Anekant Education Society’s

Tuljaram Chaturchand College, Baramati

(Autonomous)

M.Sc. Degree Program in Computer Science

(Faculty of Science & Technology)

Syllabus

M .Sc.(Computer Science) Semester -I

For Department of Computer Science

Tuljaram Chaturchand College, Baramati

Choice Based Credit System Syllabus (2022 Pattern)

To be implemented from Academic Year 2022-2023

Programme Specific Outcomes (PSOs)

For M.Sc. (Computer Science)

After completing M.Sc. Computer Science Program students will be able to:

PSO1: Enrich the knowledge in the areas like Artificial Intelligence, Web Services, Cloud Computing, Paradigm of Programming language, Design and Analysis of Algorithms, Database Technologies Advanced Operating System, Mobile Technologies, Software Project Management and core computing subjects. Choose to study any one subject among recent trends in IT provided in the optional subjects.

PSO2: Students understand all dimensions of the concepts of software application and projects.

PSO3: Students understand the computer subjects with demonstration of all programming and theoretical concepts with the use of ICT.

PSO4: Developed in-house applications in terms of projects.

PSO5: Interact with IT experts & knowledge by IT visits.

PSO6: Get industrial exposure through the 6 months Industrial Internship in IT industry.

PSO7: To make them employable according to current demand of IT Industry and responsible citizen.

M.Sc. (Computer Science) I (Sem I)

2019Pattern		2022Pattern	
Subject	Paper Code	Paper Code	Paper Title
Principles of Programming Languages	COMP4101	PSCS111	Principles of Programming Language (C)
Cryptography & Network Security	COMP4102	PSCS112	Cryptography and Cyber Forensics(C)
Database Technologies	COMP4103	PSCS113	Database Technologies (C)
Design and Analysis of Algorithms	COMP4104	PSCS114	Design and Analysis of Algorithms(C)
Programming with DOTNET	COMP4105	PSCS115	Dot Net Framework& C# (C)
Lab Course on DOT NET, PPL and Database Technologies	COMP4106	PSCS116	Lab Course on Dot Net, PPL,DBT&DAA(C)
Human Rights–I	HR-101	HR1	Human Rights – I
Introduction to Cyber Security–I	CYS-101	CYS1	Introduction to Cyber Security–I

M.Sc. (Computer Science) I (Sem II)

2019Pattern		2022 Pattern	
Paper Title	Paper Code	PaperCode	PaperTitle
Digital Image Processing	COMP4201	PSCS121	Digital Image Processing
Data Mining and Data Warehousing	COMP4202	PSCS122	Data Mining and Data Warehousing
Python Programming	COMP4203	PSCS123	Emerging Technologies: Python Programming
Advanced Operating System	COMP4204	PSCS124	Dot Net (Advanced): ASP.NET Core using MVC.
Lab Course on Python Programming & Advance Operating System	COMP4205	PSCS125	Lab course on Dot Net and Python
Project	COMP4206	PSCS126	Project

Artificial Intelligence	COMP4206	PSCS127(A)
		Or
		PSCS127(B)
			Artificial Intelligence
			Or
			Advanced Operating System

Introduction to Cyber Security–I	CYS-101	CYS-102	Introduction to Cyber Security–II

M.Sc.(Computer Science)

Semester-I

Class: M.Sc. (Computer Science) (Semester-I) PaperCode: PSCS111
Title of paper: Principles of Programming Language Paper: I
Credit -4 No.ofLectures-60

Prerequisites:

- It is assumed that student learning this course have the following background:
- Experience with an OOP language(such as Java or C++)
- Experience with procedural language (such as C)
- Working knowledge of C, C++, and Java programming.
- Basic algorithms anti structure concepts.

Objectives:

- CO1.Knowledge of, and ability to use, language features used in current programming languages
- CO2.An ability to program in different language paradigms and evaluate their relative benefits.
- CO3.An understanding of the key concepts in the implementation of common features of programming language.
- CO4.To apply suitable programming paradigm for the application.
- CO5.To program in different language paradigms and evaluate their relative benefits
- CO6.The programming paradigms of modern programming languages.
- CO7.Knowledge to compare the features of various programming languages

This course focuses on both:

Theory is covered by the textbook readings, lecture san orthotics Implementation is covered by the homework assignments.

Unit	**Title and Contents**	**No. of lectures**
Unit-I	**Programming Domains** The Art of Language Design-The Programming Language Spectrum, Why Study Programming Languages? Types of Programming Language Domains # Scientific Applications – Large Number of Floating Point Computations – FORTRAN #Business Applications –Produce Reports, Used ecimal numbers and characters– COBOL # Artificial Intelligence – Symbols rather than numbers manipulated –LISP #Systems Programming –Need Efficiency because of continuous use–C #WebSoftware–EclecticCollectionofLanguages:Markup(e.g.,XHTML),Scripting(e.g., PHP),General-Purpose(e.g., Java) #Data Analytics Applications –R Programming, Python Programming	**8**

Unit-II	**Names, Scopes and Bindings** Meaning of Names in Scope-Aliases, Object Lifetime and Storage Management: Static Allocation, Stack-based Allocation, Heap-Based Allocation, Garbage Collection The Binding of Referencing Environments - Subroutine Closures, Object Closures, Nested Subroutines, Declaration Order Scope Rules, Static Scoping, Dynamic Scoping Overloading, Polymorphism and elated concepts, Macro Expansion	**8**
Unit-III	**Data Types** Primitive Data Types - Numeric Types, Integer, Floating point, Complex, Decimal, Boolean Types, Character Types, Character String Types-Design Issues, Strings and Their Operations, String Length Operations, Implementation of Character String Types. User defined Ordinal types - Enumeration types, Designs, Evaluation, Sub range types, Evaluation, Implementation of User defined ordinal types Array types –Array initialization, Array operations, Rectangular and Jagged arrays, Slices, Evaluation, Implementation of Array Types AssociativeArrays–Structureandoperations,ImplementingAssociativearrays Record Type – Definitions of records, References to record fields, Operations on records, Evaluation, Implementation of Record types Union Type– Design issues, Discriminated versus Free unions, Evaluation, Implementation of Union types Pointer and Reference Types - Design issues, Pointer operations, Pointer problems – Dangling pointers, Lost heap dynamic variables,PointersinCandC++,Referencetypes,Evaluation,Implementationof Pointer and reference types, Representation of pointer sander francs, Solution to dangling pointer problem, Heap management	**8**
Unit-IV	**Control Flow** Expression Evaluation-Precedence and Associativity, Assignments,Initialization,OrderingWithinExpressions,Short-CircuitEvaluation Structured and Unstructured Flow–Structured Alternatives to GOTO Sequencing Selection–Short Circuited Conditions, Case/Switch Statements Iteration–EnumerationControlledLoops,Combination,Loops,Iterators,Logically Controlled Loops Recursion-IterationandRecursion,ApplicativeandNormalOrderEvaluation	**8**

<table>
<tr><td><b>Unit-VI</b></td><td><b>Data Abstraction and Object Orientation</b>
Encapsulation and Inheritance-Modules, Classes, Nesting, Type, Extensions, Extending without Inheritance
InitializationandFinalization-ChoosingaConstructor,Referencesand Values, Execution Order
Dynamic Method Binding-Virtual and Non-Virtual Methods, Abstract Classes and Interfaces, Member Lookup, Polymorphism, Object Closures
Multiple Inheritance-Semantic Ambiguities, Shared Inheritance, Replicated Inheritance, Mix-In Inheritance</td><td><b>8</b></td></tr>
<tr><td colspan="3"><b>References:</b>
Scott, Programming Language Pragmatics, 3e(With CD) ISBN 9788131222560 Kaufmann Publishers, An Imprint of Elsevier, USA
RobertW.Sebesta,ConceptsofProgrammingLanguages,EighthEdition,PearsonEducation
Carl Towns end, Introduction to Turbo Prolog
Patrick Henry Winston& Berthold Klaus Paul Horn,LISP3rdedition–BPB
M.Gabbrielli,S.Martini,,ProgrammingLanguages:PrinciplesandParadigms,SpringerISBN:9781848829138</td></tr>
</table>

NOTE:48 LECTUREFORCURRICULUM(TEACHING)&12LECTURES FORLEARNING

Mapping of this course with Programme Outcomes

Course Outcomes	**Programme Outcomes (POs)**						
	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	2	3	3	1	2	3
CO2	3	2	3	3	1	2	3
CO3	3	3	3	3	1	3	3
CO4	3	2	3	3	1	2	3
CO5	3	2	3	3	1	2	3
CO6	3	3	3	3	1	3	3
CO7	3	2	3	3	1	3	3

Weight: 1 - Partially related 2 - Moderately Related 3 - Strongly related

Justification

PO1 with All CO's : CO1, CO2, CO3, CO4, CO5, CO6, and CO7 are strongly related to PO1. They align directly with the objective of enriching knowledge in various IT areas, including programming languages, paradigms, syntax, semantics, and their practical applications.

PO2 with All CO's: CO1, CO2, CO4, and CO5 are moderately related to PO2. While they contribute to the understanding of programming languages and their features, they may not directly cover all dimensions of software applications and projects.

CO3 and CO7 are strongly related to PO2. These outcomes directly address the ability to apply programming paradigms for applications and evaluate their benefits in different language paradigms, which align with understanding the concepts of software applications and projects.

CO6 is moderately related to PO2. Understanding the concepts of ADT and OOP contributes to the broader understanding of software concepts, but it may not cover all dimensions on its own.

PO3 with All CO's All COs (CO1 through CO7) are strongly related to PO3. Each of these course outcomes involves understanding, applying, and demonstrating various programming and theoretical concepts, which align with the broader objective of understanding computer subjects with the use of ICT.

This mapping suggests a strong alignment between the Course Outcomes and the Program Outcome, indicating that the successful achievement of each Course Outcome contributes significantly to the overall understanding of computer subjects with the demonstration of programming and theoretical concepts using ICT.

PO4 with All CO's All COs (CO1 through CO7) are strongly related to PO4. Each of these course outcomes contributes to the development of in-house applications in terms of projects. The ability to evaluate, express, apply programming paradigms, and understand various programming concepts are essential skills for developing applications.

This mapping suggests a strong alignment between the Course Outcomes and the Program Outcome, indicating that the successful achievement of each Course Outcome contributes significantly to the overall ability to develop in-house applications as projects.

PO5 with All CO's All COs (CO1 through CO7) are partially related to PO5. While the course outcomes focus on enhancing programming language skills and understanding various programming concepts, they may not directly contribute to the interaction with IT experts or knowledge gained through IT visits.

This mapping indicates that the Course Outcomes are only partially related to the Program Outcome of interacting with IT experts and gaining knowledge through IT visits. Additional components related to networking, communication, and professional interaction may be necessary to fully address PO5.

PO6 with All CO’s: CO3, CO6, and CO7 are strongly related to PO6. These outcomes directly contribute to the ability to apply programming paradigms, understand concepts of ADT and OOP, and program in different language paradigms, which are essential skills during an industrial internship in the IT industry.

CO1, CO2, CO4, and CO5 are moderately related to PO6. While these outcomes involve evaluating and expressing syntax, semantics, and comparing features of programming languages, they may not be as directly tied to the practical experience gained during an industrial internship.

This mapping reflects the varying degrees of relevance between each Course Outcome and the Program Outcome, emphasizing the importance of certain outcomes in preparing students for industrial exposure.

PO7 with All CO’s All COs (CO1 through CO7) are strongly related to PO7. The course outcomes, which involve evaluating and enhancing programming language skills, understanding programming paradigms, and applying concepts like ADT and OOP, contribute directly to making students employable in the current IT industry demands.

This mapping suggests a strong alignment between the Course Outcomes and the Program Outcome, indicating that the successful achievement of each Course Outcome significantly contributes to the overall goal of making students employable and responsible citizens.

Class: M.Sc. (Computer Science) (Semester-I) **Paper Code: PSCS112**
Title of paper: Cryptography and Network Security **Paper: II**
Credit -4 **No. of Lectures: 50**

Learning Objectives:

- To enable students to get sound understanding of Info-Sys-Security, Network Security, Cryptography.
- To equip with knowledge and skills necessary to support for their career in Network Security.
- To develop attitude and interest along with necessary knowledge and skills among the students to encourage them to do further academic studies / research in this area, after the completion of their PG Course.

Course Outcomes:

1. **CO1: Understanding of Cryptographic Fundamentals:** Define and explain basic cryptographic terms and concepts and understand the principles of encryption and decryption.
2. **CO2: Knowledge of Cryptographic Algorithms:** Identify and compare various cryptographic algorithms, including symmetric and asymmetric algorithms, also analysis of the strengths and weaknesses of different encryption techniques.
3. **CO3: Network Security Protocols:** Describe common network security protocols (e.g., SSL/TLS, IPsec) and understand the role of protocols in securing communication.
4. **CO4: Firewall and Intrusion Detection Systems:** Understand the role of firewalls and intrusion detection systems in network security and configure and manage firewall rules.
5. **CO5: Incident Response and Forensics:** Develop skills in detecting and responding to security Incidents and understand the basics of digital forensics.
6. **CO6: Network Security Protocols, Policies and Compliance:** Understand the role of protocols in securing communication and understand and adhere to legal and regulatory compliance requirements.
7. **CO7: Cryptographic Applications:** Apply cryptographic techniques in real-world scenarios and analyze the security of existing cryptographic implementations.

Units	**Title & Contents**	**No. of Lectures**
Unit – I	**Introduction to Security:** The Need for Security, Security Approaches, Principles of Security, Types of Attacks	02
Unit – II	**Cryptography and techniques:** Introduction, Plain Text and Cipher Text, Substitution Techniques, Transposition Techniques, Encryption and Decryption, Symmetric and Asymmetric key cryptography, steganography.	06
Unit – III	**Symmetric Key Algorithms and AES:** Algorithm Types and Modes, Overview of Symmetric Key Cryptography, DES, IDEA, Blowfish	06
Unit – IV	**Asymmetric Key Algorithms, Digital Signature and RSA:** Brief History of Asymmetric Key Cryptography, overview, RSA Algorithm, Comparison between Symmetric & Asymmetric Key Algorithms, Digital Signature	06
Unit – V	**Digital Certificates and Public Key Infrastructure (PKI):**	03

	Introduction, Digital Certificates, private key management,	
Unit – VI	**Internet Security Protocols:** Basic Concepts, SSL, TLS, SHTTP, TSP, SET, SSL v/s SET, 3-D Secure Protocol, Electronic Money, Email Security, WAP Security, Users Authentication and Kerberos: Authentication Basics, Password, Authentication Tokens, Certificate based Authentication, Biometric Authentication, Kerberos, Security Handshake pitfalls, SSO Approaches.	14
Unit – VII	**Network Security**, Firewalls and Virtual Private Network (VPN): Brief introduction to TCP/IP, Firewalls, IP Security, VPN, Intrusion	05
Unit – VIII	**Case studies on Cryptography and Security:** Introduction, Cryptographic solutions, Secure inter branch payment transaction, Denial of services (DOS) attacks, IP Spooling Attack, Cross site scripting vulnerability (CSSV), Contract Signing, Secret Splitting, Virtual Electronics, Cookies and Privacy.	08

Reference:

- Cryptography and Network Security: Atul Kahate
- Computer Network Security: Kizza, Springer
- Network Security, Harrington, Elsevier

Mapping of this course with Programme Outcomes

Course Outcomes	Programme Outcomes (POs)						
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**
CO1	3	2	3	3	2	2	3
CO2	3	2	3	3	2	2	3
CO3	2	2	3	3	2	2	3
CO4	2	2	3	3	2	2	3
CO5	2	2	3	3	2	2	3
CO6	3	2	3	3	2	2	3
CO7	3	2	3	3	2	2	3

Weight: 1 - Partially related 2 - Moderately Related 3 - Strongly related

Justification of above mapping:

PO1 – CO1: Cryptographic fundamentals are foundational to the program's knowledge, and a strong understanding is crucial.

