return air on status

vi. In *turnAirOn()* function
1. Set air on status to true

vii. In *turnAirOff()* function
1. Set air on status to false

g. In class Main
i. Make variables to be used later
ii. Make vector for turns a car, truck or boat makes.
iii. Make vector for storing locations user goes
iv. Make vector for storing which vehicle user uses each time
v. Do following, then loop while user chooses to quit program
1. If user is at home, print a menu with options Car, Truck, Quit
a. If user selects Car
i. Using switch case, print and call functions for all the different things user can do and loop until user says they have reached destination
1. Set age if user gives an age more than or equal to previous age
2. Set price if user gives a price less than or equal to previous price
3. Return balance
4. Turn engine on after performing checks

5. Turn engine off after performing checks
6. Turn lights on / off after performing checks
7. Add / kick passengers after performing checks
8. Return mileage
9. Set coefficient of friction after performing checks
10. Return friction
11. Turn air on / off after performing checks
12. Set race car status after performing checks
13. Open / close sunroof after performing checks
14. Turn right – insert 3 in the first empty box in vector
15. Turn left – insert 1 in the first empty box in vector
16. Reached destination – ask if user wants to return to last location. If yes, quit loop and print returning messages

b. If user selects Truck
   i. Using switch case, print and call functions for all the different things user can do and loop until user says they have reached destination
      1. Set age if user gives an age more than or equal to previous age
      2. Set price if user gives a price less than or equal to previous price
      3. Return balance
      4. Turn engine on after performing checks
      5. Turn engine off after performing checks
      6. Turn lights on / off after performing checks
      7. Add / kick passengers after performing checks
      8. Return cargo capacity
      9. Add / Subtract cargo after performing checks
      10. Return mileage
      11. Set coefficient of friction after performing checks
      12. Return friction
      13. Turn air on / off after performing checks
      14. Set diesel type status after performing checks

15. Turn right – insert 3 in the first empty box in vector
16. Turn left – insert 1 in the first empty box in vector
17. Reached destination – ask if user wants to return to last location. If yes, quit loop and print returning messages

c. If user selects quit, then break out the loop

2. If user is at airport, print a menu with options Plane, Car, Truck, Quit
    a. If user selects Plane
        i. Using switch case, print and call functions for all the different things user can do and loop until user says they have reached destination
            1. Set age if user gives an age more than or equal to previous age
            2. Set price if user gives a price less than or equal to previous price
            3. Return balance
            4. Turn engine on after performing checks
            5. Turn engine off after performing checks
            6. Turn lights on / off after performing checks
            7. Add / kick passengers after performing checks
            8. Turn air on / off after performing checks
            9. Reached destination – ask if user wants to return to last location. If yes, quit loop and print returning messages
            10. Return altitude
            11. Set altitude after performing checks
            12. Return air on status
    b. If user selects Car, using switch case, print and call functions for all the different things user can do and loop until user says they have reached destination
    c. If user selects Truck, using switch case, print and call functions for all the different things user can do and loop until user says they have reached destination
    d. If user selects Quit, break out of the loop and ask which location is the user at and store it in the vector
3. If user is at marina, print a menu with options Boat, Car, Truck, Quit
    a. If user selects Boat

i. Using switch case, print and call functions for all the different things user can do and loop until user says they have reached destination
    1. Set age if user gives an age more than or equal to previous age
    2. Set price if user gives a price less than or equal to previous price
    3. Return balance
    4. Turn engine on after performing checks
    5. Turn engine off after performing checks
    6. Turn lights on / off after performing checks
    7. Add / kick passengers after performing checks
    8. Set / change launch / dock status after performing checks
    9. Set propeller level after performing checks
    10. Reached destination – ask if user wants to return to last location. If yes, quit loop and print returning messages

b. If user selects Car, using switch case, print and call functions for all the different things user can do and loop until user says they have reached destination
c. If user selects Truck, using switch case, print and call functions for all the different things user can do and loop until user says they have reached destination
d. If user selects Quit, break out of the loop and ask which location is the user at and store it in the vector

