SYMBIOSIS INTERNATIONAL (DEEMED UNIVERSITY)
(Established under section 3 of the UGC Act 1956)
Re - accredited by NAAC with 'A' Grade
Founder: Prof. Dr. S. B. Mujumdar, M.Sc.,Ph.D. (Awarded Padma Bhushan and Padma Shri by President of India)

Sub Committee - Specialization for Curriculum Development

Under Graduate

Faculty: Engineering

Sub-Committee (Specialization) : AIML

Course Name: OOPs with Java Lab

Catalog Code: 0701260403

Course Credit : 1

Course Level: 2

The students are able to:

1. Understand and demonstrate the fundamentals of object-oriented programming in Java, like defining classes, objects, invoking methods etc.
2. Demonstrate the concept and usage of inheritance, interface and packages in Java.
3. Illustrate practically the concept of exception handling in Java.
4. Implement Database connectivity in Java.
5. Demonstrate the concept of Multithreading in Java.

Pre-learning:

Knowledge of programming languages C and C++ is desirable

Course Outline

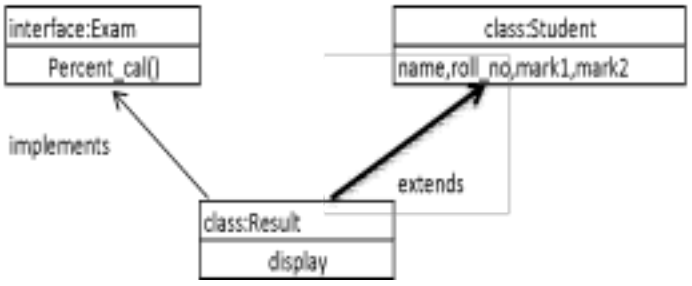
S.No.	Topic	Hours
1	Part1: Implement a menu-driven Java program (like fib or factorial) to implement these input methods in java (command line args, Scanner, BufferedReader, DataInputStream, Console)	2
	Part2: Implement a simple menu driven calculator in java to implement add, sub, mul, div, sqrt, power, mean, variance. Implement a separate Calculator class to include all related function inside that class. (mean calculation : program reads numbers from the keyboard, summing them	2

	in the process until the user enters the string “end”. It then stops input & displays the avg. of numbers)	
2	Part1: W.a.p that declares two arrays named ‘even’ and ‘odd’. Accept numbers from the user and move them to respective arrays depending on whether they are even or odd. Part2: Implement a java function that finds 2 neighboring numbers in an array with the smallest distance to each. The function should return the index of the 1st number. Part 3: Write a Java program to convert an array into ArrayList and vice versa.	4
3	Write a menu-driven Java Program to study the concepts of classes, array of objects, instance members, constructors in java. Assignment description: Create a Student class describing attributes of a student like prn, name, DoB, marks etc. Create an array of objects of Student class and perform operations like: Add students, Display, Search (by prn, by name, by position), Update/Edit and Delete.	2
4	Write a menu-driven Java Program for the following: There are 52 cards in a deck, each of which belongs to one of four suits and one of 13 ranks. Represent a deck of cards as an array of Objects	2

5	Implement the generic Shapes class as an interface s so that we can implement concrete classes like circle, triangle, rectangle class from it. 1) Write an abstract class Shape – Protected Data members: dim1, dim2, dim3, numSides – Constructor: initialize numSides and dimensions, zero and parameterized constructors – Concrete method: getmethod for numSides – Abstract methods: getArea(), getPerimeter() 2) Write a concrete subclass Rectangle – Protected Data members: dim1, dim2, 3) Write a concrete subclass RtTriangle – Protected Data members: dim1, dim2, 4) Write a concrete subclass Circle – Protected Data members: dim1, dim2, 5) In another class, write a main method to define a Rectangle and a Triangle and Circle using concepts of Dynamic method Dispatch	2

