Merit, Quality & Excellence
SUKKUR INSTITUTE OF BUSINESS ADMINISTRATION
DEPARTMENT OF COMPUTING
Date: 01.14.24

ABET -2000
COURSE BINDER

CSC-122

Object Oriented Programming

PREPARED BY

Dr. Sher Muhammad Doudpota

Dr. Ghulam Mujtaba Sheikh

Dr. Fahim Akhtar Rajput

Spring 2024

Course Syllabus (ABET -2000 Format):

CSC-122 – Object Oriented Programming
Spring 2024

Course Description	This course teaches object-oriented programming to those who have learnt basic programming concepts and are ready to learn in-depth programming. It focuses on object oriented programming using Java. In just a few years, java grew from a concept into one of the world's dominant computer language. The main topics discussed are: Moving from C to Java, Class design and objects, Data Abstraction, Encapsulation, Polymorphism and Inheritance, Interfaces, Exceptional Handling, Multi-Threading, GUI Programming and Java Database Connectivity.
Prerequisite	Introduction to Programming (Using C Language)
Textbook	1. Java 2 Complete Reference by Herbert Schildt, Latest edition 2. Beginning Java 2 by Ivor Horton, Latest edition 3. Absolute Java by Walter Savitch and Kenrick Mock, Fifth Edition
Course Learning Objectives (C.L.O.)	1. Student should know about the object-oriented programming 2. Become familiar with breaking down a problem into objects rather than procedures 3. Inheritance and polymorphism 4. Generic programming and interface 5. Object oriented software development 6. Learn what is available off the shelf to facilitate Java development (tools, libraries) 7. Graphic user interface, event handling, and animation 8. Socket programming 9. Student should have good command over the syntax of JAVA 10. Student should submit a project designed by him/her-self by properly implementing all the concepts learned during semester

<table>
<tr><td rowspan="3">Course Relationship to Program Outcomes</td><td colspan="12">Program Outcomes for Computer Science</td></tr>
<tr><td rowspan="2">Highest attainable level of Learning</td><td>a</td><td>b</td><td>c</td><td>d</td><td>e</td><td>f</td><td>g</td><td>h</td><td>i</td><td>j</td><td>k</td></tr>
<tr><td>1</td><td>2,6</td><td>1</td><td>1</td><td>1,10</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1,10</td><td>1,10</td></tr>
<tr><td>Class/ Lab Schedule</td><td colspan="12">Theory – 3 classes in a week
Laboratory – 3 classes in a week
(Class duration is 50 Minutes)</td></tr>
<tr><td>Instructional Methods</td><td colspan="12">Lectures – Tutorials – Home works – Mid. Exams – Final Exam.</td></tr>
<tr><td>Instructor</td><td colspan="12">Dr. Sher M. Doudpota, Dr. Fahim Akhtar Rajput, Dr . Ghulam Mujtaba Sheikh</td></tr>
</table>

CSC-122
Object Oriented Programming

Course Learning Objectives (C.L.O.)	Program Outcomes										
	a	b	c	d	e	f	g	h	i	j	k
1. Student should know about the object oriented programming										M	M
2. Become familiar with breaking down a problem into objects rather than procedures				L							
3. Inheritance and polymorphism											L
4. Generic programming and interface											L
5. Object oriented software development											L
6. Learn what is available off the shelf to facilitate Java development (tools, libraries)		M									
7. Graphic user interface, event handling, and animation											L
8. Java Database Connectivity											L
9. Student should have good command over the syntax of JAVA											L
10. Student should submit a project designed by him/her-self by properly implementing all the concepts learned during semester					L					L	H

H=3= High = Synthesis & Evaluation levels
M=2= Medium = Application & Analysis Levels
L=1= Low = knowledge & Comprehension Levels

Program Outcomes

(a) Knowledge of core computing, mathematics, and science fitting to the discipline.
(b) Ability to apply acquired knowledge in developing solutions using state-of-the-art methods, techniques, skills, and tools.
(c) An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
(d) An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs
(e) An ability to function effectively on teams to accomplish a common goal
(f) An understanding of professional, ethical, legal, security and social issues and responsibilities
(g) An ability to communicate effectively with a range of audiences
(h) An ability to analyze the local and global impact of computing on individuals, organizations, and society
(i) Recognition of the need for and an ability to engage in continuing professional development
(j) An ability to use current techniques, skills, and tools necessary for computing practice.

