Complete set of code is written in Java 7.

It takes the argument as

N : number of cars.

O: origin of the car.

Use of complete Object oriented design is used.

Data set in object: Car.

Filtered manually without using any library-defined function like collections.comperator() and Collections.sort().

Used self written function over it,

findCarsOfOrigin()

findCarsWithMoreAvverageHP()

File reader needs location for input of file kindly change it accordingly.

Solution to the problem:

```
package com.company;

import java.io.*;
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class FindCars {

    public static void main(String[] args) throws IOException {

        File initialFile = new File("src/cars_input.txt"); //file location
        BufferedReader reader = new BufferedReader(new FileReader(initialFile));

        String st;
        ArrayList<Car> carsArrayList = new ArrayList<>();
        while ((st = reader.readLine()) != null){
            String[] s = st.split(",");
            double horsepower = Double.parseDouble(s[2]);
            Car car = new Car(s[0],s[1],horsepower);
            carsArrayList.add(car);

        }
```

```
        int no = Integer.parseInt(args[0]);
        String requiredOrigin = args[1];
        ArrayList<Car> requiredOriginCars =
findCarsOfOrigin(requiredOrigin,carsArrayList);
        List<Car> requiredCars =
findCarsWithMoreAverageHP(requiredOriginCars,no);
        for(Car c : requiredCars){
           System.out.println(c.toString());
        }


    }


    private static List<Car> findCarsWithMoreAverageHP(ArrayList<Car>
requiredOriginCars, int no) {
        double sum = 0;
        for(Car c: requiredOriginCars){
              sum = sum+ c.getHorsepower();
        }
        double averageHorsePower = sum/requiredOriginCars.size();
        List<Car> requiredCars = new ArrayList<>();
        for(Car c:requiredOriginCars){
           if(c.getHorsepower() > averageHorsePower){
              requiredCars.add(c);
           }
        }
         requiredCars =  requiredCars.subList(0,no);
        return  requiredCars;
```

```
    }

    private static ArrayList<Car> findCarsOfOrigin(String requiredOrigin,
ArrayList<Car> allCars) {
        ArrayList<Car> filteredCars = new ArrayList<>();
        for(Car c: allCars){
            if(c.getOrigin().equals(requiredOrigin)){
                filteredCars.add(c);
            }

        }
        return  filteredCars;
    }


    public static class Car {
        String carName;
        String origin;
        double horsepower;

        Car(String carName, String origin, double horsepower){
            this.carName = carName;
            this.origin = origin;
            this.horsepower = horsepower;
        }

        public String getCarName() {
```

```
    return carName;
}

public void setCarName(String carName) {
    this.carName = carName;
}

public String getOrigin() {
    return origin;
}

public void setOrigin(String origin) {
    this.origin = origin;
}

public double getHorsepower() {
    return horsepower;
}

public void setHorsepower(double horsepower) {
    this.horsepower = horsepower;
}

@Override
public String toString() {
    return "Cars{" +
```

```
                "carName='" + carName + '\'' +
                ", origin='" + origin + '\'' +
                ", horsepower=" + horsepower +
                '}';
        }
    }
}
```