	Part2: In this exercise, take an abstract class which is defined below and develop two classes. The abstract class represents the basic building block for employees in a personnel database. The code is shown below:
	
	abstract class Employee {
	private String name, address;
	protected int basicSalary;
	
	public String getName(){
	return name;
	}
	
	public String getAddress(){
	return address;
	}
	
	public int getBasicSalary(){
	return basicSalary;
	}
	
	public void setAddress(String add){
	address = add;
	}
	
	public void setName(String nm){
	name = nm;
	}
	
	public void setBasicSalary(int sal){
	basicSalary = sal;
	}
	public abstract int getMonthlySalary();
		2

	} The class contains three instance variables which hold the name, address and basic yearly salary of an employee. Aim of this exercise Generate concrete classes from an abstract class: ● Copy the code above into the file Employee.java in a folder. Make this class public. ● Write the code for a class NormalEmployee which extends the class above. This class should have a single method which calculates themonthly () salary for an employee. Compile the class. ● Write the code for a class BonusEmployee which extends the class Employee.java. This class describes an employee who has a monthly bonus added to their monthly salary. Compile the class ● Create a fourth file which tests the implementation of NormalEmployee and BonusEmployee files by creating suitable objects.	
6	Part 1: An implementation of IntStack (integer stack) that uses fixed storage as well as "growable" using interface. Create a user defined package “pkg_Stack” where the interface is stored. The other two complete classes will need to import the package ‘pkg_Stack’ and then use it. Part 2: Program to implement the following Multiple Inheritance.  	4
7	Write a Java application that will be able to add, subtract, multiply, divide, compare, convert to floating point, and find absolute value for rational numbers, with exception handling	2

8	Part 1 : Write a Java Program to find the factorial of 'n' integers (as command line arguments CLA). Write your own exception "MyExcep" to validate integer values to be in certain range.	2

	Part 2: Define an exception class called "NOMATCHEXCP" that is thrown when the string from keyboard is not equal to "India". Write a Demo program using try-catch block that shows the use of this user-defined exception. class NOMATCHEXCP should have a parameterized constructor and the exception message generated should show the line number and the erroneous String that was inputted by the user.	2
9	Write a Java Program demonstrating the database connectivity in java.	2
10	Write a Java Program for Thread Creating and use of its various methods.	2

Pedagogy

1. Interactive teaching and discussions in lab
2. Lab sessions using software tools like JDK, NetBeans , EditPlus etc
3. Mini Project

Books Recommended

1. "Java 2: The Complete Reference", 3rd Edition, Patrick Naughton , Herbert Schildt Osborne Publishing, (1999)
2. "Programming With Java: A Primer", 3rd Edition, E. Balagurusamy, Tata McGraw - Hill Education (2008)
3. "Java How to Program", 9th Edition, Deitel and Deitel, Prentice Hall
4. "Core Java: An Integrated Approach", 1st Edition, R. Nageswara Rao , DreamTech Publication(2008)

Suggested Assessment/ Evaluation Methods

A) Continuous Assessment

a) Lab Test
b) Lab Assignments
c) Quizzes

Course Outcomes:

Students will be able to:

1. Implement object-oriented concepts using Java.
2. Develop reusable programs using the concepts of inheritance, polymorphism, interfaces, and

packages.

3. Implement Java programs to implement exception handling concepts.
4. Develop Java programs that access and manipulate data from databases.
5. Develop multithreading concepts in Java

Benchmarked against similar courses in other national/ international universities /organizations

S. No.	Name of the Course	Name of University where it is offered
1	Practical Course based on Object Oriented Programming using Java - I	University of Pune
2.	Programming Methodology	Stanford Engineering

Sub-specialization committee

Name of Member	Kalyani Kadam	Pooja Kamat	Dr. Sonali Tidke
Designation	Assistant Professor (CS, IT, and AIML)	Assistant Professor (CS)	Associate Professor (CS)
Org. / Inst.	SIT	SIT	SIT
Signature			

Name of Expert:

Name of Member	Mr. Vinod Satpute
Designation	Specialist Analyst, CDC,
Org. / Inst.	NetCracker Technologies Private Limited
Signature	

Signature of Dean:

Date: