COMP101 LAB6

# Requirements

Write a program which with the minimal amount of questions identifies the class and type of a vehicle and then calculates the cost for the vehicle to go through a tunnel given weather it is a weekday or not

# Analysis and Design

There will be 3 classes within my solution, a Vehicle class, responsible for identifying the vehicle type and class, a TunnelTollCost Class, responsible for calculating the cost of the trip and a TunnelTollCostUser Class responsible for producing the understandable output to the user

## PseudoCode

### Vehicle Class

Vehicle.getIntInput(int MinValue)

INT Value = -1

WHILE Value<minValue

Value = INPUT INT

IF Value < minValue

OUTPUT The number must be greater than or equal to to minValue

OUTPUT enter a different number:

ELSE

BREAK

RETURN Value

Vehicle.getDoubleInput(int MinValue)

DOUBLE Value = -1

WHILE Value<minValue

Value = INPUT DOUBLE

IF Value < minValue

OUTPUT The number must be greater than or equal to to minValue

OUTPUT enter a different number:

ELSE

BREAK

RETURN Value

Vehicle.inputWheels()

OUTPUT Please enter the number of wheels:

wheels = getIntInput(2)

Vehicle.inputLength()

OUTPUT lease enter the Vehicle Length:

length = getDoubleInput(0)

these are repeated for inputting axles and weight

Vehicle.Identify()

inputWheels()

IF wheels<4

RETURN 1

inputLength()

IF wheels==4 AND length <= 15

RETURN 2

inputAxles()

IF wheels==4 AND axles ==2

RETURN 4

inputWeight()

IF weight<2

RETURN 3

IF 2<weight<3.5

RETURN 5

IF weight>3.5

RETURN 6

Vehicle.setClass(INT vehicleClass)

vehicleClass = vehicleClass

Vehicle. identifyClass()

switch (vehicleClass)

CASE 0 :

RETURN

CASE 1 :

OUTPUT Motorbike (Class 1)

BREAK

CASE 2 :

OUTPUT Car (Class 2)

BREAK

CASE 3 :

OUTPUT Car with Trailer (Class 3)

BREAK

CASE 4 :

OUTPUT Van (Class 4)

BREAK

CASE 5 :

OUTPUT Small lorry/bus (Class 5)

BREAK

CASE 6 :

OUTPUT Large Lorry (Class 6)

BREAK

## Class Diagram

|  |  |  |
| --- | --- | --- |
| TunnelTollCostUser |  | Vehicle |
|  | int: wheels  double: length  int: axles  double: weight  int: vehicleClass  scanner: input |
| main() |
| |  | | --- | | TunnelTollCost | | boolean: isWeekend | | double: getCharge(int vehicleClass)  double: getMonthlyPass(int vehicleClass)  double: getMonthlyCost(int vehicleClass, int weekendJourneys, int weekdayJourneys) | | int :getIntInput(int minValue)  double: getDoubleinput(double minValue)  void: inputWheels()  void: inputLength()  void: inputAxles()  void: inputWeight()  int: identify()  void: setClass(int vehicleClass)  void: identifyClass() |

# Testing

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Wheels | Length | Axles | Weight | Weekend | Vehicle Class | Cost | As Expected |
| 2 |  |  |  | True | 1 | 1.50 | Yes |
| 4 | 7 |  |  | True | 2 | 2.40 | Yes |
| 6 | 16 | 3 | 2 | True | 3 | 2.85 | Yes |
| 4 | 17 | 2 |  | False | 4 | 4.40 | Yes |
| 4 | 22 | 4 | 2.2 | False | 5 | 8.00 | Yes |
| 8 | 33 | 4 | 3.6 | False | 6 | 12.00 | Yes |
| 1 | Asks for greater number to be input | | | | | | Yes |

Please enter the number of wheels: 8

Please enter the Vehicle Length: 33

Please enter the Number of axles: 4

Please enter the Vehicle Weight: 3.6

Input true if travelling on a weekend (sat/sun) and false otherwise: false

A Large Lorry (Class 6) traveling on a weekday will be charged 12.00

Please enter the number of wheels: 1

The number must be greater than or equal to to 2

Please enter a different number: