Sl. No.	Assignment	B-1	B-2
01.	Assignment 01: Introduction to Java Programming	17.10.2022	13.10.2022
02.	Assignment 02: Basic Java Programs implementing loop control structures	31.10.2022	20.10.2022
03.	Assignment 03: Solving Pattern problems using loop control structures	07.11.2022	27.10.2022
04.	Assignment 04: Classes and Objects, Constructors, Arrays and Array of Objects.	14.11.2022	03.11.2022
05.	Assignment 05: Inheritance	21.11.2022	10.11.2022
06.	Assignment 06: Method Overloading, Abstract classes, Interface and Polymorphism	28.11.2022	17.11.2022
07.	Assignment 07: Package	12.12.2022	24.11.2022
08.	Assignment 08: Access Specifiers and DSA implementation	19.12.2022	01.12.2022
09.	Assignment 09: Exception Handling	26.12.2022	15.12.2022
10.	Assignment 10: Multithreading	02.01.2023	22.12.2022
11.	Assignment 11: Implementation of AWT	09.02.2023	29.12.2022

Assignment 01

Sl. No.	Question
1.	Write a Java program to check whether an input number is Palindrome or not.
2.	Write a Java program to check whether an input number is Armstrong or not.
3.	Write a Java program to find factorial of a number.
4.	Write a Java program to find GCD and LCM of two numbers.
5.	Write a Java program to check whether the input number is Prime or not.

Assignment 02

Sl. No.	Question
1.	Write a Java program to find all prime numbers between given pair of range.
2.	Write a Java program to find all Armstrong numbers between a given pair of range.
3.	Write a Java program to find all Palindrome numbers within a given pair of range.
4.	Write a Java program to print the Fibonacci series up to a given range.
5.	Write a Java program to check if the square root of reverse of a number is equal to reverse of that original number.
6.	Write a Java program to display the multiplication table up to a given number.

Assignment 03

Print the following pattern:

1.
```
*
* *
* * *
* * * *
* * * * *
```

2.
```
        1
      2 1
    3 2 1
  4 3 2 1
5 4 3 2 1
  4 3 2 1
    3 2 1
      2 1
        1
```

3.
```
        1
      1 2 1
    1 2 3 2 1
  1 2 3 4 3 2 1
1 2 3 4 5 4 3 2 1
  1 2 3 4 3 2 1
    1 2 3 2 1
      1 2 1
        1
```

4.
```
        *
      *   *
    *       *
  *           *
*               *
  *           *
    *       *
      *   *
        *
```

