Bozhidar Mindov

COS221a

Vladimir Georgiev

Course Project Specification

**Car Dealership**

**Project idea:**

My idea of a course project is to make a car dealership program. The aim of the program will be to create different types of vehicle objects, using the implemented classes, place them in a collection and later sort this collection. The program will have 8 classes in total.

**Class Hierarchy:**

Diagram

Description automatically generated

**Classes:**

There will be a base class **Vehicle** with 2 abstract methods. Each class inheriting **Vehicle** will have to implement its own version of the methods.

Class **Car,** that inherits **Vehicle**, will create generalized car objects. Classes **SUV** and **Van**, inherit **Car**, and will represent a more specific types of cars.

Class **Truck,** whichinherits **Vehicle,** will create generalized truck objects. A class called **Pickup-truck,** inherits **Truck**, will be implemented, and will represent a more specific type of truck.

Class **Motorcycle,** which inherits **Vehicle,** will create generalized motorcycle objects. A class called **Electrical-scooter**, inherits **Motorcycle,** will be implemented, and will represent a more specific type of motorcycle.

Each of the classes will utilize some sort of a fuel type enum. Example in class Car: **enum FuelTypes {petrol, diesel, hybrid-diesel, hybrid-petrol, electric}**

**Polymorphic methods:**

There will be a couple of polymorphic methods, that each of the classes will have to implement.

The polymorphic methods will be:

**int CalculateWarranty()** = 0; It will return the years of warranty that a certain vehicle has, usually based on its fuel type and the type of vehicle.

**double CalculateTotalPrice(**) = 0; It will calculate and return the total price of a vehicle. It will be based on the type of the vehicle, its power, size and other properties, such as fuel type.

**double CalculateInsurance()** = 0; It will calculate and return the price of the monthly insurance that will have to be paid for the vehicle. It will again be based on the type of the vehicle, its power, size, and other properties, such as fuel type.

**Algorithms that will be used:**

In the main function, data members needed for the creation of class objects will be read from a file (.txt or maybe .xlsx). Vehicle objects will then be created and placed in a **vector.** From then on, the items will be sorted by **totalPrice**, **insurance** or even **warranty period** using **Merge Sort/Quick sort.**

A basic search could also be implemented using **BM/KPM** pattern matching algorithms. For example, it will allow the user to type in the console the model of a car, and if the car appears in the vector of vehicles, it will be printed to the console, along some of its data members.