PO1 – CO2: Knowledge of cryptographic algorithms is a key aspect of program knowledge, contributing significantly.

PO1 – CO3: Understanding network security protocols is essential in demonstrating broad knowledge of the program.

PO1 – CO4: Knowledge of firewalls and intrusion detection systems is vital for a comprehensive understanding of the program.

PO1 – CO5: Incident response and forensics knowledge enhances the practical understanding of the program's knowledge.

PO1 – CO6: Understanding protocols, policies, and compliance contributes to the holistic knowledge of network security.

PO1 – CO7: Applying cryptographic techniques in real-world scenarios demonstrates a deep understanding of program knowledge.

PO2 – CO1, CO2, CO3, CO4, CO5, CO6, CO7 (communication aspects): Communication and collaboration skills are essential for social competence, making these outcomes significant.

PO3 – CO1, CO2, CO3, CO4, CO5, CO6, CO7(analytical skills) : Critical thinking is exercised in problem-solving, making these course outcomes crucial for PO3.

PO4 - CO1, CO2, CO3, CO4, CO5, CO6, CO7 (individual and team work): Personal and professional competence is directly linked to individual and collaborative performance, emphasizing these outcomes.

PO5 - CO1, CO2, CO3, CO4, CO5, CO6, CO7 (research and ethics): Emphasizing academics, research ethics, and awareness of intellectual property rights align with scientific temper.

PO6 - CO1, CO2, CO3, CO4, CO5, CO6, CO7 (technology-oriented): Continuous learning in the context of technological changes is crucial, making these outcomes directly align with PO6.

PO7 - CO1, CO2, CO3, CO4, CO5, CO6, CO7 (social concern, ethics): Demonstrating empathetic social concern, fair national development, and ethical commitment align with PO7.

Class : M. Sc.(Computer Science)(Semester-I) **Paper Code: PSCS113**
Title of paper: Database Technologies **Paper: III**
Credit :4 **No. of Lectures: 60**

Prerequisites: Knowledge of RDBMS

Course objectives:

TostudytheusageandapplicationsofParallelandDistributeddatabases, Object relational database and emerging systems.

To acquire knowledge on NoSQL databases.

Course Outcomes:

CO1.Comparedifferentdatabasetechnologies.

CO2.CompareandcontrastNoSQLdatabaseswithRDBMS

CO3. Identify the data models for relevant problems.

CO4. Demonstrate the basic elements of an object relational database.

CO5. Compare different mobile databases.

CO6. Compare different NoSQL products.

CO7. Master CRUD and other operations on MongoDB.

Units	**Title & Contents**	**No. ofLectures**
Unit–I	**Parallel and Distributed Databases** Database System Architectures: Centralized and Client-Server Architectures, Server System Architectures, Parallel Systems, Distributed Systems, ParallelDatabases:I/OParallelism,InterandIntraQueryParallelism,InterandIntraoperationParallelism **Distributed Database Concepts:** DistributedDataStorage,DistributedTransactions,CommitProtocols,ConcurrencyControl,DistributedQueryProcessing,ThreeTierClientServer Architecture	08
Unit–II	**Object and Object Relational Databases** Concepts for Object Databases: Object Identity, Object Structure, Type Constructors, EncapsulationofOperators,Methods,Persistence,TypeandClassHierarchies,Inheritance, ComplexObjects,ObjectDatabaseStandards, Languages and Design: ODMG Model, ODL, OQL ObjectRelationalandExtendedRelationalSystems:ObjectRelationalfeaturesinSQL/Oracle, Case Studies	12

Unit–III	**Mobile Databases** Location and Handoff Management, Effecton Mobility on Data Management, Location Dependent Data Distribution,Mobile Transaction Models, Concurrency Control, Transaction Commit Protocols, Mobile Database Recovery Schemes, Examples: Oracle Database Lite, Microsoft SQL Server Compact	08
Unit–IV	**Introduction to NoSQL** Concepts and evolution, History of NoSQL, Different No SQL products: MongoDB, CouchDB, Cassandra, Exploring MongoDB, Advantages of MongoDB over RDBMS, Interfacing and Interacting with NoSQL, Sharding, Replication	08
Unit–V	**Working with NoSQL** NoSQL Storage Architecture, CRUD operations with MongoDB, Querying, Modifying and Managing NoSQL datastores,Indexing and ordering datasets, Surveying database internals Migrating from RDBMS to NoSQL, Implementing NoSQL with PHP	08
Unit–VI	**MongoDB Aggregation and data management** Introduction to aggregation,Types of Aggregation, Performance Tuning, Export and Import of data to and from MongoDB, Capped collections/Expired data from TTL hands on examples.	06
	References: 1. Henry Korth, Abraham Silberschatz and S.Sudarshan, “Database System Concepts”, Sixth Edition, McGraw Hill, 2011. 2. M. Tamer Ozsu and Patrick Valduriez, “Principles of Distributed Database Systems”, Third Edition, Springer, 2011. 3. Thomas Connolly and Carolyn Begg, “Database Systems, A Practical Approach to Design, Implementation and Management”, Fourth Edition, Pearson Education, 2008, Fifth Edition, Pearson Education, 2010, Sixth Edition, Pearson Education, 2015. 4. R. Elmasri, S. B. Navathe, “Fundamentals of Database Systems”, Seventh Edition, Pearson Education, 2017. 5. C. J. Date, A. Kannan, S. Swamynathan,“An Introduction to Database Systems”, Eighth Edition, Pearson Education, 2006. 6. Dan Sullivan,“NoSQL for Mere Mortals”, First Edition, Pearson Education,2015. 7. Kristina Chodorow, “MongoDB The Definitive Guide”, Second Edition, O’Reilly,2013. 8. https://www.mongodb.com/docs/manual/	

Mapping of this course with Program Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	3	3	3	3	3	3
CO2	3	3	3	3	3	3	3
CO3	3	3	3	3	3	3	3
CO4	3	3	3	3	3	3	3
CO5	3	3	3	3	3	3	3
CO6	3	3	3	3	3	3	3
CO7	3	3	3	3	3	3	3

Weight: 1 - Partially related 2 - Moderately Related 3 - Strongly related

Justification of Mapping of PO1 to all COs

PO1: Enrich the knowledge in the areas like Artificial Intelligence, Web Services, Cloud Computing, Paradigm of Programming language, Design and Analysis of Algorithms, Database Technologies Advanced Operating System, Mobile Technologies, Software Project Management, and core computing subjects.

CO1. Compare different database technologies.

Justification: Enriching knowledge in Database Technologies, as emphasized in PO1, involves understanding and comparing different database technologies, making CO1 strongly related to the program outcome.

CO2. Compare and contrast NoSQL databases with RDBMS.

Justification: The comparison between NoSQL databases and RDBMS is integral to the knowledge enrichment in Database Technologies, a key area mentioned in PO1.

CO3. Identify the data models for relevant problems.

Justification: Identifying data models for specific problems is a fundamental aspect of Database Technologies, contributing to the enriched knowledge outlined in PO1.

