

OBJECT ORIENTED PROGRAMMING

SECTION – C [WEDNESDAY OCTOBER 13, 2021: 2:00 PM – 5:00 PM]

ASSIGNMENTS – 07 (RP07)

CODE: ASSIGN07

NOTES:

- i) Create files with the following file naming conventions: If your roll number ends with **abc**, year of admission is **2019** and assignment code is **Assign07** then, use the file name as follows: **Assign072019abc.cpp** (use appropriate extension .cpp suitably).
*For example, if the roll number ends with 127; year of admission is 2019 & the assignment code is Assign07, then the file name should be **Assign072019127.cpp***
- ii) Strictly follow the file naming convention. Otherwise, it would attract a penalty up to 20%.

PROBLEM:

[Total Marks: 20]

You may choose C++ (.cpp) or JAVA (.java) to solve these problems using different inheritances

Note: Use Random numbers as necessary. Use only Public Derivation for inheriting the classes.

- a) [4 Marks] Define a class: **Vehicle** consisting of the following States and Behaviours:

States:

- i. **make:** brand of the car – enum(Tata, Ford, BMW, Rolls-Royce, Benz, Leyland, Isuzu)
- ii. **color:** color of the car – enum(Blue, Black, White, Grey, Silver)
- iii. **year:** the year in which the vehicle has been manufactured in [2010, 2021]
- iv. **Fuel:** either diesel or petrol variants
- v. **Engine:** this shows the capacity of the engines in real numbers [1.0, 3.0]

Behaviour:

- i. **isClimbUpHill():** The capacity of the vehicle to climb the hill (True, False)
 - ii. **getAccelerate():** This shows the acceleration capacity. This may use engine and fuel type as arguments. Acceleration can be defined as the ratio between the change in velocity over the change in time. You can assume the velocity as a function of using fuel type and engine capacity.
 - iii. **getNumSeats():** this is to specify the number of seats one can use to configure the vehicle.
- b) [4 Marks] Write a method to populate an array of n **Vehicles** where n in [5, 20] with the following state variables. Use random number generation for the state variables as defined above.
 - c) [3 Marks] Apply single inheritance to generate an array of n **Trucks**, each has a minimum of 2 doors, 2 seating capacity and 10 tons load capacity. Add additional features as you may require for the derived class. Truck will have more than 4 types (and consider this as an even number)
Vehicle ← Trucks
 - d) [3 Marks] Apply multilevel inheritance to generate an array of n **Cars** using:
Vehicle ← LMV ← Cars
Each car may have different doors (2 or 4) and different seating capacity. Add additional features as you may require for the derived class.
 - e) [4 Marks] Apply multiple inheritance to generate an array of n pickup trucks that has both features of cars and trucks with the towing capacity between [2, 5] tons. Add additional features as you may require for the derived class.
 - f) [2 Marks] Write a method in the base class to print the details of the vehicles including its behaviours: **void printDetails()**. Apply inheritance of this method in each derived class with the same name.