



National University of Sciences & Technology (NUST)
School of Electrical Engineering and Computer Science (SEECs)
Department of Computer Science

Object-Oriented Programming

Course Code:	CS212	Semester:	2 nd
Credit Hours:	3+1	Prerequisite Codes:	CS110 FoCP
Instructor:	Ms. Hirra Anwar	Class:	BSCS-8AB
Office:	A-216, SEECs Faculty Block	Telephone:	9085 2183
Lecture Days:	Wednesday & Friday	E-mail:	hirra.anwar@seecs.edu.pk
Class Room:		Consulting Hours:	Thursday 3:00-5:00 pm (or by prior email)
Lab Engineer:	Nadeem Nawaz	Lab Engineer Email:	nadeem.nawaz@seecs.edu.pk
	Mehwish Kiran		mehwish.kiran@seecs.edu.pk
Knowledge Group:	KG-PL	Updates on LMS:	Regular updates

Course Description:

The aim of this course is to provide the students about understanding of the principles of the object oriented programming. The students will learn the concepts of object based programming and object oriented programming. In this course, students will also learn about various techniques to develop components for business logic and graphical user interface using the concepts of Inheritance, polymorphisms and interface. In order to design and develop interactive java based applications, the students will also study Model View Controller Architecture under the heading of Graphical User Interface. At the end of the course, student will learn advanced topics like file handling, multithreading, database handling.

In last week, we will include some advance concepts such as Android Programming and Basic Reflection API.

Course Objective:

After completing this course, following objectives will be achieved

- Understanding of objects
- Understanding and usage of the concepts of inheritance and polymorphism objects in real world applications
- Understanding the syntax of Technology (Java).
- Students should be able to design and develop medium size applications using Java.
- Students should be familiar with various basic and advance components of GUI development and will be able to create GUI based applications using problem solving approach.
- Students should be familiar with file handling, JDBC, and java.util packages.
- Students should be capable to design and develop multithreaded java applications.

Course Learning Outcomes (CLOs):

At the end of the course the students will be able to:		PLO Mapping**	BT Level*
1.	Understand the difference between procedural and Object Oriented Programming paradigms.	A	C-2
2.	Demonstrate the ability to create and use OOP constructs to map real world scenarios.	B	C-5
3.	Develop programs using object oriented techniques.	C	C-3
4.	Use the latest IDEs to enable quick development, testing, documentation, and packaging of programs.	I	C-3

* BT= Bloom's Taxonomy, C=Cognitive domain, P=Psychomotor domain, A= Affective domain

- Knowledge(C-1), Comprehension(C-2), Application(C-3), Analysis(C-4), Synthesis(C-5), Evaluation(C-6)
- Perception(P-1), Set(P-2), Guided Response(P-3), Mechanism(P-4), Complete Overt Response(P-5), Adaption(P-6), Organization(P-7)
- Receiving(A-1), Responding(A-2), Valuing(A-3), Organization(A-4), Internalizing(A-5)

** Description of Program Learning Outcomes (PLOs) is available on website and in a separate document.



National University of Sciences & Technology (NUST)
School of Electrical Engineering and Computer Science (SEECs)
Department of Computer Science

Topics to be Covered:

1. Introduction	7. Polymorphism
2. Basic I/O & Flow Control	8. Event driven Programming
3. Classes and Objects	9. File Handling
4. Arrays and Strings	10. Exception Handling
5. Object based Programming	11. Multithreading
6. Inheritance	12. Advanced Topics

Lecture Breakdown:

Week No.	Topics	Assessment	Remarks
1	Java Basics/Introduction <ul style="list-style-type: none"> • Introduction to Java Programming Language • Compiling & Executing Simple Java Programs • Java execution cycle 		
2	Flow Control & Data Types <ul style="list-style-type: none"> • Decisions • Loops and Iterations • Switch statement • Data types & Operators • Operator Precedence 		
3	Classes and Objects <ul style="list-style-type: none"> • Basic Concepts • Encapsulation & Data Hiding • Access modifiers • Constructors • Setters & Getters 		
4	Classes and Objects <ul style="list-style-type: none"> • Copy Constructor • Default and No-Argument Constructors • Method (Function) Overloading • Static Class Members • this Reference 		
5	Arrays and Strings/ Functions <ul style="list-style-type: none"> • Arrays- 1D & 2D arrays • String Class and Functions • String Manipulation 		
6	OHT-1		
7	Inheritance <ul style="list-style-type: none"> • Super classes and sub classes • Protected members • Method (Function) Overriding • Constructors in subclasses • Case Study 		



National University of Sciences & Technology (NUST)
School of Electrical Engineering and Computer Science (SEECs)
Department of Computer Science

8	Polymorphism <ul style="list-style-type: none"> Abstract Classes & Methods Polymorphic Behavior Final Methods and Classes Case Study
9	Interfaces and Abstract Classes <ul style="list-style-type: none"> Creating and Using Interfaces Case Study
10	Event Driven Programming-Part I <ul style="list-style-type: none"> Events Swing Library
11	Event Driven Programming-Part II <ul style="list-style-type: none"> Event Listeners Adapters
12	OHT-2
13	Exception Handling & File Handling <ul style="list-style-type: none"> Try/Catch/Finally Re-throwing Exceptions File Handling
14	Multithreading <ul style="list-style-type: none"> Java multithreading concepts Designing Multithreaded applications
15	Database Handling <ul style="list-style-type: none"> Sample applications & DB connectivity JDBC
16	Advanced Topics
17	Advanced Topics
18	ESE