CO4. Demonstrate the basic elements of an object-relational database.

Justification: Demonstrating the basic elements of an object-relational database is directly aligned with the goal of enriching knowledge in Database Technologies, a core component of PO1.

CO5. Compare different mobile databases.

Justification: Comparing different mobile databases is part of understanding Mobile Technologies, which is explicitly mentioned in the enriched knowledge areas outlined in PO1.

CO6. Compare different NoSQL products.

Justification: Comparing different NoSQL products is directly related to Database Technologies, one of the areas specified in PO1.

CO7. Master CRUD and other operations on MongoDB.

Justification: Mastering CRUD operations on MongoDB aligns with the knowledge enrichment in Database Technologies, as mentioned in PO1, contributing directly to practical skills in the field.

Justification of Mapping of PO2 to all COs

PO2: Students understand all dimensions of the concepts of software application and projects.

CO1. Compare different database technologies.

Justification: Understanding different database technologies is fundamental to comprehending the dimensions of software applications, aligning strongly with the goal of PO2.

CO2. Compare and contrast NoSQL databases with RDBMS.

Mapping: 3 - Strongly related

Justification: Comparing and contrasting NoSQL databases with RDBMS contributes to a deeper understanding of database technologies, an essential dimension in software application concepts as per PO2.

CO3. Identify the data models for relevant problems.

Justification: Identifying data models for specific problems is directly connected to the conceptual understanding of software applications, a key dimension of PO2.

CO4. Demonstrate the basic elements of an object-relational database.

Justification: Demonstrating the basic elements of an object-relational database is integral to understanding the structure of software applications, reinforcing the goals of PO2.

CO5. Compare different mobile databases.

Justification: Comparing different mobile databases is directly linked to the mobile technology aspect, contributing to a comprehensive understanding of software applications as outlined in PO2.

CO6. Compare different NoSQL products.

Justification: Comparing different NoSQL products enhances students' awareness of diverse database solutions, contributing to a well-rounded understanding of software application concepts, in line with PO2.

CO7. Master CRUD and other operations on MongoDB.

Justification: Mastering CRUD operations on MongoDB involves practical skills relevant to software applications, directly supporting the comprehensive understanding sought in PO2.

Justification of Mapping of PO3 to all COs

PO3: Students understand the computer subjects with a demonstration of all programming and theoretical concepts with the use of ICT.

CO1. Compare different database technologies.

Justification: Demonstrating a comparison of different database technologies involves practical application, aligning directly with the goal of understanding computer subjects through the use of ICT, as specified in PO3.

CO2. Compare and contrast NoSQL databases with RDBMS.

Justification: Demonstrating a comparison between NoSQL databases and RDBMS with the use of ICT supports the practical application of theoretical concepts, enhancing students' understanding of computer subjects as outlined in PO3.

CO3. Identify the data models for relevant problems.

Justification: Identifying data models for specific problems involves applying theoretical concepts in a practical context using ICT, directly supporting the goal of understanding computer subjects in PO3.

CO4. Demonstrate the basic elements of an object-relational database.

Justification: Demonstrating the basic elements of an object-relational database requires practical application of theoretical concepts with the use of ICT, contributing to the understanding of computer subjects in line with PO3.

CO5. Compare different mobile databases.

Justification: Comparing different mobile databases involves practical application using ICT, aligning with the goal of demonstrating programming and theoretical concepts in the context of computer subjects as stated in PO3.

CO6. Compare different NoSQL products.

Justification: Comparing different NoSQL products requires a demonstration of theoretical knowledge and practical skills with the use of ICT, contributing directly to the understanding of computer subjects in PO3.

CO7. Master CRUD and other operations on MongoDB.

Justification: Mastering CRUD operations on MongoDB involves practical mastery of programming concepts and theoretical knowledge, directly supporting the goal of understanding computer subjects with the use of ICT as per PO3.

Justification of Mapping of PO4 to all COs

PO4: Developed in-house applications in terms of projects.

CO1. Compare different database technologies.

Justification: Developing in-house applications requires choosing suitable database technologies, making the comparison of different technologies crucial, aligning with the goal of PO4.

CO2. Compare and contrast NoSQL databases with RDBMS.

Justification: The choice between NoSQL databases and RDBMS is critical in developing in-house applications, making the comparison and contrast directly relevant to the goals of PO4.

CO3. Identify the data models for relevant problems.
Justification: Identifying data models is an integral part of the development process for in-house applications, aligning with the practical aspects of CO3 and the overall goal of PO4.
CO4. Demonstrate the basic elements of an object-relational database.
Justification: Demonstrating the basic elements of an object-relational database is directly applicable to the development of in-house applications, supporting the objectives of CO4 and PO4.
CO5. Compare different mobile databases.
Justification: Developing in-house applications for mobile platforms involves choosing appropriate mobile databases, making the comparison of different mobile databases relevant to the goals of PO4.
CO6. Compare different NoSQL products.
Justification: Choosing the right NoSQL product is crucial in the development of in-house applications, emphasizing the relevance of comparing different NoSQL products to the objectives of PO4.
CO7. Master CRUD and other operations on MongoDB.
Justification: Mastering CRUD operations on MongoDB is directly applicable to the development of in-house applications, aligning with the practical skills needed for successful project implementation in the context of PO4.

Justification of Mapping of PO5 to all COs

PO5: Interact with IT experts & knowledge by IT visits.
CO1. Compare different database technologies.
Justification: Interacting with IT experts during IT visits provides an opportunity to discuss and understand various database technologies, aligning directly with the goal of comparing different technologies as specified in CO1.
CO2. Compare and contrast NoSQL databases with RDBMS.
Justification: IT visits offer a practical setting to engage with IT experts and explore the differences between NoSQL databases and RDBMS, contributing directly to the comparison goals outlined in CO2.
CO3. Identify the data models for relevant problems.
Justification: Interacting with IT experts allows students to gain insights into real-world problems and the corresponding data models used to address them, aligning with the identification of data models outlined in CO3.
CO4. Demonstrate the basic elements of an object-relational database.
Justification: Engaging with IT experts during visits provides a platform to discuss and demonstrate the basic elements of an object-relational database, contributing directly to the goals outlined in CO4.
CO5. Compare different mobile databases.
Justification: Interacting with IT experts during visits allows students to discuss and compare different mobile databases in real-world scenarios, aligning with the goals of CO5.
CO6. Compare different NoSQL products.
Justification: IT visits provide an opportunity to discuss and compare various NoSQL products with industry experts, contributing directly to the goals of CO6.
CO7. Master CRUD and other operations on MongoDB.
Justification: Interacting with IT experts during visits facilitates discussions on practical aspects like mastering CRUD operations on MongoDB, aligning directly with the goals outlined in CO7.

Justification of Mapping of PO6 to all COs

PO6: Get industrial exposure through the 6 months Industrial Internship in IT industry.
CO1. Compare different database technologies.
Justification: The industrial internship provides a practical platform to compare different database technologies in real-world scenarios, aligning directly with the goals of CO1.
CO2. Compare and contrast NoSQL databases with RDBMS.
Justification: The industrial internship involves hands-on experience in comparing and contrasting NoSQL databases with RDBMS, contributing directly to the goals of CO2.
CO3. Identify the data models for relevant problems.
Justification: Working on real-world projects during the industrial internship requires students to identify and apply appropriate data models for specific problems, aligning closely with the skills outlined in CO3.
CO4. Demonstrate the basic elements of an object-relational database.
Mapping: 3 - Strongly related

Justification: The industrial internship offers a practical environment for students to actively demonstrate their understanding of the basic elements of an object-relational database, reinforcing the theoretical knowledge outlined in CO4.

CO5. Compare different mobile databases.

