

## Assignment # 1

**Submission Date: 16 October 2023**

### CLO-2

**Problem 1:** Create a Person class with two private attributes: name and age. The constructor initializes these attributes when creating a Person object. The compareAge method takes another Person object (otherPerson) as an argument and compares their ages, printing out the result.

- Create two Person objects, person1 and person2, with different names and ages.
- Call the compareAge method on person1 and pass person2 as an argument to compare their ages.
- The method compares the ages and prints a message indicating whether person1 is younger, older, or of the same age as person2.

**Problem 2:** Design a BankAccount class with the following characteristics:

Data Members:

- Account No of type String
- Account Title of type String
- Account Balance of type double
- Account Status of type boolean

Member Functions:

- A no-parameter constructor that initializes the account holder's name to "Unknown," account number to "N/A," and balance to 0.0.
- A parameterized constructor that takes the account holder's name, account number, and initial balance as arguments.
- A copy constructor that replicates another BankAccount object.
- Setter and getter methods for the account holder's name, account number, and balance.
- A method deposit(double amount) to add funds to the account balance.
- A method withdraw(double amount) to deduct funds from the account balance.
- A method transfer(BankAccount recipient, double amount) to transfer money from the current account to another account of the same class.

**Problem 3:** Rectangle Class

Design a Rectangle class with the following features:

- A no-parameter constructor that initializes the rectangle's width and height to 1.0.
- A parameterized constructor that accepts the width and height as arguments.
- A copy constructor that duplicates another Rectangle object.
- Setter and getter methods for the rectangle's width and height.
- A method calculateArea() to compute the area of the rectangle.
- A method calculatePerimeter() to calculate the perimeter of the rectangle.
- A method compareAreas(Rectangle otherRectangle) that compares the area of the current rectangle with another rectangle and prints the result.

#### **Problem 4: Employee Class**

Create an Employee class with the following functionalities:

Data Members:

- First Name of type String
- Last Name of type String
- Age of type int
- Salary of type double

Member Functions

- A no-parameter constructor that initializes the employee's name to "Unknown," employee ID to 0, and salary to 0.0.
- A parameterized constructor that accepts the employee's name, employee ID, and initial salary.
- A copy constructor that replicates another Employee object.
- Setter and getter methods for the employee's name, employee ID, and salary.
- A method giveRaise(double percentage) to increase the employee's salary by a given percentage.
- A method compareSalaries(Employee otherEmployee) that compares the salary of the current employee with another employee and prints the result.

#### **Problem 5: Create a ComplexNumber class with the following functionalities:**

- A no-parameter constructor that initializes the complex number to  $0 + 0i$ .
- A parameterized constructor that accepts real and imaginary parts as arguments.
- A copy constructor that duplicates another ComplexNumber object.
- Setter and getter methods for the real and imaginary parts.
- A method add(ComplexNumber other) that adds another complex number to the current complex number and returns a new ComplexNumber object with the result.
- A method multiply(ComplexNumber other) that multiplies another complex number with the current complex number and returns a new ComplexNumber object with the result.
- A method conjugate() that returns the conjugate of the current complex number as a new ComplexNumber object.

**Problem 6:** Suppose you operate several hot dog stands distributed throughout town. Define an Encapsulated class named HotDogStand that has an instance variable for the hot dog stand's ID number and an instance variable for how many hot dogs the stand has sold that day. Create a constructor that allows a user of the class to initialize both values. Also create a method named justSold that increments by one the number of hot dogs the stand has sold. The idea is that this method will be invoked each time the stand sells a hot dog so that you can track the total number of hot dogs sold by the stand. Write a main method to test your class with at least three hot dog stands that each sell a variety of hot dogs. Use get function to display the hot dogs sold for each object.

**Problem 7:** Create a class — Distancel with two constructors (no argument, and two argument), two data members (feet and inches) . Create setter, getter and display method. Create a method that adds two Distance Objects and returns the added Distance Object.