

Faculty of Computers & Information Technology

Advanced Programming

Information :

Course Code : CS224

Level : Undergraduate

Course Hours : 3.00- Hours

Department : Faculty of Computers & Information Technology

Area Of Study :

Explain the different object-oriented programming (Java) concepts.
 Design software programs using object oriented patterns (Java).
 Implement solutions to problems using an object-oriented programming language (Java). Implement a full GUI Program
 Use socket programming for designing multi-user programs
 Implement a programs using file I/O Streams

Description :

This course is designed to give the student a deep view to object oriented programming. It also covers many topics related to creating rich desktop applications. Students get involved in practical applications regarding the implementation of GUI Components, File Streaming, Threading and Basic Socket Programming.

Course outcomes :

a.Knowledge and Understanding: :

1 -	Outline the OOP (Java) concepts.
2 -	Explain the OOP (Java) concepts (Encapsulation, Inheritance, Polymorphism, Abstraction and Interfaces).
3 -	State the different GUI design methods and tools for OOP (Java).
4 -	Identify the different I/O stream methods for OOP (Java).

b.Intellectual Skills: :

1 -	Analyze a given problem to design a system using OOP (Java) concepts.
2 -	Select appropriate OOP (Java) methodologies and GUI components for a given problem solution and setting out their limitations, restrictions and errors.
3 -	Differentiate between algorithms, methods and techniques used in I/O streams for OOP (Java).

c.Professional and Practical Skills: :

1 -	Design, implement and test for OOP (Java) techniques to solve various problems.
2 -	Apply OOP (Java)concepts for implementing a computer based system.
3 -	Create a complete user interface programs for OOP (Java).
4 -	Use I/O streams libraries and socket for OOP (Java).

d.General and Transferable Skills: :

1 -	Exploit a range of learning resources.
-----	--

2 - Work on a team

Course Topic And Contents :

Topic	No. of hours	Lecture	Tutorial / Practical
Classes and Objects	4	2	2
OOP : Encapsulation and Inheritance	4	2	2
OOP: Polymorphism (static and dynamic binding)	4	2	2
OOP: Abstraction and Interface.	4	2	2
Graphical User Interface (GUI): • GUI Class Hierarchy • Frames and panels	4	2	2
Graphical User Interface (GUI): • GUI Components	4	2	2
Graphical User Interface (GUI): • Drawing , Using Colors, Fonts, and Font Metrics, Geometric Figures	4	2	2
File Streams	4	2	2
Mid Term Exam	2		
File Streams	4	2	2
Socket Programming	4	2	2
Socket Programming	4	2	2
Course Project	4	2	2
Final Exam	2		

Teaching And Learning Methodologies :

Interactive Lectures including discussion

Practical Lab Sessions

Self-Study (Project / Reading Materials / Online Material / Presentations)

Problem Solving

Course Assessment :

Methods of assessment	Relative weight %	Week No	Assess What
Assignments	10.00	3	
Final Exam	40.00	14	
Midterm Exam (s)	20.00	9	
Quizzes	10.00	5	
Team Work Projects	20.00	7	

Course Notes :

Course Notes are available with all the slides used in lectures in electronic form on Learning Management System (Moodle)

Recommended books :

Poo, Danny, Kiong, Derek, Ashok, Swarnalatha , "Object-Oriented Programming and Java", (last version), 978-1-84628-963-7

Web Sites :

<https://docs.oracle.com/javase/tutorial/java/concepts/>