Course Calendar

Week. No	Topic	Topic Detail
1	Moving C to Java	Procedural vs. OOP Why OOP? History of Programming Languages Why JAVA? History of JAVA Buzzwords JDK, JRE and JVM
2	Introduction to OOP	Programming Paradigm Abstraction Three OOP Principles Encapsulation Inheritance & Polymorphism Together
3	Introduction to Java	First Java application A second Short Example Two Control Statements (if and for) Using Block Code Lexical Issues Programming Style Java Documentation
4	Fundamental Elements of Language	Fundamental Elements of Java (Datatypes, Variables and Arrays) Primitive Types Literals String Literal Escape Sequence Scope and lifetime of Variables Type Conversion and Casting Automatic Type Promotion in Expressions Arrays (One Dimension and Multi Dimension) Uneven Multidimensional Arrays
5.	Operators	Operators (Arithmetic, Bitwise, Relational and Logical) Ternary operator Control Statements (Selection, iteration, jump) Introduction to classes
7.	Classes & Methods	Constructor new Operator Parameterized Constructor this keyword Instance Variable Hiding Garbage Collection finalize () Method

8.	**Encapsulation:** Closer look at Classes, Methods, Access Controllers & References	Methods Overloading Methods Overloading Methods – Automatic Conversion Overloading Constructors Using Objects as Parameters Object to its Constructor as Parameter A Closer Look at Argument Passing Arguments in JAVA Call by value Call by Reference Returning Objects Recursion Access Control (Specifiers: public, private protected) Understanding static and final Nested Classes Inner Class within any block scope
9.	Inheritance	Inheritance Extends Supper & Subclass Advantages of Inheritance Extension in existing class Reference a Subclass Object Using *super* Multilevel Hierarchy Constructors Call Method Overriding
10.	Polymorphism	Method Overloading Using Abstract Classes 3 usage of final Packages and Interfaces Packages Compilation & Executing
11	Packages & Interfaces	Importing Packages Creating & Importing packages Interfaces Defining an interface Implementing Interfaces Accessing through references Polymorphic Methods Partial Implementation of an interface Interfaces Can Be Extended
12.	Exception Handling and File IO	Exception Handling Uncaught exceptions Try, catch Throw, throws and final File I/O Streams Simple file I/O Create a file Add the record to the file Read from existing file

14.	Graphical User Interface	Introduction to GUI Awt & Swing Steps for GUI Creation First Java GUI Application JFrame Swing Components 2 different ways (Composition and Inheritance) Layout Managers (FlowLayout, GridLayout, BorderLayout)
15.	Java Database Connectivity	Introduction Database Database Basics SQL Basics Four Kind of JDBC Drivers JDBC-ODBC Bridge Preparing Statement Getting ResultSet Updates in Database

CSC-122
Object Oriented Programming

Course Learning Objectives (C.L.O.)	Program Outcomes											Assessment Tools					
	ab	c	d		e	f	g	h	i	jk		T1	T2	T3	T4	T5	T6
1. Student should know about the object oriented programming																	
2. Become familiar with breaking down a problem into objects rather than procedures																	
3. Inheritance and polymorphism																	
4. Generic programming and interface																	
5. Object oriented software development																	
6. Learn what is available off the shelf to facilitate Java development (tools, libraries)																	
7. Graphic user interface, event handling, and animation																	
8. Java Database Connectivity																	
9. Student should have good command over the syntax of JAVA																	
10. Student should submit a project designed by him/her-self by properly implementing all the concepts learned during semester																	