

Programming Methodology - Lab

2022-2023

Lab task 2. Divide and Conquer algorithms



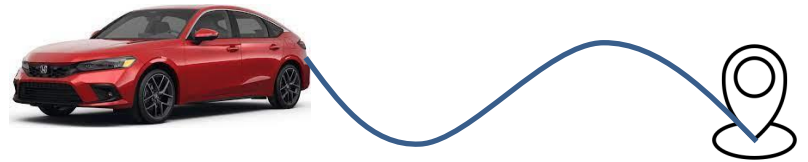
Programming Methodology - Lab

2022-2023

Lab task 2. Divide and Conquer algorithms

SESSION 1

Fuel Consumption. Session 1 (I)

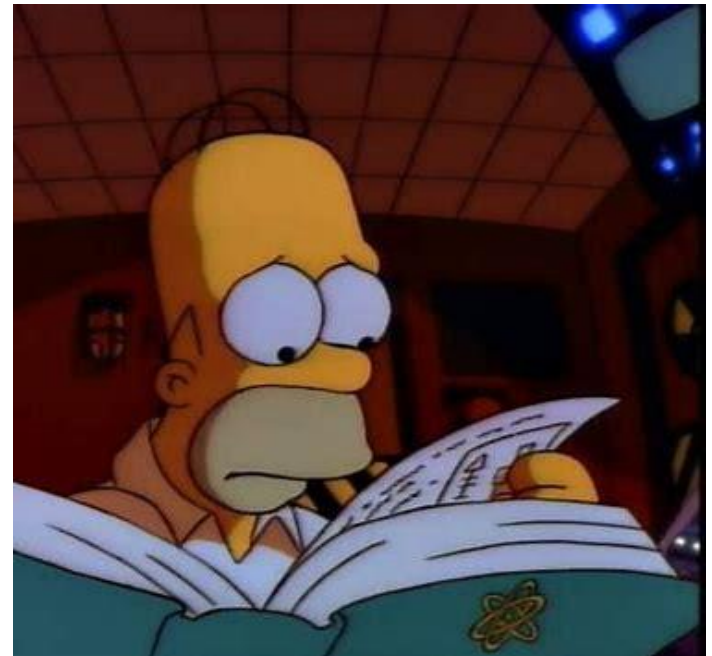


- **General assumptions:**

- A dataset of cars ("*cars_dataset.csv*")

- Each car:

- **Model**
 - Fuel type
 - Transmission type
 - Number of seats
 - **Fuel tank capacity**
 - **(Average) fuel consumption**



- All vehicles will perform the same route

- N POIs (read from keyboard)
 - Distances between POIs (randomly generated)

Fuel Consumption. Session 1 (II)

- **Objectives:**

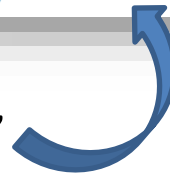
1. Calculate the total consumption of each vehicle
2. Show the cars that can complete the trip, sorted in non-decreasing order of total consumption
3. Indicate the vehicles, if any, that could not complete the trip

- **In this first session (I):**

- A base Java program to do:

- Read the file “*cars_dataset.csv*”
- Read number of POIs from keyboard
- Generate random number about distances between POIs
 - Random class: [Random \(Java Platform SE 8 \) \(oracle.com\)](https://docs.oracle.com/javase/8/docs/api/java/util/Random.html)

```
Hyundai Creta,Diesel,5.0,Automatic,50.0,7.1
Range Rover,Petrol,7.0,Automatic,90.0,11.35
...
```



Fuel Consumption. Session 1 (III)

- In this first session (II):
 - Design the **Divide and Conquer strategy** to solve the problem for:
 - Calculating **total consumption** of each vehicle, considering
 - Distances between POIs
 - Car tank capacity
 - Average fuel consumption

When the base case is achieved?

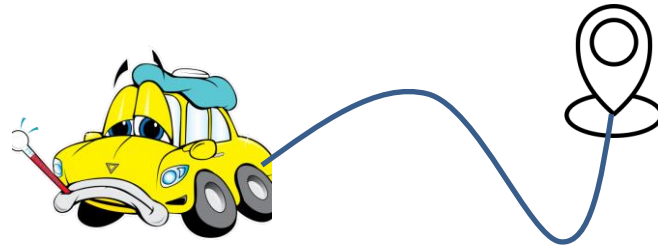
What could be done in the general case?

Do not forget to consider when the trip cannot be completed!



Fuel Consumption. Session 1 (IV)

- In this first session (III):
 - Study the most appropriate method to list the **vehicles capable of completing the trip** (based on their total consumption)



“Result pieces”

(1) Reading dataset and showing formatted data

```
Processing cars_dataset...
##### CARS #####
Maruti Alto K10-Petrol-5.0-Automatic-27.0-4.09
Maruti Brezza-Petrol-5.0-Automatic-48.0-5.21
Mahindra Thar-Diesel-4.0-Automatic-57.0-10.48
Mahindra XUV700-Diesel-7.0-Automatic-60.0-6.16
Mahindra Scorpio-N-Diesel-7.0-Automatic-57.0-9.03
Toyota Fortuner-Diesel-7.0-Automatic-80.0-9.85
Hyundai Creta-Diesel-5.0-Automatic-50.0-7.1
Tata Punch-Petrol-5.0-Automatic-37.0-9.87
```

(2) Read N POIs and show distances

```
Distances (in kms) among numbered POIs
[0 --> 1]: 86
[1 --> 2]: 53
[2 --> 3]: 122
[3 --> 4]: 34
[4 --> 5]: 121
[5 --> 6]: 118
[6 --> 7]: 87
```

Programming Methodology - Lab

2022-2023

Lab task 2. Divide and Conquer algorithms

SESSION 2

Fuel Consumption. Session 2 (I)

- In this second session (I):
 - Implement the **Divide and Conquer algorithm** to solve the problem.
 - Show the results:
 - Sorted information about the total consumption of the vehicles
 - Vehicles that cannot complete the trip
 - Carried out several simulations with different POIs



Fuel Consumption. Session 2 (II)

“Result pieces”

(1) Reading dataset and showing formatted data

...

```
Total consumption for the car BMW X7: 27.324(Tank capacity: 80.0)
Total consumption for the car Hyundai Alcazar: 27.3861(Tank capacity: 50.0)
Total consumption for the car Hyundai Xcent Prime: 27.758699999999997(Tank capacity: 65.0)
Total consumption for the car Porsche Cayenne: 28.0071(Tank capacity: 90.0)
Total consumption for the car Volkswagen Tiguan: 28.255499999999998(Tank capacity: 60.0)
Total consumption for the car Nissan GT-R: 28.3176(Tank capacity: 74.0)
```

...

```
Total consumption for the car Rolls Royce Wraith: 74.52(Tank capacity: 83.0)
The car Mahindra Thar will not complete the trip. Tank capacity: 57.0
The car Tata Punch will not complete the trip. Tank capacity: 37.0
The car Mahindra Bolero will not complete the trip. Tank capacity: 60.0
The car Maruti Swift will not complete the trip. Tank capacity: 37.0
The car Tata Nexon will not complete the trip. Tank capacity: 44.0
```

Fuel Consumption. Session 2 (III)

- In this second session (II):
 - Calculate the theoretical **complexity of the Divide and Conquer** algorithm
 - Justify the answer
 - Calculate the **complexity of all methods/algorithms** implemented
 - Justify the answer

- ✓ Decide the technique to be applied!
- ✓ Give the closed form
- ✓ Indicate the order of growth
- ✓ Justify