5.
```
          A
        B A B
      C B A B C
    D C B A B C D
  E D C B A B C D E
```

6.
```
1
2  3
6  5  4
7  8  9  10
15 14 13 12 11
```

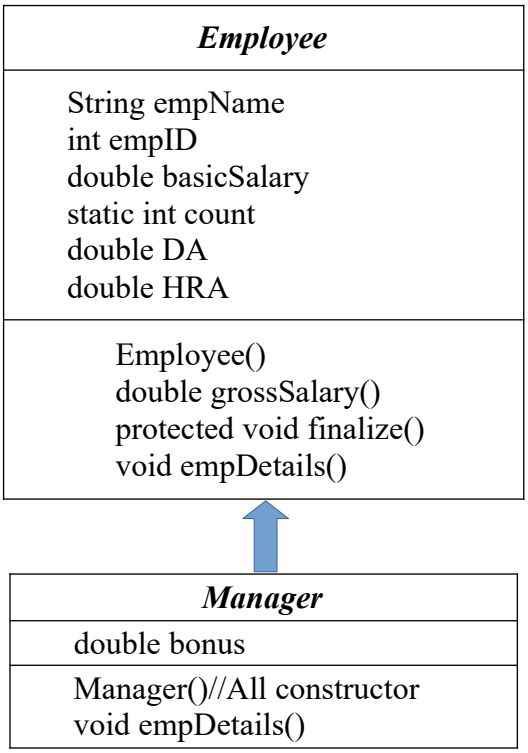
Assignment 04

Sl. No.	Question
1.	Design a class ***Time*** having data members ***hour***, ***minute*** and ***second***. It should also have methods ***displayTime()*** to display the time in HH:MM:SS format and ***addTime()*** to add two Time objects. Test these methods by creating a ***main()*** method in another class. Implement it by using parameterized constructor to initialize its data members.
2.	Create a class ***Complex*** for performing arithmetic with complex numbers. Complex numbers have the form ***realPart + imaginaryPart * i***. Write a program to test your class. Use floating point variables to represent the private data of the class. Provide a constructor that enables an object of this class to be initialized when it is declared. Provide a no-argument constructor with default values in case no initializers are provided. Provide public methods that perform the following operations: a. Add two complex numbers: The real parts are added together and the imaginary parts are added together. b. Subtract two complex numbers: The real parts of the right operand is subtracted from the real part of the left operand, and the imaginary part of the right operand is subtracted from the imaginary part of the left operand. c. Print complex numbers in the form ***(realPart, imaginaryPart)***
3.	Design a class named ***Rectangle*** to represent a rectangle. The class contains: a. Two double data fields names width and height that specify the width and height of the rectangle. The default values are 1 for both width and height. b. A no-argument constructor that creates a default rectangle. c. A constructor that creates a rectangle with the specified width and height. d. A method named ***getArea()*** that returns the area of this rectangle. e. A method named ***getPerimeter()*** that returns the perimeter.
4.	Write a program to find the sum and average if values of an array.
5.	Write a program to implement linear search. If present, display the number of times it is present.
6.	Design a class ***Student*** having data members age and mark. It should have methods ***input()*** to take details of student and ***show()*** method to display the details of the student. Test these methods by creating an array of object of ***Student***.

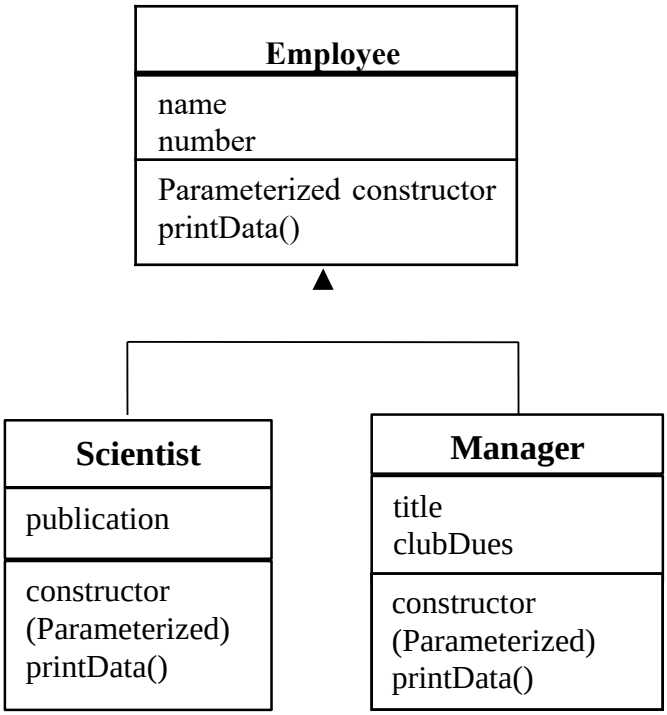
Assignment 05

Sl. No.	Question
1.	Write a Java program to implement Single Inheritance.
2.	Write a Java program to implement Multi-Level Inheritance.
3.	Write a Java program to implement Hierarchical Inheritance.
4.	Write a Java program to find the largest number between two numbers using Single Inheritance.
5.	Write a Java program to find the smallest number between three numbers using Multi-Level Inheritance.
6.	Write a Java program to achieve Multiple Inheritance in Java through Interface.