8. Test Plan Version 2

Test Strategy	Test Number	Description	Input	Expected Output	Actual Output	Pass/Fail
Valid	1	Age of vehicle is greater than previously stored value	Previous: "5" New: "10"	Age stored successfully		
Valid	2	Price of vehicle is less than previously stored value	Previous: "15000" New: "10000"	Price stored successfully		

		unless storing it 1st time				
Valid	3	Price value is always positive	"15000"	"Price stored successfully"		
Valid	4	User enters corresponding number for choosing either true or false for race car status	"1" for true	"Race car status updated"		
Valid	5	User enters corresponding number for choosing either true or false for diesel type status	"1" for true	"Diesel type status updated"		
Valid	6	User enters corresponding number for choosing from all vehicles on list	"1" for Car when 1: Car 2: Truck 3: Quit	"You are driving car now"		
Valid	7	Speed is between 0 mph and 45 mph	"40"	"Speed is now 40"		
Valid	8.1	Speed of car / truck is changed while engine is on, speed is not the same as previous	Engine: On Speed previous: 40 Speed new: "45"	"Speed is now 45"		
Valid	8.2	Speed of boat is changed while engine is on, speed is not the same as previous, propeller is under water	Engine: On Speed previous: 40 Speed new: "45" Propeller level: 2	"Speed is now 45"		

		for boat, and boat is not docked	Boat: launched			
Valid	9	User turns on engine while it was previously off	Previous: engine off New: engine on	"Engine is now on"		
Valid	10.1	User turns off engine while speed is 0 mph, engine is previously on, air is off, sun roof is closed	Speed: 0 Previous: engine on New: engine off Air: off Sunroof: closed	"Engine is now off"		
Valid	10.2	User turns off engine while speed is 0 mph, engine is previously on, propeller level for boat is 0, boat is docked	Speed: 0 Previous: engine on New: engine off Propeller level: 0 Boat: docked	"Engine is now off"		
Valid	10.3	User turns off engine while speed is 0 mph, engine is previously on, air is off, and plane altitude is 0 ft	Speed: 0 Previous: engine on New: engine off Air: off Altitude: 0	"Engine is now off"		
Valid	11	User turns on the lights while they are previously off, and while engine is on	Previous: Lights off New: lights on Engine: on	"Lights are now on"		
Valid	12	User turns off the lights while they are previously on, and while engine is on	Previous: Lights on New: lights off Engine: on	"Lights are now off"		

Valid	13.1	Passengers are added to have total of between 1 and 200 for plane, while speed = 0	Previous: 50 Add: 100 Total: 150 Speed: 0	"Passengers added successfully"		
Valid	13.2	Passengers are added to have total of between 1 and 4 for Car, while speed = 0	Previous: 2 Add: 2 Total: 4 Speed: 0	"Passengers added successfully"		
Valid	13.3	Passengers are added to have total of 1 for Truck, while speed = 0	Previous: 0 Add: 1 Total: 1 Speed: 0	"Passengers added successfully"		
Valid	13.4	Passengers are added to have total of 8 for Boat, while speed = 0	Previous: 4 Add: 4 Total: 8 Speed: 0	"Passengers added successfully"		
Valid	14	Passengers are kicked out to have total of between 0 and 200 for plane, while speed = 0	Previous: 150 Kick: 50 Total:100 Speed = 0	"Passengers kicked out successfully"		
Valid	15	User enters original mileage 1000	"1000"	"Mileage see successfully"		
Valid	16	Coefficient of friction is between 0 and 1	"0.5"	"Coefficient is updated successfully"		
Valid	17	User turns air off while it is previously on and while engine is on	Previous: air on New: air off Engine: on	"Air is not off"		

