Siliguri Institute of Technology

Sukna, Siliguri-734009

Lab Manual

Object Oriented Programming with JAVA

Subject Code: **MCAN-293**

Bidyut Das
Assistant Professor
Department of MCA

Code: MCAN-293	Paper: Object Oriented Programming Lab using JAVA	
Contacts Hours / Week: 4	Total Contact Hours: 40	Credit: 2

Course Outcome:

After successful completion of this course, students will be able to:

- ✓ Apply object-oriented principles or features in software design process to develop Java programs for real life applications.
- ✓ Reduce the complexity of procedural language by employing different OOP technologies for developing robust and reusable software.
- ✓ Develop programs using stream classes for various I/O operations and design concurrent.
- ✓ Design graphical user interface to develop user interactive applications.

UNITS	COURSE CONTENT
1	Assignments on class, constructor, overloading, inheritance, overriding.
2	Assignments on wrapper class, arrays.
3	Assignments on developing interfaces- multiple inheritance, extending interfaces.
4	Assignments on creating and accessing packages.
5	Assignments on multithreaded programming
6	Assignments on applet programming

Assignment 1

[Assignment on class, constructor, overloading.]

1. Write a Java program that calculates factorial of a number.
2. Write a program to swap two values using object reference. Your program should have a swap method.
3. Write java program to print Biggest of 3 Numbers using Logical Operators.
4. Write a java program to print first 10 numbers in Fibonacci series. Also print their sum.
5. Write a java program to print Factorial of a given number using recursion.
6. Write a java program to print all natural prime numbers up to the given prime number.
7. Write a java program to print sum of n terms in the series 1/1! +1/2!+1/3!.....
8. Write A Java Program to print Quadratic roots using command line arguments.
9. Write a Java program that prints the season name corresponding to its month number using If-else and switch- case statements.
10. Write a Java Program to define a class, describe its constructor, overload the Constructors and instantiate its object.
11. Write a Java Program to define a class, define instance methods and overload them and use them for dynamic method invocation.
12. Write a Java Program to demonstrate use of nested class.

Assignment 2

[Assignments on wrapper class, arrays & inheritance.]

1. Write a Java Program to implement array of objects.
2. Write a Java Program to implement bubble sort .
3. Write a Java Program to find the largest and smallest element from an array.
4. Write a Java Program to find addition of two matrices.
5. Write a program to create a class Student2 along with two method getData(),printData() to get the value through argument and display the data in printData. Create the two objects s1 ,s2 to declare and access the values from class STtest.
6. WAP using parameterized constructor with two parameters id and name. While creating the objects obj1 and obj2 passed two arguments so that this constructor gets invoked after creation of obj1 and obj2.

7. Write a program in java to generate an abstract class A also class B inherits the class A. generate the object for class B and display the text "call me from B".

8. Write a java program in which you will declare two interface sum and Add inherits these interface through class A1 and display their content.

9. Write a java program in which you will declare an abstract class Vehicle inherits this class from two classes car and truck using the method engine in both display "car has good engine" and "truck has bad engine".

10. Write a Java program that creates a Class, namely Student.
i.Ensure that Age instance variable of the Class is never accessed directly, and its value is never less than 4 and greater than 40 for any Object of the Class (use methods to validate and assign the value).
ii.Ensure that the constructor always assigns a unique value to Enrollment_No instance variable for every Object of the Class (use a static class variable for counting objects, say Object_Counter).
iii.Ensure that when an Object is removed, the Object_Counter is automatically decremented (use finalize()), and whenever required the variable can only be accessed using a method even without an Object reference (make the counter private and use a static method to access it).

11. Write a Java program that creates a Class namely A that has a private instance variable and method, a protected instance variable and method, a default instance variable and method, and a public instance variable and method. Create another Class say B that inherits from A.
i.Show that all except private members are inherited.
ii.Show that an inherited instance variable can be shadowed (with the same or weaker access visibility) but can be accessed using super keyword in the sub-class.
iii.Show that the reference variable of type A or B can't access an overridden method of A in the Object of B.

12. Write a JAVA Program to demonstrate Constructor overloading and Method overloading.
13. Write a JAVA Program to demonstrate Dynamic method dispatch.
14. Write Java programs that use both recursive and non-recursive functions for implementing the following searching methods:
(a) Linear search
(b) Binary search

15. Write Java programs to implement the List ADT using arrays and linked lists.

16. Write Java programs to implement the following using an array.
(a)Stack ADT
(b)Queue ADT

17. Write Java programs to implement the following using a singly linked list.
(a)Stack ADT
(b)Queue ADT

Assignment 3

1. **Two Sum**

Given an array of integers nums and an integer target, return indices of the two numbers such that they add up to target.

You may assume that each input would have exactly one solution, and you may not use the same element twice.

You can return the answer in any order.

Example 1:

Input: nums = [2,7,11,15], target = 9
Output: [0,1]
Explanation: Because nums[0] + nums[1] == 9, we return [0, 1].