Lab Experiments:

Lab 01	Java Basics [Data types and Operators]
Lab 02	Flow Control [Loops and Decisions]
Lab 03	Classes and Objects
Lab 04	Constructors & Method Overloading, Static Members
Lab 05	Arrays & Strings
Lab 06	Inheritance
Lab 07	Polymorphism
Lab 08	Interfaces and Abstract Classes
Lab 09	Event Driven Programming
Lab 10	GUI
Lab 11	Exception Handling & File Handling



National University of Sciences & Technology (NUST)
School of Electrical Engineering and Computer Science (SEECs)
Department of Computer Science

Lab 12	Multithreading
Lab 13	Advanced Topic: JDBC
Lab 14	Advanced Topic: Android Programming

Tools / Software Requirement:

Java Development Kit (JDK) 7 or later, Eclipse/Netbeans

Books:

Text Book: Harvey M. Deitel, Paul J. Deitel, Java How To Program, Early Objects 11th Edition, by Deitel & Deitel, ISBN-13: 9780134743356, published by Prentice Hall, 2018

Reference 1. Ivor Horton's Beginning Java, Java 7 Edition, Publisher: Wrox Press; Java 7 Edition edition (2011)

Books: 2. Bruce Eckel, Thinking in Java, 4th Edition, ISBN-13: 978-0131872486, ISBN-10: 0131872486

Course Assessment

Exam:	2 One Hour Tests (OHT) and 1 End Semester Exam (ESE)
Home work:	3 Assignments minimum
Lab Assignments:	14 Lab Reports , 1 Lab Exam (optional)
Semester Project:	1 Report for the term/semester project
Quizzes:	4 - 6 Quizzes

Course Assessment Weightages (In accordance with NUST statutes)

Theory: 75%
• Quizzes: 10%
• Assignments: 10%
• OHT-1: 15%
• OHT-2: 15%
• End Semester Exam: 50%
Practical: 25%
• Lab Assignments: 70% (17.5 of 100)
• Project : 30% (7.5 of 100)

**The grading criteria may change as we proceed in the semester*

Grading Policy:

Quiz Policy: The quizzes will be unannounced and normally last for ten minutes. The question framed is to test the concepts involved in last few lectures. Number of quizzes that will be used for evaluation is at the instructor's discretion.



National University of Sciences & Technology (NUST)
School of Electrical Engineering and Computer Science (SEECs)
Department of Computer Science

Assignment Policy:	In order to develop comprehensive understanding of the subject, assignments will be given. Late assignments will not be accepted / graded. All assignments will count towards the total (No 'best-of' policy). The students are advised to do the assignment themselves. Copying of assignments is highly discouraged and violations will be dealt with severely by referring any occurrences to the disciplinary committee. The questions in the assignment are meant to be challenging to give students confidence and extensive knowledge about the subject matter and enable them to prepare for the exams.
Lab Conduct:	The labs will be conducted for three hours every week. A lab handout will be given in advance for study and analysis. The lab handouts will also be placed on LMS. The students are to submit their results by giving a lab report at the end of lab for evaluation. One lab report per group will be required. However, students will also be evaluated by oral viva during the lab.
Plagiarism:	SEECs maintains a zero tolerance policy towards plagiarism. While collaboration in this course is highly encouraged, you must ensure that you do not claim other people's work/ ideas as your own. Plagiarism occurs when the words, ideas, assertions, theories, figures, images, programming codes of others are presented as your own work. You must cite and acknowledge all sources of information in your assignments. Failing to comply with the SEECs plagiarism policy will lead to strict penalties including zero marks in assignments and referral to the academic coordination office for disciplinary action.

BSCS Program Learning Outcomes

Program Learning Outcomes (PLOs) describe what students are expected to know and are able to do by the time of graduation in light of the knowledge, skills and attitude they acquire while progressing through the program. The BSCS graduates of NUST-SEECs will demonstrate the following attributes for the organization they join;

- (A) **Computing Knowledge:** An ability to apply knowledge of computing and mathematics appropriate to the discipline.
- (B) **Problem Analysis:** An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution.
- (C) **Design/Development of Solutions:** An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs.
- (D) **Individual & Team Work:** An ability to function effectively on teams to accomplish a common goal.
- (E) **Ethics:** An understanding of professional, ethical, legal, security and social issues and responsibilities.
- (F) **Communication:** An ability to communicate effectively with a range of audiences.
- (G) **Societal Impact:** An ability to analyze the local and global impact of computing on individuals, organizations, and society.
- (H) **Lifelong Learning:** Recognition of the need for and an ability to engage in continuing professional development.
- (I) **Modern Tool Usage:** An ability to use current techniques, skills, and tools necessary for computing practice.
- (J) **Investigation:** An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.
- (K) **Project Management:** An ability to apply design and development principles in the construction of software systems of varying complexity.