Valid	18	User turns air on while it is previously off, while engine is on, and while sunroof is closed	Previous: air off New: air on Engine: on Sunroof: closed	"Air is now on"		
Valid	19	User opens sunroof while it is previously closed, while air is off, and while engine is on	Previous: sunroof closed New: sunroof open Air: off Engine on	"Sunroof is now open"		
Valid	20	User closes sunroof while it is previously open and while engine is on	Previous: sunroof open New: sunroof closed Engine: on	"Sunroof is now closed"		
Valid	21	User sets cargo capacity in beginning to be between 1 and 100	"50"	"Current cargo capacity set"		
Valid	22	User adds cargo to have total of 100 or less cargo if number of cargo added is not 0 while truck speed = 0	Previous: 50 Add: 25 Total: 75 Speed: 0	"Cargo added successfully"		
Valid	23	User enters propeller level to be between 1 and 5 for boat while boat engine is on	Level: 2 Engine: on	"Propeller level updated successfully"		

Valid	24	User selects to launch boat if it is not previously launched	Select "launch" Previous: docked	"Boat is now launched"		
Valid	25	User selects to dock boat if it is not previously docked	Select "dock" Previous: launched	"Boat is now docked"		
Valid	26	User sets altitude of plane to be between 15,000 ft to 35,000 ft while speed 100 – 140 mph and while altitude is previously 0	Speed: 130 Set new altitude: "30000" Previous altitude: 0	"Altitude of plane updated successfully"		
Valid	27	User sets altitude of plane to be 0 while speed 100 – 140 mph and while altitude is previously 15,000 ft to 35,000	Speed: 110 Set new altitude: "0" Previous altitude: 18000	"Altitude of plane updated successfully"		
Invalid	1	Age of vehicle is less than previously stored value	Previous: 15 New: 10	"Invalid input: Age cannot be less than last one"		
Invalid	2	Price of vehicle is more than previously stored value unless storing it 1st time	Previous: 15000 New: 20000	"Invalid input: Price cannot be more than last one"		
Invalid	3	Price value is negative	"-25000"	"Invalid input: Price		

				cannot be negative”		
Invalid	4	User enters number not corresponding to choosing either true or false for race car status	“3”	“Invalid input: Try again”		
Invalid	5	User enters number not corresponding to choosing either true or false for race diesel type status	“100”	“Invalid input: Try again”		
Invalid	6	User enters number not corresponding to choosing either Car or Truck	“-4”	“Invalid input: Try again”		
Invalid	7	Speed of car / truck is less than 0	“-15”	“Invalid input: Speed cannot be less than 0”		
Invalid	8	Speed of car / truck is more than 45	“55”	“Invalid input: Speed cannot be more than 45”		
Invalid	9	User selects to change speed while engine is off	Engine: off Previous speed: 0 New: 45	“Invalid input: Cannot change speed while engine is off”		
Invalid	10	User selects to change speed to the previously stored speed value	Previous: 40 New: 40	“Invalid input: Cannot change speed to the same value as before”		

Invalid	11	User selects to change speed while boat propeller level is 0	Previous: 0 New: 30 Propeller level: 0	"Invalid input: Cannot change speed while boat propeller level is not in water"		
Invalid	12	User selects to change speed while boat is not launched	Previous: 0 New: 20 Boat: docked	"Invalid input: Cannot change speed while boat is docked"		
Invalid	13	User selects to turn engine on while it's on	Previous: Engine on New: Engine on	"Invalid input: Cannot turn engine on while it is already on"		
Invalid	14	User selects to turn engine off while it's off	Previous: Engine off New: Engine off	"Invalid input: Cannot turn engine off while is already off"		
Invalid	15	User selects to turn engine off while speed is more than 0	Previous: engine on New: engine off Speed: 14	"Invalid input: Cannot turn engine off while vehicle is moving"		
Invalid	16	User selects to turn engine off while air is on	Air: on Previous: engine on New: engine off	"Invalid input: Cannot turn engine off while A/C is on"		
Invalid	17	User selects to turn engine off while sun roof is open	Sunroof: open Previous: engine on	"Invalid input: Cannot turn engine of while		