Example 2:

Input: nums = [3,2,4], target = 6
Output: [1,2]

Example 3:

Input: nums = [3,3], target = 6
Output: [0,1]

---

2. 3Sum

Given an integer array nums, return all the triplets [nums[i], nums[j], nums[k]] such that i != j, i != k, and j != k, and nums[i] + nums[j] + nums[k] == 0.

Notice that the solution set must not contain duplicate triplets.

Example 1:

Input: nums = [-1,0,1,2,-1,-4]
Output: [[-1,-1,2],[-1,0,1]]

Example 2:

Input: nums = []
Output: []

Example 3:

Input: nums = [0]
Output: []

---

3. 3Sum Closest

Given an integer array nums of length n and an integer target, find three integers in nums such that the sum is closest to target.

Return the sum of the three integers.

You may assume that each input would have exactly one solution.

Example 1:

Input: nums = [-1,2,1,-4], target = 1
Output: 2
Explanation: The sum that is closest to the target is 2. (-1 + 2 + 1 = 2).

Example 2:

Input: nums = [0,0,0], target = 1
Output: 0

---

4. Write a program to demonstrate use of implementing interfaces.
5. Write a program to demonstrate use of extending interfaces. import java.lang.*;
6. Write a program to implement multiple inheritance in java.

Assignment 4 [String]

1. Letter Combinations of a Phone Number

Given a string containing digits from 2-9 inclusive, return all possible letter combinations that the number could represent. Return the answer in any order.
A mapping of digit to letters (just like on the telephone buttons) is given below. Note that 1 does not map to any letters.

Example 1:
Input: digits = "23"
Output: ["ad","ae","af","bd","be","bf","cd","ce","cf"]
Example 2:
Input: digits = ""
Output: []
Example 3:
Input: digits = "2"
Output: ["a","b","c"]

---

2. **Longest Common Prefix**

Write a function to find the longest common prefix string amongst an array of strings.

If there is no common prefix, return an empty string "".

Example 1:

Input: strs = ["flower","flow","flight"]
Output: "fl"

Example 2:

Input: strs = ["dog","racecar","car"]
Output: ""
Explanation: There is no common prefix among the input strings.

---

3. **Password Checker**

You are given a function.
int CheckPassword(char str[], int n);
The function accepts string str of size n as an argument. Implement the function which returns 1 if given string str is valid password else 0.
str is a valid password if it satisfies the below conditions.

– At least 4 characters
– At least one numeric digit
– At Least one Capital Letter
– Must not have space or slash (/)
– Starting character must not be a number

Assumption:
Input string will not be empty.

Example:

Input:
aA1_67
Output:
1

Sample Input:
a987 abC012
Output:
0

4. Java program to find all permutations of a given String using recursion.

For example, given a String "XYZ", this program will print all 6 possible permutations of input e.g. XYZ, XZY, YXZ, YZX, ZXY, XYX

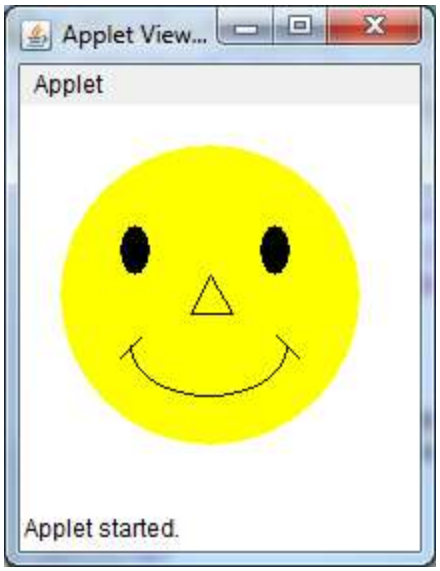
Assignment 5 [Multithreading]

1. Write a java program in which thread sleep for 5 sec and change the name of thread.
2. Write a java program for multithread in which user thread and thread started from main method invoked at a time each thread sleep for 1 sec.
3. Write a java program for to solve producer consumer problem in which a producer produces a value and consumer consume the value before producer generate the next value.
4. Write a java program for to solve printer synchronization problem in which all the jobs must be completed in order.
5. Write a program to implement the concept of threading by extending Thread Class

6. Write a program to implement the concept of threading by implementing Runnable Interface
7. Write a JAVA program using Synchronized Threads, which demonstrates Producer Consumer concept.

Assignment 6 [Applet]

1. Write an Applet code to draw the following smiley face



2. Write a program using Applet to display a message in the Applet.
3. Write a Java Program to demonstrate Mouse events

Assignment 7 [Exception Handling]

1. Write a program to implement the concept of Exception Handling using predefined exception.
2. Write a program to implement the concept of Exception Handling by creating user defined exceptions.

Assignment 8 [File Handling]

1. Write a java program to create a file and write the text in it and save the file.
2. Write a java program to read a file and display the content on screen.
3. Write a java program to create a folder.
4. Write a java program to rename a file.
5. Write a java program in which data is read from one file and should be written in another file.name of both file is given through command line arguments.
6. Write a java program in which data is read from one file and should be written in another file line by line.
7. Write a java program to read the the file using BufferReader.