Justification: The industrial internship provides an opportunity to compare different mobile databases in a real-world context, aligning directly with the goals of CO5.

CO6. Compare different NoSQL products.

Justification: The industrial internship involves exposure to various NoSQL products and their applications, directly supporting the goals of CO6.

CO7. Master CRUD and other operations on MongoDB.

Justification: The industrial internship provides a hands-on opportunity to master practical operations like CRUD on MongoDB, aligning directly with the goals outlined in CO7.

Justification of Mapping of PO7 to all COs

PO7: To make them employable according to the current demand of the IT Industry and a responsible citizen.

CO1. Compare different database technologies.

Justification: The ability to compare different database technologies is directly linked to employability in the IT industry, meeting the demand for a well-rounded skill set as specified in PO7.

CO2. Compare and contrast NoSQL databases with RDBMS.

Justification: The comparison between NoSQL databases and RDBMS is crucial for employability in the IT industry, aligning with the current demand for versatile skills, as outlined in PO7.

CO3. Identify the data models for relevant problems.

Justification: Identifying data models is a fundamental skill for employability, directly contributing to the ability to address relevant problems in the IT industry, in line with the goals of PO7.

CO4. Demonstrate the basic elements of an object-relational database.

Justification: Demonstrating the basic elements of an object-relational database is directly related to employability, showcasing practical skills required in the IT industry, as per the objectives of PO7.

CO5. Compare different mobile databases.

Justification: Comparing different mobile databases is aligned with the current demand for skills in mobile technologies, contributing directly to employability in the IT industry, as specified in PO7.

CO6. Compare different NoSQL products.

Justification: Comparing different NoSQL products is directly related to staying current with industry trends, enhancing employability in the IT industry, in accordance with the goals of PO7.

CO7. Master CRUD and other operations on MongoDB.

Justification: Mastering CRUD operations on MongoDB is a practical skill directly relevant to the IT industry, contributing to employability as outlined in PO7.

Class: M.sc Computer science) (Semester-I) **Paper Code:PSCS114**

Title of Paper: Design & Analysis of Algorithm **Paper: IV**

Credit:4 **No. of lectures: 60**

Course objective:
Student successfully completing this course will be able to

- Understand Basic Algorithm Analysis techniques and the use o- asymptotic notation
- Understand different design strategies
- Understand the use of data structures in improving algorithm performance
- Understand classical problem and solutions
- Learn a variety of useful algorithms
- Understand classification of problems

Course Outcomes:

CO1: Understand the Basic algorithms and data structure concepts.

CO2: Understand Basic programming concepts.

CO3: To develop the ability to analyze the running time and prove thecorrectness of

basic algorithms

CO4: To be able to design efficient algorithms for moderately difficult computational problems, using various algorithm design techniques taught in the course.

CO5: To be able to prove the hardness of NP-Hard problems using simple reductions.

CO6: To be able to do performance analysis of simple approximation algorithms.

CO7: Able to analyze String matching algorithms.

Units	Title and Contents	No. of Lectures
Unit–I	**Analysis & Design Strategies** Algorithm definition, space complexity, time complexity, worst case–best case –average case complexity, asymptotic notation, sorting algorithms(insertion sort, heap sort) sorting in linear time, searching algorithms, recursive algorithms (Tower of Hanoi, Permutations). **Divide and Conquer**-control abstraction, binary search, merge sort, Quick sort, Strassen's matrix Multiplication	**10**
Unit–II	**Greedy Method** Knapsack problem ,job sequencing with deadlines ,minimum-costs Spanning trees**,** Kruskal's and Prim's algorithm, optimal storage on tapes, optimal merge patterns, Huffman coding	**10**
Unit–III	**Dynamic programming** Matrix chain multiplication, single source shortest paths, Dijkstra's algorithm, Bellman-ford algorithm, all pairs shortest path, longest common subsequence, string editing , 0/1knapsackproblem,Travelingsalesperson problem.	**10**
Unit–IV	**Decrease and conquer** DFS and BFS, Topological sorting, Strongly connected components	**6**

Unit-V	**Backtracking & Branch and Bound Technique** General method, 8 queen’s problem, sum of subset problem, graph coloring problem, Hamiltonian cycle. FIFO, LIFO, LCBB, TSP problem,0/1knapsackProblem.	**8**
Unit–VI	**Transform and Conquer &Problem Classification** Horner’s Rule and Binary Exponentiation–Problem Reduction Non deterministic algorithm, The class of P,NP, NP-hard and NP-Complete problems, significance of Cook’s theorem.	**6**

NOTE: 50 LECTURE FOR CURRICULUM (TEACHING)&10LECTURESFORLEARNING

References:

1. EllisHorowitz,SartajSahni&SanguthevarRajasekaran,ComputerAlgorithms,Galgotia.

2. T.Cormen,C.Leiserson,&R.Rivest,Algorithms,MITPress,19901

3. A.Aho,J.Hopcroft,&J.Ullman,TheDesignandAnalysisofComputerAlgorithms,AddisonWesley,1974

4. DonaldKnuth,TheArtofComputerProgramming(3vols.,variouseditions,1973-81),AddisonWesley

5. StevenSkiena,TheAlgorithmManual,SpringerISBN:9788184898651

6. Jungnickel,Graphs,NetworksandAlgorithms,Springer,ISBN:354021905

Mapping of this course with Program Outcomes

Course Outcomes	**Program Outcomes (POs)**						
	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	2	3	2	1	3	3
CO2	3	2	3	2	1	3	3
CO3	3	2	3	2	1	3	3
CO4	3	2	3	3	1	3	3
CO5	3	2	3	3	1	3	3
CO6	3	2	3	2	1	3	3
CO7	3	2	3	2	1	3	3

Weight: 1 - Partially related 2 - Moderately Related 3 - Strongly related

1. **Justification of PO1 to ALL CO’s:**

CO1: Understand the Basic algorithms and data structure concepts.
PO1: Enrich the knowledge in the areas like Design and Analysis of Algorithms.
Mapping: 3 (Strongly related)
CO2: Understand Basic programming concepts.
PO1: Enrich the knowledge in the areas like Paradigm of Programming language.
Mapping: 3 (Strongly related)
CO3: To develop the ability to analyze the running time and prove the correctness of basic algorithms.
PO1: Enrich the knowledge in the areas like Design and Analysis of Algorithms.
Mapping: 3 (Strongly related)
CO4: To be able to design efficient algorithms for moderately difficult computational problems, using various algorithm design techniques taught in the course.

PO1: Enrich the knowledge in the areas like Design and Analysis of Algorithms.
Mapping: 3 (Strongly related)
CO5: To be able to prove the hardness of NP-Hard problems using simple reductions.
PO1: Enrich the knowledge in the areas like Design and Analysis of Algorithms.
Mapping: 3 (Strongly related)
CO6: To be able to do performance analysis of simple approximation algorithms.
PO1: Enrich the knowledge in the areas like Design and Analysis of Algorithms.
Mapping: 3 (Strongly related)
CO7: Able to analyze String matching algorithms.
PO1: Enrich the knowledge in the areas like Design and Analysis of Algorithms.
Mapping: 3 (Strongly related)
Overall, all the Course Outcomes (CO) is strongly related to the Program Outcome (PO) - "Enrich the knowledge in the areas like Design and Analysis of Algorithms."

2. Justification of PO2 to ALL CO's:

CO1: Understand the Basic algorithms and data structure concepts.
PO2: Students understand all dimensions of the concepts of software application and projects.
Mapping: 2 (Moderately related)
CO2: Understand Basic programming concepts.
PO2: Students understand all dimensions of the concepts of software application and projects.
Mapping: 2 (Moderately related)
CO3: To develop the ability to analyze the running time and prove the correctness of basic algorithms.
PO2: Students understand all dimensions of the concepts of software application and projects.
Mapping: 2 (Moderately related)
CO4: To be able to design efficient algorithms for moderately difficult computational problems, using various algorithm design techniques taught in the course.
PO2: Students understand all dimensions of the concepts of software application and projects.
Mapping: 2 (Moderately related)
CO5: To be able to prove the hardness of NP-Hard problems using simple reductions.
PO2: Students understand all dimensions of the concepts of software application and projects.
Mapping: 2 (Moderately related)

CO6: To be able to do performance analysis of simple approximation algorithms.
PO2: Students understand all dimensions of the concepts of software application and projects.
Mapping: 2 (Moderately related)
CO7: Able to analyze String matching algorithms
PO2: Students understand all dimensions of the concepts of software application and projects.
Mapping: 2 (Moderately related)
Overall, all the Course Outcomes (CO) are moderately related to the Program Outcome (PO) - "Students understand all dimensions of the concepts of software application and projects."

3. Justification of PO3 to ALL CO's:

CO1: Understand the Basic algorithms and data structure concepts
PO3: Students understand the computer subjects with the demonstration of all programming and theoretical concepts with the use of ICT.
Mapping: 3 (Strongly related)
CO2: Understand Basic programming concepts.
PO3: Students understand the computer subjects with the demonstration of all programming and theoretical concepts with the use of ICT

Mapping: 3 (Strongly related)
CO3: To develop the ability to analyze the running time and prove the correctness of basic algorithms.
PO3: Students understand the computer subjects with the demonstration of all programming and theoretical concepts with the use of ICT.
Mapping: 3 (Strongly related)
CO4: To be able to design efficient algorithms for moderately difficult computational problems, using various algorithm design techniques taught in the course.
PO3: Students understand the computer subjects with the demonstration of all programming and theoretical concepts with the use of ICT.
Mapping: 3 (Strongly related)
CO5: To be able to prove the hardness of NP-Hard problems using simple reductions.
PO3: Students understand the computer subjects with the demonstration of all programming and theoretical concepts with the use of ICT.
Mapping: 3 (Strongly related)
CO6: To be able to do performance analysis of simple approximation algorithms.
PO3: Students understand the computer subjects with the demonstration of all programming and theoretical concepts with the use of ICT.
Mapping: 3 (Strongly related)
CO7: Able to analyze String matching algorithms.
PO3: Students understand the computer subjects with the demonstration of all programming and theoretical concepts with the use of ICT.
Mapping: 3 (Strongly related)
Overall, all the Course Outcomes (CO) are strongly related to the Program Outcome (PO) - "Students understand the computer subjects with the demonstration of all programming and theoretical concepts with the use of ICT."

4.Justification of PO4 to ALL CO's:

CO1: Understand the Basic algorithms and data structure concepts.
PO4: Developed in-house applications in terms of projects.
Mapping: 2 (Moderately related)

CO2: Understand Basic programming concepts.
PO4: Developed in-house applications in terms of projects.
Mapping: 2 (Moderately related)
CO3: To develop the ability to analyze the running time and prove the correctness of basic algorithms.
PO4: Developed in-house applications in terms of projects.
Mapping: 2 (Moderately related)
CO4: To be able to design efficient algorithms for moderately difficult computational problems, using various algorithm design techniques taught in the course.
PO4: Developed in-house applications in terms of projects.
Mapping: 3 (Strongly related)
CO5: To be able to prove the hardness of NP-Hard problems using simple reductions.
PO4: Developed in-house applications in terms of projects.
Mapping: 3 (Strongly related)
CO6: To be able to do performance analysis of simple approximation algorithms.
PO4: Developed in-house applications in terms of projects.
Mapping: 2 (Moderately related)
CO7: Able to analyze String matching algorithms.
PO4: Developed in-house applications in terms of projects

Mapping: 2 (Moderately related)
Overall, CO4 and CO5 are strongly related to PO4, as they directly involve the design and development of algorithms for computational problems. CO1, CO2, CO3, CO6, and CO7 are moderately related to PO4, as they contribute to the foundational understanding and skills needed for software development projects.

5.Justification of PO5 to ALL CO's:

CO1: Understand the Basic algorithms and data structure concepts.
PO5: Interact with IT experts & knowledge by IT visits.
Mapping: 1 (Partially related)
CO2: Understand Basic programming concepts.
PO5: Interact with IT experts & knowledge by IT visits.
Mapping: 1 (Partially related)
CO3: To develop the ability to analyze the running time and prove the correctness of basic algorithms.
PO5: Interact with IT experts & knowledge by IT visits.
Mapping: 1 (Partially related)
CO4: To be able to design efficient algorithms for moderately difficult computational problems, using various algorithm design techniques taught in the course.
PO5: Interact with IT experts & knowledge by IT visits.
Mapping: 1 (Partially related)
CO5: To be able to prove the hardness of NP-Hard problems using simple reductions.
PO5: Interact with IT experts & knowledge by IT visits.
Mapping: 1 (Partially related)
CO6: To be able to do performance analysis of simple approximation algorithms.
PO5: Interact with IT experts & knowledge by IT visits.
Mapping: 1 (Partially related)
CO7: Able to analyze String matching algorithms.
PO5: Interact with IT experts & knowledge by IT visits.
Mapping: 1 (Partially related)

Overall, all the Course Outcomes (CO) are partially related to the Program Outcome (PO) - "Interact with IT experts & knowledge by IT visits." The theoretical and algorithmic understanding gained through the courses may not be directly tied to the experiential learning and interaction with IT experts during visits.

6.Justification of PO6 to ALL CO's:

CO1: Understand the Basic algorithms and data structure concepts.
PO6: Get industrial exposure through the 6 months Industrial Internship in IT industry.
Mapping: 3 (Strongly related)
CO2: Understand Basic programming concepts.
PO6: Get industrial exposure through the 6 months Industrial Internship in IT industry.
Mapping: 3 (Strongly related)
CO3: To develop the ability to analyze the running time and prove the correctness of basic algorithms.
PO6: Get industrial exposure through the 6 months Industrial Internship in IT industry.
Mapping: 3 (Strongly related)
CO4: To be able to design efficient algorithms for moderately difficult computational problems, using various algorithm design techniques taught in the course.
PO6: Get industrial exposure through the 6 months Industrial Internship in IT industry.
Mapping: 3 (Strongly related)
CO5: To be able to prove the hardness of NP-Hard problems using simple reductions.
PO6: Get industrial exposure through the 6 months Industrial Internship in IT industry.

Mapping: 3 (Strongly related)
CO6: To be able to do performance analysis of simple approximation algorithms.
PO6: Get industrial exposure through the 6 months Industrial Internship in IT industry.
Mapping: 3 (Strongly related)
CO7: Able to analyze String matching algorithms.
PO6: Get industrial exposure through the 6 months Industrial Internship in IT industry.
Mapping: 3 (Strongly related)
Overall, all the Course Outcomes (CO) are strongly related to the Program Outcome (PO) - "Get industrial exposure through the 6 months Industrial Internship in IT industry." The skills and knowledge acquired in the courses directly contribute to the practical experience gained during the industrial internship.

7.Justification of PO7 to ALL CO's:

CO1: Understand the Basic algorithms and data structure concepts.
PO7: To make them employable according to the current demand of the IT Industry and responsible citizens. Aware them to publish their work in reputed journals.
CO1 with PO7: 3 (Strongly related)
CO2: Understand Basic programming concepts.
PO7: To make them employable according to the current demand of the IT Industry and responsible citizens. Aware them to publish their work in reputed journals.
CO2 with PO7: 3 (Strongly related)
CO3: To develop the ability to analyze the running time and prove the correctness of basic algorithms.
PO7: To make them employable according to the current demand of the IT Industry and responsible citizens.Aware them to publish their work in reputed journals.
CO3 with PO7: 3 (Strongly related)
CO4: To be able to design efficient algorithms for moderately difficult computational problems, using various algorithm design techniques taught in the course.
PO7: To make them employable according to the current demand of the IT Industry and responsible citizens. Aware them to publish their work in reputed journals.
CO4 with PO7: 3 (Strongly related)
CO5: To be able to prove the hardness of NP-Hard problems using simple reductions.
PO7: To make them employable according to the current demand of the IT Industry and responsible citizens.Aware them to publish their work in reputed journals.
CO5 with PO7: 3 (Strongly related)
CO6: To be able to do performance analysis of simple approximation algorithms.
PO7: To make them employable according to the current demand of the IT Industry and responsible citizens.Aware them to publish their work in reputed journals.
CO6 with PO7: 3 (Strongly related)
CO7: Able to analyze String matching algorithms.
PO7: To make them employable according to the current demand of the IT Industry and responsible citizens.Aware them to publish their work in reputed journals.
CO7 with PO7: 3 (Strongly related)
Overall, all the Course Outcomes (CO) are strongly related to the Program Outcome (PO7) - "To make them employable according to the current demand of the IT Industry and responsible citizens." However, when considering the awareness to publish work in reputed journals (PS08), the mapping is partially related (weightage 1).

Class: M.Sc. (Computer science) I (Semester-I)	**Paper Code: PSCS115**
Title of Paper: Dot Net Framework &C#	**Paper: V**
Credit: 4	**No.oflectures:60**

Learning Objectives:

Able to understand the DOTNET framework, C# language features and Windows applicationdevelopmentusingC#.Net.

Course Outcomes:

CO1.Ability to write the Visualized programming and design different real life problems

CO2.Explain the three pillars of object oriented programming.

CO3.Develop working knowledge of C# programming constructs and the .NET Framework.

CO4.Write an object oriented program using custom classes.

CO5.Build and debug well-formed Web Forms with ASP.

CO6.Perform form validation with validation controls.

CO7.Create simple data binding applications

PartI: C#		
1.	**Introduction to DOTNET Framework& C#** **a.** Introduction to DOTNET **b.** DOT NET class framework **c.** Common Language Runtime i. Overview ii. Elements of .NET application iii. MemoryManagement iv. Garbage Collector **d. User and Program Interface** **e. Language features** i. Variables and Expressions, type conversion ii. Flow Control iii. Functions, Delegates iv. Debugging and error handling, exception handling (System Defined and User Defined) **f. Object Oriented Concepts** i. Defining classes, class members, Interfaces, properties ii. Access modifiers, Implementation of class, interface and properties iii. Concept of hiding base class methods, Overriding iv. Event Handling **g. Collections, Comparisons and Conversions** i. Defining and using collections, Indexers, iterators ii. Type comparison, Value Comparison **h. Generics** **i.** Using generics, ii. Defining Generics, iii. Generic Interfaces iv. Generic methods v. Generic Delegate	**10**

2.	**Window Programming** **a. Window Controls** i. Common Controls	**10**
	ii. Container Controls iii. Menus and Toolbars iv. Printing v. Dialogs vi. Data tools **b.** Deploying Window Application i. Deployment Overview ii. Building the project: Installation **c.** Data Handling i. File SystemData ii. XML Data, JSON iii. DatabasesandADO.NET **d.** Reporting Tools i. Data Report ii. Crystal Report	
3.	**Dot NET Assemblies** a. Components b. .NET Assembly features c. Structure of Assemblies d. Calling assemblies, private and shared assemblies	**2**
4.	**LINQ** a. Operators b. SQL c. Objects d. Dataset e. XML f. Entities g. Lambda Expressions	**6**
5.	**Entity Framework** **a.** Overview **b.** Architecture **c.** Environment setup a. Data base Setup b. Entity Data Model c. DB Context d. Entity Types e. Entity Relationships f. EntityLifecycle EntityApproaches	**10**
6.	**Data base Operations** a. CRUD b. Concurrency c. Transactions d. Views e. Index f. Stored Procedures g. Disconnected Entities h. Table Valued Functions i. Native SQL j. Projection Quires	**10**

Recommended Text and Reference books:

1. Beginning VisualC#, Wrox Publication
2. Professional VisualC#,WroxPublication
3. InsideC#,byTomArcherISBN:0735612889MicrosoftPressÂ©2001,403pages
4. BeginningASP.NET3.5,WroxPublication
5. ProgrammingASP.NET3.5byJesseLiberty,DanMaharry,DanHurwitz,O'Reilly
6. IllustratedC#2008,Solis,PublicationAPRESS,ISBN978-81-8128-958-2
7. ProfessionalC#4.0and.NET4byChristianNagel,BillEvjen,JayGlynn,KarliWatson,
8. Morgan Skinner, WROX
9. Beginning C# Object-Oriented Programming By Dan Clark, Apress
10. ADO.NETExamplesandBestPracticesforC#Programmers,ByPeterD.BlackburnApress
11. Database Programming with C#, By Carsten Thomsen, Apress
12. https://www.tutorialspoint.com

Mapping of this Course Outcomes with Programme Outcomes

Course Outcomes	**Programme Outcomes (POs)**						
	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	3	3	3	2	3	3
CO2	3	3	3	3	3	3	3
CO3	3	3	3	3	2	3	2
CO4	3	3	3	3	2	2	3
CO5	3	2	3	3	2	3	3
CO6	2	2	2	2	3	3	3
CO7	2	1	2	2	2	3	2

Mapping of PO1 With All CO's with Justification :
CO1: with PO1: as it aligns with the goal of enriching knowledge in areas like Artificial Intelligence, Web Services, Cloud Computing, and other core computing subjects.
CO2: with PO1: as understanding the principles of object-oriented programming is fundamental in areas like software design, which is part of the enriched knowledge in programming languages and design paradigms.
CO3: With PO1: as C# and .NET Framework are key components in the study of programming
Languages, web services, and software project management.
CO4: With PO1: as it directly involves applying knowledge of object-oriented programming inCreating software solutions, contributing to the enriched knowledge in programming languages andsoftware design.
CO5: With PO1: as it involves practical skills in web development and aligns with the areas of webservices, cloud computing, and mobile technologies.
CO6: With PO1 : as form validation is part of web development and contributes to knowledge in webservices, but it&#39;s not as broad as some other aspects covered in PO1.
CO7: With PO1: as data binding is a specific aspect of web development and contributes to
knowledge in web services and database technologies, but it may not cover a broad range of areasmentioned in PO1.

Mapping of PO2 With All CO’s with Justification :

CO1: With PO2 : as the ability to write visualized programs and design real-life problems contributes to a comprehensive understanding of software application concepts and projects.

CO2: With PO2 : as understanding the three pillars of object-oriented programming is fundamental to grasping the conceptual dimensions of software application and projects.

CO3: With PO2: as acquiring a working knowledge of C# and the .NET Framework is essential forunderstanding software application concepts and projects.

CO4: With PO2 : as writing object-oriented programs with custom classes demonstrates practicalapplication, contributing to a comprehensive understanding of software application concepts andprojects.

CO5: With PO2 : as building and debugging web forms with ASP contributes to understandingsoftware applications, but it may not cover all dimensions of software projects.

CO6: With PO2: as form validation is a specific aspect of software applications and contributes tounderstanding certain dimensions of projects, but it&#39;s not as comprehensive as other aspects.

CO7: With PO2 : as creating data binding applications contributes to understanding certain aspectsof software applications but may not cover all dimensions of software projects.

Mapping of PO3 With All CO’s with Justification :

CO1: With PO3 : as the ability to write visualized programs and design real-life problems
Demonstrates understanding through practical application, facilitated by the use of Information andCommunication Technology (ICT).