			New: engine off	sunroof is open"		
Invalid	18	User selects to turn engine off while boat propeller level is more than 0	Propeller level: 1 Previous: engine on New: engine off	"Invalid input: Cannot turn engine off while boat propeller is inside water"		
Invalid	19	User selects to turn engine off while boat is not docked	Boat: launched Previous: engine on New: engine off	"Invalid input: Cannot turn engine off while boat is launched"		
Invalid	20	User selects to turn engine off while plane altitude is more than 0	Altitude: 20,000 Previous: engine on New: engine off	"Invalid input: Cannot turn engine off while plane is in air"		
Invalid	21	User selects to turn lights on while they are on	Previous: lights on New: lights on	"Invalid input: Cannot turn lights on if they already on"		
Invalid	22	User selects to turn lights on while engine is off	Engine: off Previous: lights off New: lights on	"Invalid input: Cannot turn lights on if engine is off"		
Invalid	23	User selects to turn lights off while are off	Previous: lights off New: lights off	"Invalid input: Cannot turn lights off while they are already off"		
Invalid	24	User selects to add passengers to have total of 0	Previous passengers: 0 Add: "0" Total: 0	"Invalid input: Cannot add 0 passengers"		

Invalid	25	User selects to add passengers to have total of more than 4 passengers in car	Previous passengers: 2 Add: 3 Total: 5	"Invalid input: Cannot add passengers to have total of more than 4"		
Invalid	26	User selects to add passengers to have total of more than 1 passengers in truck	Previous passengers: 0 Add: 2 Total: 2	"Invalid input: Cannot add passengers to have total of more than 1"		
Invalid	27	User selects to add passengers to have total of more than 8 passengers in boat	Previous passengers: 3 Add: 10 Total: 13	"Invalid input: Cannot add passengers to have total of more than 8"		
Invalid	28	User selects to add passengers to have total of more than 200 passengers in plane	Previous passengers: 100 Add: 150 Total: 250	"Invalid input: Cannot add passengers to have total of more than 200"		
Invalid	29	User selects to kick 0 passengers out	Kick "0" passengers out	"Invalid input: Cannot kick 0 passengers out"		
Invalid	30	User selects to kick passengers out to have total of negative passengers	Previous passengers: 2 Kick: 3 Total: -1	"Invalid input: Cannot kick out more passengers than there are"		
Invalid	31	User enters a negative number for mileage	"-1000"	"Invalid input: Cannot have negative mileage"		

Invalid	32	Coefficient of friction is less than 0	"-1"	"Invalid input: Cannot have negative coefficient of friction"		
Invalid	33	Coefficient of friction is more than 1	"2"	"Invalid input: Cannot have coefficient of friction more than 2"		
Invalid	34	User selects to turn air off while it is off	Previous air: off New air: off	"Invalid input: Cannot turn engine off if it is already off"		
Invalid	35	User selects to turn air on while it is on	Previous air: on New air: on	"Invalid input: Cannot turn A/C on while it is already on"		
Invalid	36	User selects to turn air on while engine is off	Engine: off Previous air: off New air: on	"Invalid input: Cannot turn air on if engine is off"		
Invalid	37	User selects to turn air on while sunroof is open	Sunroof: on Previous air: off New air: on	"Invalid input: Cannot turn air on if sunroof is open"		
Invalid	38	User selects to open sunroof while it is on	Previous sunroof: open New sunroof: open	"Invalid input: Cannot open sunroof if it is already open"		
Invalid	39	User selects to open sunroof while engine is off	Engine: off Previous sunroof: closed	"Invalid input: Cannot open sunroof if		