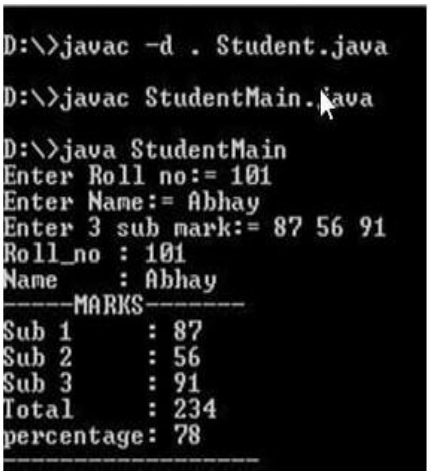
Assignment 06

Sl. No.	Question
1.	Write a Java program to implement the Method Overloading concept.
2.	Design the following class hierarchies with appropriate driver class and main() method. You may add more member functions if required. Also add the parameterized constructor. 
3.	Use the concept of abstract class and abstract method for the above class. Create an object reference of ***Employee*** class using which access the necessary member function of ***Manager*** class.
4.	Design a interface ***Shape*** and implement in ***Square, Rectangle*** and ***Triangle*** class from ***Shape*** class. By creating the object reference of ***Shape*** to access the member ***area()*** of different shape.

5.	Design following class hierarchies with appropriate main() function. You may add more member functions if required.



Assignment 07

Sl. No.	Question
1.	Create a package ***btech*** which has one class ***Student***. Accept student details through parameterized constructor of Student class. Write a method ***display()*** to display the student details. Create another class ***Test*** containing the ***main()*** method which will use the package btech and calculate total marks and percentage of marks. One sample output is shown below. 
2.	Create a sub-package called ***arithmetic*** under the package ***btech***. The ***arithmetic*** package should contain a class ***MyMath*** having methods to deal with different arithmetic operations (addition, subtraction, multiplication, division and mod). Create a class ***Test*** containing the main method which will use the methods of sub-package ***arithmetic***.
3.	Create a sub-package named ***shapes*** under ***package*** org. Create some classes in the package representing some common geometric shapes like ***Square***, ***Triangle***, ***Circle*** and so on. The classes should the ***area()*** and ***perimeter()*** methods in them. Compile the package. Use this package to find area and perimeter of different shapes as chosen by the user.
4.	Write a program to create a package named ***folder1*** to take a number and create another package name ***folder2*** in which you find the largest between two numbers.

Assignment 08

Sl. No.	Question
1.	Write a Java program to implement protected and public access specifiers.
2.	Write a Java program to implement Stack using class and object.
3.	Write a Java program to implement Linear Queue by using class and object.
4.	Write a Java program to implement Circular Queue by using class and object.

Assignment 09

Sl. No.	Question
1.	Write a Java program to read two numbers ***a*** and ***b*** and calculate ***a/(a-b)***. The program should check the value of ***a-b*** before dividing with ***a***, it should throw an exception if ***a-b*** is zero. In the exception handler the program should display appropriate message to the user.
2.	Write a class ***Account*** with the following properties and methods: Properties: String ***name***, int ***acc_no***, double ***balance*** Methods: void ***deposit(int num)***, void ***withdraw(int num)***, void ***transfer(Account acc1, Account acc2, int amt)*** Assume that an account needs to have a minimum balance of 500. If an attempt is made to withdraw or transfer, which results in balance going below 500, throw a user-defined exception called ***MinimunBalanceException***. Use throw and throw wherever necessary.
3.	Write a Java program to compare names of two person (first name and last name) and return result as: a. Fully matched if both first and last names are same b. Same first names if only first name matches c. Same last names if only last name matches d. No match otherwise e. **NameFormatException** if any of the person name has any illegal entry i.e. numbers within name or name consisting of either one part or three parts.
4.	Write Java programs to implement all cases of exception handling: a. ArithmeticException b. ArrayIndexOutOfBoundsException c. NullPointerException d. NumberFormatException
5.	Write Java programs to implement exception handling by using: a. ***try*** block and multiple ***catch*** block b. Nested ***try-catch*** block c. ***finally*** block d. ***throw*** keyword e. ***throws*** keyword f. User-defined Exception/ Custom Exception g. Generic Exception

Assignment 10

Sl. No.	Question
1.	Write a Java program to implement Multithreading by extending ***Thread*** class.
2.	Write a Java program to implement Multithreading by implementing ***Runnable*** interface.
3.	Write a Java program to implement ***sleep()***.
4.	Write a Java program to implement ***join()***.
5.	Write a Java program to get and assign name to a thread.

Assignment 11

Sl. No.	Question
1.	Write a Java program to create a button by implementing AWT(Abstract Window Toolkit).
2.	Write a Java program to create a Calculator by implementing AWT.