Lab 3

Intake 49

1. Create an Abstract class **Student** that contains a method take_exam(), implement the method in the child classes PhdStudent and GradStudent in which PhdStudent takes exam by giving his final defense presentation while the graduate student gives a written paper.

2. Write an interface called **Movaable**, which contains 4 abstract methods moveUp(), moveDown(), moveLeft() and moveRight(). This interface must be overridden by two classes named MovaablePoint and MovaableCircle. Create necessary variables in the classes and implement the methods.

3. Create a **SavingsAccount** class. Use a static data member annualInterestRate to store the annual interest rate for each of the savers. Each member of the class contains a private data member savingsBalance indicating the amount the saver currently has on deposit. Provide member function calculateMonthlyInterest that calculates the monthly interest by multiplying the balance by annualInterestRate divided by 12; this interest should be added to savingsBalance. Provide a static member function modifyInterestRate that sets the static annualInterestRate to a new value. Write a program to test class SavingsAccount. Instantiate two different objects of class SavingsAccount, saver1 and saver2, with balances of $2000.00 and $3000.00, respectively. Set the annualInterestRate to 3 percent. Then calculate the monthly interest and print the new balances for each of the savers. Then set the annualInterestRate to 4 percent, calculate the next month's interest and print the new balances for each of the savers.

4. You are given an interface **AdvancedArithmetic** which contains a method named divisor_sum(int n). You need to write a class called MyCalculator which implements the interface. Divisor_sum function just takes an integer as input and return the sum of all its divisors. For example, divisors of 6 are 1, 2, 3 and 6, so divisor_sum should return 12. The value of n will be at most 1000.

5. Create an interface named **People.** This must include getfirstname() and getlastname() and displayname() methods. Two child classes named Student and Teacher must implement methods to display full name for an individual person.

6. Suppose you have a Train class. You have to keep track of how many trains have started their journey after 10 a.m. Write a code to implement the scenario.

7. Create an interface named **Account** including getName(), setName(), getPassword() and setPassword(). Create another interface named **Email** which includes getOtp(), setOtp(), and verifyEmail(). Create a class Person which must implement them. Complete the code.