			New sunroof: open	engine is off"		
Invalid	40	User selects to open sunroof while air is on	Air: on Previous sunroof: closed New sunroof: open	"Invalid input: Cannot open sunroof if A/C is on"		
Invalid	41	User selects to close sunroof while it is closed	Previous sunroof: closed New sunroof: closed	"Invalid input: Cannot close sunroof if it is already closed"		
Invalid	42	User sets cargo capacity in the beginning to be negative	"-50"	"Invalid input: Cannot have a negative cargo"		
Invalid	43	User sets cargo capacity in the beginning to be greater than 100	"150"	"Invalid input: Cannot have more than 100 cargo"		
Invalid	44	User adds cargo to have total of more than 100 cargo	Previous cargo: 10 Add: "100" Total: 110	"Invalid input: Cannot add cargo to have total of more than 100 cargo"		
Invalid	45	User adds cargo to have total of the same amount as before	Previous cargo: 75 Add: 0 Total: 75	"Invalid input: Cannot add 0 cargo"		
Invalid	46	User selects to add cargo while speed of vehicle is greater than 0	Speed: 40 Select add cargo	"Invalid input: Cannot add cargo if vehicle is moving"		

Invalid	47	User enters propeller level to be the same level as before	Previous level: 1 New: "1"	"Invalid input: Cannot set propeller level to be same as before"		
Invalid	48	User enters propeller level to be negative	"-4"	"Invalid input: Propeller level cannot be negative"		
Invalid	49	User enters propeller level to be more than 5	"10"	"Invalid input: Propeller level cannot be more than 5"		
Invalid	50	User selects to change propeller level while engine is off	Engine: off Previous level: 4 New: 5	"Invalid input: Cannot change propeller level if engine is off"		
Invalid	51	User selects to launch boat while it is already launched	Previous boat: launched New: Launch	"Invalid input: Cannot launch boat if it is already launched"		
Invalid	52	User selects to dock boat while it is already docked	Pervious boat: docked New: docked	"Invalid input: Cannot dock boat if it is already docked"		
Invalid	53	User selects to set altitude of plane to be less than 15,000 ft while speed is 141-500 mph	Speed: 135 Previous altitude: 15000 Set altitude: "1"	"Invalid input: Cannot change altitude to be less than 15,000 if		

		while landing / taking off		speed is 141 – 500 mph ”		
Invalid	54	User selects to set altitude of plane to be more than 35,000 ft while speed is 100-140 mph while landing / taking off	Speed: 135 Previous altitude: 0 Set altitude: “99000”	“Invalid input: Plane cannot go above altitude of 35,000”		
Invalid	55	User selects to set altitude of plane to be between 15,000 ft to 35,000 ft while speed is less than 100 while landing / taking off	Speed: 99 Previous altitude: 0 Set altitude: “30150”	“Invalid input: Cannot change altitude if speed is less than 100 mph”		
Invalid	56	User selects to set altitude of plane to be between 15,000 ft to 35,000 ft while speed is more than 140 while landing / taking off	Speed: 141 Previous altitude: 0 Set altitude: “30150”	“Invalid input: Cannot change altitude if speed is more than 140 mph”		
Invalid	57	User selects to increase speed of plane while it is previously 500 mph (max)	Previous speed: 500 Add: 1 Total: 501	“Invalid input: Plane cannot go more than 500 mph”		
Invalid	58	User selects to decrease speed of plane while it is previously 120 mph (min)	Previous speed: 120 Subtract: 1 Total: 119	“Invalid input: Plane cannot go less than 120 mph to maintain altitude”		

Invalid	59	User selects to set altitude of plane to be 0 while speed is more than 140 mph	Speed: 145 Previous altitude: 15001 Set altitude: 0	"Invalid input: Cannot land plane if speed is more than 140 mph"		
Invalid	60	User selects to set altitude of plane to be 0 while plane is already on land	Previous altitude: 0 Speed 120 Set altitude: 0	"Invalid input: Cannot land plane if it is already on ground"		
Invalid	61	User selects to change altitude while engine is off	Engine: off Set altitude: "15001"	"Invalid input: Cannot change altitude if engine is off"		

9. Code

10. Updated Algorithm

a.

11. Test Plan Version 3

Test Strategy	Test Number	Description	Input	Expected Output	Actual Output	Pass/Fail

12. Screenshots

13. Status