Lambton College | In Toronto

Final Exam

[Total Marks: 100]

Class: 2019F MAD 3004
Instructor: Mohammad Kiani

Evaluation: 35%
Due Date: Thursday, 24th October 2019, 17:00 PM

READ CAREFULLY ENTIRE QUESTION DESCRIPTION

Instruction:
Your assignment consists of three parts which each part completes the previous part. Your are provided with a main file that you are not allowed to modify that. You should complete the rest of the project so that it conforms to the provided files.

Part 1

The goal of this assignment is to manage the salaries of the employees of a software engineering company. An employee is characterized by his/her name (the field name must remain unchanged after its initialisation); his/her birth year; his/her age as a computed property which computes and returns his/her age regarding the current year and his/her year of birth; a monthly income and an occupation rate (the percentage of time worked monthly; for example, 80%). The employee has also a vehicle. There exist three types of employees according to the diagram:

Class **Employee** has three subclasses **Manager, Tester** and **Programmer.**

- Managers (class **Manager**) specifically characterized by the number of travelled days and by the number of new clients brought to the company.
- Testers (class **Tester**) characterized by the number of bugs they managed to solve while testing.
- Programmers (class **Programmer**) characterized by the number of projects they have completed.

Make sure all your classes are well encapsulated. To do a good encapsulation, try to use "Set" and/or "Get" to expose ONLY what needs to be exposed by each class (hint: for a good encapsulation, all properties MUST be private and accessed only via Get and Set methods).

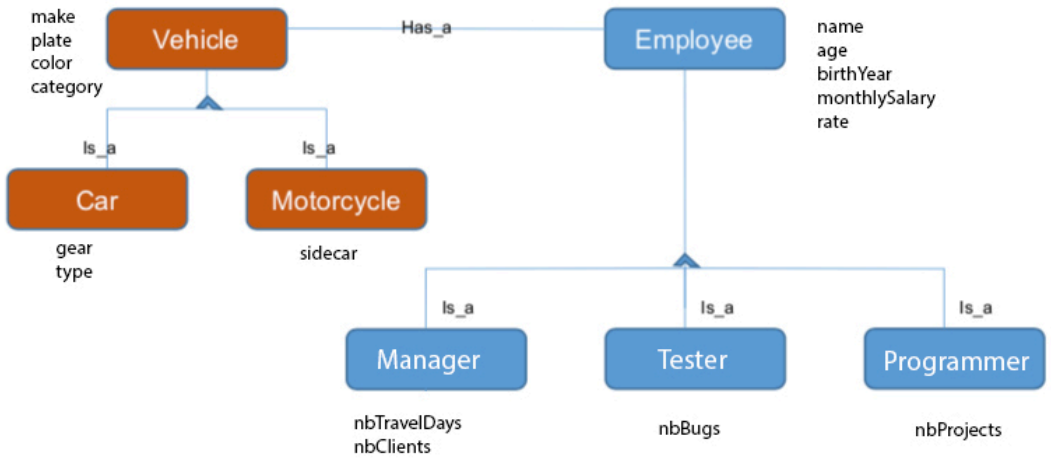
Start your work by equipping your classes with initializers enabling the initialisation of all their properties. The initializers must be compatible with the provided main part; the occupation rate will be 100% by default. Moreover, if an initializer receives as parameter an occupation rate lower than 10%, the effective occupation rate of the employee must be set to 10%. Similarly, if the occupation rate received by the initializer is greater than 100%, the effective occupation rate will be set to 100%. The initializer of an Employee will display the message:

```
We have a new employee:
<…>
```

(where <...> must correspond to what is displayed in the execution examples given below). This part must match the test part 1 in the execution example.

Part 2

In this part, you are supposed to assign a vehicle for each employee. Create a project according to the diagram below and develop the specified classes:



Here is the list of classes you see in the diagram:

- Class **Vehicle** contains four properties: make, plate, color and category; the category indicates if the vehicle in question is a 'sport or a 'family' vehicle.
- Class **Car** has two properties: 'gear' which indicates if the car is automatic or manual; and 'type' which indicates the type of the car i.e. sport, SUV, hatchback, minivan or sedan.
- Class **Motorcycle** has one property which indicates if the motorcycle has a sidecar.

Add a method *annualIncome()* (yearly income) which computes and returns the yearly income of an employee as follows:

- Each employee has a base yearly income computed as 12 times the monthly income multiplied by the occupation rate.
- For a manager a bonus of 500 dollars per client brought to the company is added as well as 100 dollars per day for the expenditure of the travelled days. (you can define two constants like GAIN_FACTOR_CLIENT equals to 500 and GAIN_FACTOR_TRAVEL equals to 100)
- For a tester, a bonus of 10 dollars per corrected bug is added (a constant can be used like GAIN_FACTOR_ERROR equals 10)
- For a programmer, a bonus of 200 dollars per completed project will be added (a constant can be defined GAIN_FACTOR_PROJECTS equals to 200)

Add all the necessary code so that the execution of the provided main produces the output given in the execution examples below.

Part 3

In this part, you are supposed to assign a contract for employees. Employees can have either a permanent or temporary contract. You do not modify the previous parts.

Permanent employees

A permanent employee is characterized by the following attributes:

- The number of days already worked in the month.
- The fixed monthly salary (it will be assumed that there are 20 working days per month).
- The number of children.
- Civil status (civil registration).
- The amount of the monthly premium for children.

The salary of permanent employees consists of the fixed monthly salary plus the children's bonus (if the employee is married!). The bonus will be multiplied by the number of children if the employee has more than one child.

The cumulative salary for a permanent employee is proportional to the number of days worked in the month.

Temporary Employees

Temporary employees are characterized by:

- Their hourly wages.
- The number of hours already done in the month.

Temporary employees are paid by the hour. For temporary employees, the cumulative salary is proportional to the hours worked.

An employee can be able to change his/her contract from permanent to temporary and vide versa. Refer to the provided code for your reference.

The accumulated salary in the category of origin will have to be calculated by means of a method, `accumulatedSalary`, well chosen. It will need to be converted to be compatible with its new category.

For example, a permanent employee, married with a child, transferred after 15 days of work with a monthly salary of 5000 dollars and a monthly child allowance of 450 dollars will have accumulated a salary equal to:

$$15 * (5000 + 450)/20 = 4'087,5 \text{ dollars}$$

Note: Test log is attached