CO2: With PO3 : as explaining the three pillars of object-oriented programming involves theoreticalunderstanding and can be demonstrated through ICT tools, enhancing the comprehension ofcomputer subjects.

CO3: With PO3 : as developing working knowledge using ICT tools in C# and .NET contributes to apractical and theoretical understanding of computer subjects.

CO4: With PO3 : as writing object-oriented programs using custom classes, especially with the use ofICT, demonstrates a practical application of theoretical concepts in computer subjects.

CO5: With PO3: as building and debugging web forms using ASP with the aid of ICT tools enhances the practical and theoretical understanding of computer subjects, especially in web development.

CO6: With PO3: as performing form validation involves practical application with ICT tools,
Contributing to the understanding of certain dimensions of computer subjects.

CO7: With PO3 : as creating data binding applications with the use of ICT tools contributes to apractical understanding of certain aspects of computer subjects.

Mapping of PO4 With All CO’s with Justification :

CO1: With PO4 : as the ability to write visualized programs and design real-life problems is crucial inthe development of in-house applications as it involves translating conceptual ideas into practicalsolutions.

CO2: With PO4 : as understanding the three pillars of object-oriented programming is fundamentalin the development of in-house applications, guiding the design and structure of the software.

CO3: With PO4: as a working knowledge of C# and the .NET Framework is essential for effectivelydeveloping in-house applications using these technologies.

CO4: With PO4: as writing object-oriented programs with custom classes is a practical skill necessaryfor the development of in-house applications, ensuring the creation of well-structured andmaintainable code.

CO5: With PO4 : as building and debugging web forms with ASP is directly applicable to the development of web-based in-house applications.

CO6: With PO4 : as form validation is a specific aspect of developing user-friendly in-house applications, contributing to the overall quality of the software.

CO7:With PO4: as creating data binding applications is relevant to the development of in-houseapplications, especially when dealing with data integration and user interface design.

Mapping of PO5 With All CO’s with Justification :

CO1: With PO5 : as the ability to write visualized programs and design real-life problems contributesto skills useful during interactions with IT experts but may not be the sole focus of such visits.CO2: With PO5: as understanding the three pillars of object-oriented programming provides afoundational knowledge useful for interacting with IT experts but may not be the primary focus of
such visits.
CO3: With PO5 : as developing a working knowledge of C# and the .NET Framework is beneficial fordiscussions with IT experts but might not be the sole purpose of IT visits.CO4: With PO5 : as writing object-oriented programs with custom classes contributes to the practicalskills that can be discussed during interactions with IT experts but may not be the central focus ofsuch visits.CO5: With PO5 : as building and debugging web forms with ASP is a practical skill that can be relevant during interactions with IT experts but may not be the primary objective of IT visits.CO6: With PO5 : as performing form validation is a specific skill that can be discussed during interactions with IT experts but may not be the primary purpose of such visits.CO7: With PO5 : as creating data binding applications can be a topic of discussion during interactions with IT experts but may not be the central theme of IT visits.
Mapping of PO6 With All CO's with Justification :CO1: With PO6: as the ability to write visualized programs and design real-life problems is crucial forsuccess fully undertaking and contributing to industrial internship projects in the IT industry.CO2: With PO6: as understanding the three pillars of object-oriented programming is essential forgrasping software design principles and applying them in real-world industrial projects during theinternship.CO3: With PO6: as a working knowledge of C# and the .NET Framework is directly applicable to manyindustrial projects in the IT industry, making this competency highly relevant during the internship.CO4: With PO6: as writing object-oriented programs with custom classes is a practical skill that canbe directly applied in the development of software solutions during the industrial internship.CO5: With PO6: as building and debugging web forms with ASP is a practical skill that aligns withmany web-based projects encountered in the IT industry during the internship.

CO6: With PO6: as performing form validation is a practical aspect that contributes to the quality ofuser interfaces, which is often a requirement in industrial projects undertaken during the internship.
CO7: With PO6: as creating data binding applications is directly applicable to projects involving dataintegration, a common aspect of industrial IT projects during the internship.
Mapping of PO7 With All CO's with Justification :
CO1: With PO7: as the ability to write visualized programs and design real-life problems is a
fundamental skill that directly contributes to employability in the IT industry by demonstratingpractical problem-solving capabilities.
CO2: With PO7: as understanding the three pillars of object-oriented programming is a foundationalknowledge that is highly relevant to the demands of the IT industry, contributing to theemployability of graduates.
CO3: With PO7: as developing a working knowledge of C# and the .NET Framework is directly alignedwith the current demands of the IT industry, enhancing the employability of students.
CO4: With PO7: as the ability to write object-oriented programs with custom classes is a practicalskill that is in high demand in the IT industry, making graduates more employable.
CO5: With PO7: as building and debugging web forms with ASP is a practical skill that is directlyapplicable to web development, a field with significant demand in the IT industry.
CO6: With PO7 : as form validation is a practical aspect that contributes to the development ofrobust and user-friendly applications, enhancing the employability of graduates.
CO7: With PO7 : as creating data binding applications is a practical skill that aligns with the industrydemand for professionals who can work with data integration, making graduates more employable.

Class: M.Sc. (Computer science)-I (Semester-I)	**Paper Code: PSCS116**
Title of Paper: Lab Course On DOT NET & DT	**Paper: VI (Lab Course)**
Credit:4(3Hr.Practical/week/batch)	**No. of Practical's: 12**

Course Outcomes:

CO1.Ability to write the Visualized programming and design different real life problems

CO2.Explain the three pillars of object oriented programming.

CO3.Develop working knowledge of C# programming constructs and the .NET Framework.

CO4.Write an object oriented program using custom classes.

CO5.Build and debug well-formed Web Forms with ASP.

CO6.Perform form validation with validation controls.

CO7.Create simple data binding applications.

Console Application	
Assignment1	Parameter Modifiers (ref, out, params) Delegates
Assignment2	Polymorphism, Exception Handling, Collection, Generics
Windows Application	
Assignment3	Windows Controls
Assignment4	ADO .Net
Assignment5	Crystal Report
Entity Framework	
Assignment6	CRUD Operations using Entity Framework
Database Technologies Assignments	
Assignment7	Mango DB: Creating collections and inserting Documents
Assignment8	Mango DB: Querying the collections

Mapping of this Course Outcomes with Programme Outcomes

Course Outcomes	**Programme Outcomes (POs)**						
	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	3	3	3	2	3	3
CO2	3	3	3	3	2	3	3
CO3	3	3	3	3	2	3	3
CO4	3	3	3	3	2	3	3
CO5	3	3	3	3	2	3	3
CO6	3	3	3	3	2	3	3
CO7	3	3	3	3	2	3	3

Mapping of PO With All CO’s with Justification :
All CO: With PO1 : as the ability to write visualized programs and design real-life problems contributes to a comprehensive understanding of software application concepts and projects.
All CO: With PO2 : as understanding the three pillars of object-oriented programming is fundamental to grasping the conceptual dimensions of software application and projects.
All CO: With PO3: as acquiring a working knowledge of C# and the .NET Framework is essential forunder standing software application concepts and projects.
All CO: With PO4 : as writing object-oriented programs with custom classes demonstrates practical application, contributing to a comprehensive understanding of software application concepts and projects.
All CO: With PO5 : as building and debugging web forms with ASP contributes to understanding software applications, but it may not cover all dimensions of software projects.
All CO: With PO6: as form validation is a specific aspect of software applications and contributes to understanding certain dimensions of projects, but it&#39;s not as comprehensive as other aspects.
All CO: With PO7 : as creating data binding applications contributes to understanding certain aspects of software applications but may not cover all dimensions of software projects.