TSU Software, Materia: Programación, Clave 50086, Semestre 17-P
Practica 03_A. Operadores y estructuras de control.

1. Objetivos

Los objetivos buscados en esta practica son los siguientes:

- Reconocer y utilizar los distintos tipos de operadores del lenguaje Java.
- Reconocer y utilizar las distintas estructuras de control del lenguaje Java.
- Utilizar los operadores y estructuras de control para resolver problemas.

2. Ejercicios.

Exercise 1.

Uber is introducing a new feature for drivers that tells them if they'll need gas within the next hour. It analyzes all of the drive distances (in miles) that the driver has traveled in the past `12` hours to make the recommendation.

Given the current gas level (in gallons), the drive data, and the average mileage of the vehicle, calculate the average amount of gas spent per hour and return `true` if the driver is likely to need a refill in the next hour, `false` otherwise.

Assume that the driver will need more gas if the average gas consumption per hour is **greater** than the amount of gas they have at the given moment. The average mileage is measured in *miles per gallon* and it shows how many miles a vehicle can travel on one gallon of gas.

Example

For `driveDistances = [12, 6, 17, 5, 20]`, `currentGasLevel = 0.25` and `avgMileage = 25`, the output should be

```
gasPrediction(driveDistances, currentGasLevel, avgMileage) = false.
```

Since for `driveDistances = [12, 6, 17, 5, 20]`, `currentGasLevel = 0.25g` and `avgMileage = 25mpg`, the total distance traveled is equal to `60` and it can be shown that on average the driver spends `0.2g` of gas per hour, which is less than `0.25g`.

Input/Output

- **[time limit] 4000ms**
- **[input] array.float driveDistances**

  An array of positive number representing the drive distances (in miles) that the driver has traveled in the past `12` hours.
  Constraints:

  `1 ≤ driveDistances.length ≤ 10`,

  `4.0 ≤ driveDistances[i] ≤ 120.0`.
- **[input] float currentGasLevel**

  The current gas level in gallons.

  Constraints:

  `0.25 ≤ currentGasLevel ≤ 10.0`.
- **[input] float avgMileage**

  The average mileage of the vehicle.

  Constraints:

  `1.0 ≤ avgMileage ≤ 60.0`.
- **[output] boolean**

  `true` if the driver is likely to need a refill in the next hour, `false` otherwise.