CSC120 Week 4 Lab: Comparison

Instructor: Mitsu Ogihara

The goal of this lab is to write a program that compares four cars. The program receives from the user the model, the total mileage, and the total gas consumed for each of the four cars and computes the miles per gallon for preparation. The information to use consists of `String carN`, `double milesN`, `gasN`, and `mpgN`, where `N` ranges from 1 to 4. The `carN`, `milesN`, and `gasN` values are entered by the user and `mpgN` is computed as `milesN / gasN`.

Comparing Four Numbers

After receiving the input from the user, the program reorders the cars in the decreasing order of mileage, in the decreasing order of gas, and in the decreasing order of mpg. To reorder cars, we use a method `takeoutMax` that quantities for comparison along with their labels. Using the concept of method overloading, we create three versions of `takeoutMax`, one that handles four cars, one that handles three cars, and one that handles two cars. The one that handles two cars receives four parameters `double h`, `double i`, `String xh`, and `String xi`. The program first computes the maximum `m` among `h` and `i`. Then if `m == h`, it prints `xh` and `h` in one line and then prints `xi` and `i` in one line. Otherwise, the order between the two lines is reversed. To report the label and the value, the method uses `printf` with the format `ORDER` defined as

```
public static final String ORDER = "%s %.2f\n";
```

The version for three cars takes three `double` numbers and three `String` values, `e`, `f`, `g`, `xe`, `xf`, and `xg`. In a manner similar to the two-car version, the program computes the maximum of the three `double` values first. The program then checks if the maximum is equal to `e`, `f`, or `g`. If the value matches the maximum, the program reports the label and the value, and then gives the remaining two numbers and two labels to the two-car version.

Here is how the program works. The first four lines collect information from the user. The program then produces the ranking based on MPG, Mileage, and Gas.

```
Car 1 make, milage, gas: Toyota 23300 1005
Car 2 Enter make, milage, gas: Honda 25400 1009
Car 3 make, milage, gas: BMW 35000 3000
Car 4 make, milage, gas: Ford 45000 3530
MPG
Honda 25.17
Toyota 23.18
Ford 12.75
BMW 11.67
Mileage
```

```
Ford 45000.00
BMW 35000.00
Honda 25400.00
Toyota 23300.00
Gas
Ford 3530.00
BMW 3000.00
Honda 1009.00
Toyota 1005.00
```

A Strategy

In writing the three versions of `takeoutMax`, you may start with the two-car version, then proceed to a higher number.

1. Write the main method that receives information for the four cars.
2. Write the two-car version. Make sure that the code works by calling the method with just two cars, ignoring the other two.
3. Write the three-car version. Make sure that the code works by calling the method with just three cars, ignoring the last one.
4. Write the four-version.

In this manner, even if you do not complete the work, you will be able to receive a partial credit.