Question 1 **[35 Marks]**

instructions:

- A starter code is provided for this question in **program1.cpp**
- For question (a) and (b), write your answers in the answer booklet provided.
- As for questions (c) and (d), modify the same program**, program1.cpp.**

Given a C++ program named **program1.cpp** that contains two classes named `Course`, and `Student`. This program is intended to store information about course enrollment by students. Then the user can search for a student by their matric number and print the information. Analyze the program and answer the following questions:

a. Based on the class declarations in the program, determine the type of relationship the two classes form? Provide the reason for your answer (4 marks)

b. Discuss how can `map` be usefull to develop such a program (6 marks)

c. Complete the class `Student` in the same **program1.cpp** by implementing of the following methods:

   i. `enrollToCourse().`

   This method will enroll the student to a course passed as a parameter. The course will be added the `courses` list.

   (2 marks)

   ii. `getEnrolledCount().`

   Returns the number of courses enrolled by the student

   (2 marks)

   iii. `getTotalCredit().`

   Returns the number of credits taken by the student

   (4 marks)

   iv. `printCourses().`

   Prints all the courses enrolled by the student

   (2 marks)

d. Complete the main function based on the following tasks:

   i. Create the list of available courses using `vector`. You can create the list with hard-coded data. Use the sample data from the file **question1_data.txt**.

i. (4 marks)

ii. Create the list of students using `map` using student matric number as the key. Enroll the students to some courses. You can create the list with hard-coded data. Use the sample data from the file **question1_data.txt**.

(6 marks)

iii. Write the code that asks the user to enter a matric number of a student. Then search for the student based on the matric number and finally print out the summary information about the student and the list of courses he/she enrolls to. Figure 1 shows examples of expected result of the program. Text in bold indicates user input.

(5 marks)

```
Enter the matric number=> A16EC4041

Information of found student
============================
Name:Mario Max
Matric:A16EC4041
Number of courses enrolled:4
Total credit carried:11
List of courses enrolled
------------------------
Course: Programming Technique I   Credit=3
Course: Digital Logic   Credit=3
Course: Graduate Success Attributes   Credit=2
Course: Programming Technique I   Credit=3
```

Run 1

```
Enter the matric number=> A19EC4002

Information of found student
============================
Name:Anna Mull
Matric:A19EC4002
Number of courses enrolled:5
Total credit carried:15
List of courses enrolled
------------------------
Course: Programming Technique II   Credit=3
Course: Operating Systems   Credit=3
Course: Digital Logic   Credit=3
Course: Web Programming   Credit=3
Course: Software Engineering   Credit=3
```

Run 2

```
Enter the matric number=> A18EC4044

Information of found student
============================
Name:Jimmy Changa
Matric:A18EC4044
Number of courses enrolled:5
Total credit carried:16
List of courses enrolled
-----------------------
Course: Web Programming   Credit=3
Course: Object-Oriented Programming   Credit=4
Course: Application Development   Credit=4
Course: Final Year Project I   Credit=2
Course: Programming Technique II   Credit=3
```

Run 3

```
Enter the matric number=> A16EC4045

Information of found student
============================
Name:Wilma Mumduya
Matric:A16EC4045
Number of courses enrolled:0
Total credit carried:0
List of courses enrolled
```

Run 4

```
Enter the matric number=> A23EC0001

The students is not found
```

Run 5

Figure 1: Expected outputs from Program 1,

Question 2 **[65 Marks]**

instructions:

- No starter code is provided for this question . You will have to write the program from scratch.
- For question (a) and (b), write your answers in the answer booklet.
- As for questions (c), write a complete C++ program and save it in a single file named **program2.cpp**

Consider the UML class diagram in Figure 2 and answer the following questions:

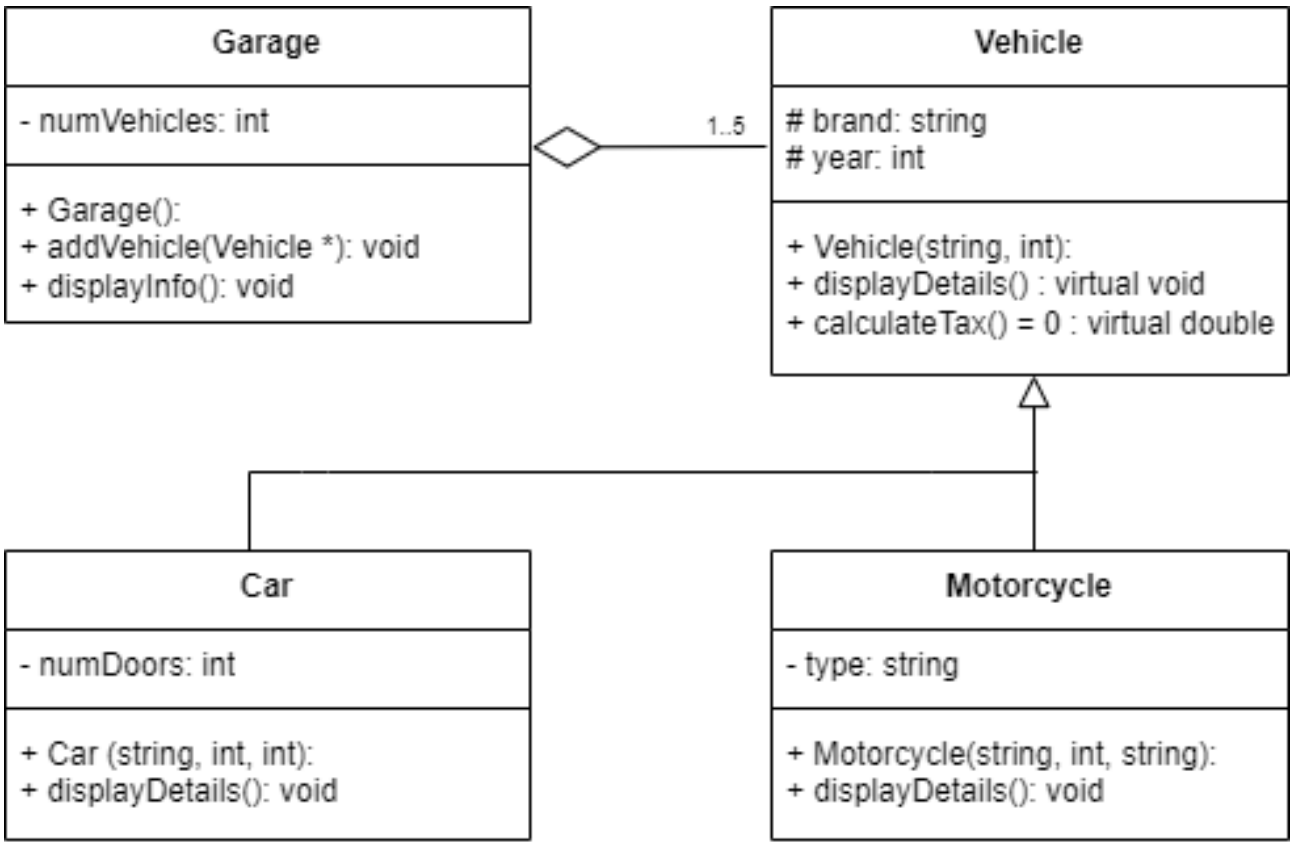

Figure 2: UML class diagram

a. Based on the class diagram, determine whether the diagram has an abstract class. Provide the reason for your answer. (5 marks)

b. Explain the difference between a pure virtual function and a virtual function in the class diagram?

(10 marks)

c. Write a complete C++ program based on the UML class diagram in **Figure 2**. Implement all the classes with their member variables (attributes) and member functions (methods) as shown in the diagram. The purpose of each function is as the name implies, and more explanations are provided below. Your program should be able to generate the output shown in **Figure 3**. Tasks to be accomplished for the program are as follows:

i. Define the class **`Vehicle`**. The cl ass has three member functions:

- Constructor with arguments: to initialize all the member attributes for the class.
- **`displayDetails()`**: to display a vehicle's brand and year. Please allow the function to be polymorphic.
- **`calculateTax()`** is a pure virtual function.

(7 marks)

ii. Define the class **`Motorcycle`** as a child of the class **`Vehicle`** . The class has three member functions:

- Constructor with arguments: to initialize all the member attributes for the class, including the parent's attributes.
- **`displayDetails()`**: to display a motorcycle's type, brand and year. The function should invoke the parent's **`displayDetails()`** function.
- **`calculateTax()`**: to return the motorcycle's tax, which is MYR50.

(9 marks)

iii. Define the **`Car`** class as a child of the **`Vehicle`** class. The class has three member functions:

- Constructor with arguments: to initialize all the member attributes for the class, including the parent's attributes.
- **`displayDetails()`**: to display a car's type, brand and the number of doors. The function should invoke the parent's **`displayDetails`** function.
- **`calculateTax()`**: to return the car's tax. The tax is calculated using the formula: 100 + (number of doors × 50).

(9 marks)

iv. Define the class **`Vehicle`**. The cl ass has three member functions:

- Constructor with arguments: to initialize all the member attributes for the class, including the parent's attributes.
- **`displayDetails()`**: to display a car's type, brand and the number of doors. The function should invoke the parent's **`displayDetails`** function.
- **`calculateTax()`**: to return the car's tax. The tax is calculated using the formula: 100 + (number of doors × 50).

(9 marks)

v. Define the **`Garage`** class. The class has three member functions:

- Default constructor: to initialize all the member attributes for the class with suitable values.
- **`AddVehicle():`** to assign the element in the array of **`Vehicle`** pointers with the passed argument. It also updates the number of vehicles.
- **`DisplayInfo()`**: to display the details of the vehicles in the garage.

(13 marks)

vi. Define the **`main`** function:

- Create a **`Garage`** object.
- Create an array of **`Vehicle`** pointers that dynamically allocates cars and motorcycles.Use sample data provided in the file **question2_data.txt**
- Add all the created vehicles to the **`Garage`** object.
- Display the details of all the vehicles in the garage.

(7 marks)

vii. Add appropriate statements in the method `AddVehicle()` of class `Garage` to handle the error if the garage is full using an exception approach. **Figure 4** shows the example of program output when the garage is full. (5 marks)

```
***** Garage Details *****

1. Car Details:
   Brand: Toyota
   Year : 2015
   Number of Doors: 4
   Tax  : MYR300
-------------------------

2. Car Details:
   Brand: BMW
   Year : 2019
   Number of Doors: 2
   Tax  : MYR200
-------------------------

3. Motorcycle Details:
   Brand: Honda
   Year : 2020
   Type : Sport
   Tax  : MYR50
-------------------------

4. Car Details:
   Brand: Nissan
   Year : 2018
   Number of Doors: 5
   Tax  : MYR350
-------------------------

5. Motorcycle Details:
   Brand: Harley-Davidson
   Year : 2017
   Type : Cruiser
   Tax  : MYR50
-------------------------
```

Figure 3: Program output

```
An error occurred: The maximum number of vehicles has been
reached!!
```

Figure 4: Sample output of error handling