MANAGEMENT, ENTREPRENEURSHIP AND CYBER LAW
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2016 -2017)
SEMESTER – V

Subject Code	15CS51	IA Marks	20
Number of Lecture Hours/Week	4	Exam Marks	80
Total Number of Lecture Hours	50	Exam Hours	03

CREDITS – 04

Course objectives: This course will enable students to

- Explain the principles of management, organization and entrepreneur.
- Discuss on planning, staffing, ERP and their importance
- Infer the importance of intellectual property rights and relate the institutional support

Module – 1	Teaching Hours
Introduction – Meaning, nature and characteristics of management, scope and functional areas of management, goals of management, levels of management, brief overview of evolution of management. Planning- Nature, importance, types of plans, steps in planning, Organizing- nature and purpose, types of organization.	**10 Hours**
Module – 2	
Staffing- meaning, process of recruitment and selection. Directing and controlling- meaning and nature of directing, leadership styles, motivation theories. Controlling- meaning, steps in controlling, methods of establishing control, Communication- Meaning and importance, Coordination- meaning and importance	**10 Hours**
Module – 3	
Entrepreneur – meaning of entrepreneur, types of entrepreneurship, stages of entrepreneurial process, role of entrepreneurs in economic development, entrepreneurship in India, barriers to entrepreneurship. Identification of business opportunities- market feasibility study, technical feasibility study, financial feasibility study and social feasibility study.	**10 Hours**
Module – 4	
Preparation of project and ERP - meaning of project, project identification, project selection, project report, need and significance of report, contents, formulation, guidelines by planning commission for project report **Enterprise Resource Planning: Meaning and Importance- ERP** and Functional areas of Management – Marketing / Sales- Supply Chain Management – Finance and Accounting – Human Resources – Types of reports and methods of report generation	**10 Hours**
Module – 5	
Small Scale Industry: case study(Microsoft), Shahnaz Husain(Ayurveda Entrepreneur), Government policy towards SSI, Case study(Captain G R Gopinath), case study (N R Narayana Murthy & Infosys) **Institutional support: case study:** Amar Gopal Bose and Bose corporation, supporting agencies of Govt for SSI, Different schemes, Types of Help, Case study Dr Devi Prasad Shetty. **Introduction to IPR.**	**10 Hours**

Course outcomes: The students should be able to:

- Define management, organization, entrepreneur, planning, staffing, ERP and outline their importance in entrepreneurship

• Utilize the resources available effectively through ERP
• Make us of IPRs and institutional support in entrepreneurship

Question paper pattern:

The question paper will have TEN questions.

There will be TWO questions from each module.

Each question will have questions covering all the topics under a module.

The students will have to answer FIVE full questions, selecting ONE full question from each module.

Text Books:

1. Principles of Management -P. C. Tripathi, P. N. Reddy; Tata McGraw Hill, 4th Edition, 2010.
2. Dynamics of Entrepreneurial Development & Management -Vasant Desai Himalaya Publishing House.
3. Entrepreneurship Development -Small Business Enterprises -Poornima M Charantimath Pearson Education – 2006.
4. Management and Entrepreneurship - Kanishka Bedi- Oxford University Press-2017

Reference Books:

1. Management Fundamentals -Concepts, Application, Skill Development Robert Lusier – Thomson.
2. Entrepreneurship Development -S S Khanka -S Chand & Co.
3. Management -Stephen Robbins -Pearson Education /PHI -17th Edition, 2003

COMPUTER NETWORKS [As per Choice Based Credit System (CBCS) scheme] (Effective from the academic year 2016 -2017) SEMESTER – V			
Subject Code	15CS52	IA Marks	20
Number of Lecture Hours/Week	4	Exam Marks	80
Total Number of Lecture Hours	50	Exam Hours	03
CREDITS – 04			

Course objectives: This course will enable students to

- Demonstration of application layer protocols
- Discuss transport layer services and understand UDP and TCP protocols
- Explain routers, IP and Routing Algorithms in network layer
- Disseminate the Wireless and Mobile Networks covering IEEE 802.11 Standard
- Illustrate concepts of Multimedia Networking, Security and Network Management

Module – 1	Teaching Hours
Application Layer: Principles of Network Applications: Network Application Architectures, Processes Communicating, Transport Services Available to Applications, Transport Services Provided by the Internet, Application-Layer Protocols. The Web and HTTP: Overview of HTTP, Non-persistent and Persistent Connections, HTTP Message Format, User-Server Interaction: Cookies, Web Caching, The Conditional GET, File Transfer: FTP Commands & Replies, Electronic Mail in the Internet: SMTP, Comparison with HTTP, Mail Message Format, Mail Access Protocols, DNS; The Internet's Directory Service: Services Provided by DNS, Overview of How DNS Works, DNS Records and Messages, Peer-to-Peer Applications: P2P File Distribution, Distributed Hash Tables, Socket Programming: creating Network Applications: Socket Programming with UDP, Socket Programming with TCP. **T1: Chap 2**	**10 Hours**
Module – 2	
Transport Layer : Introduction and Transport-Layer Services: Relationship Between Transport and Network Layers, Overview of the Transport Layer in the Internet, Multiplexing and Demultiplexing: Connectionless Transport: UDP,UDP Segment Structure, UDP Checksum, Principles of Reliable Data Transfer: Building a Reliable Data Transfer Protocol, Pipelined Reliable Data Transfer Protocols, Go-Back-N, Selective repeat, Connection-Oriented Transport TCP: The TCP Connection, TCP Segment Structure, Round-Trip Time Estimation and Timeout, Reliable Data Transfer, Flow Control, TCP Connection Management, Principles of Congestion Control: The Causes and the Costs of Congestion, Approaches to Congestion Control, Network-assisted congestion-control example, ATM ABR Congestion control, TCP Congestion Control: Fairness. **T1: Chap 3**	**10 Hours**
Module – 3	
The Network layer: What's Inside a Router?: Input Processing, Switching, Output Processing, Where Does Queuing Occur? Routing control plane, IPv6,A Brief foray into IP Security, Routing Algorithms: The Link-State (LS) Routing Algorithm, The Distance-Vector (DV) Routing Algorithm, Hierarchical Routing,	**10 Hours**

Routing in the Internet, Intra-AS Routing in the Internet: RIP, Intra-AS Routing in the Internet: OSPF, Inter/AS Routing: BGP, Broadcast and Multicast Routing: Broadcast Routing Algorithms and Multicast. **T1: Chap 4: 4.3-4.7**	
Module – 4	
Mobile and Multimedia Networks: Cellular Internet Access: An Overview of Cellular Network Architecture, 3G Cellular Data Networks: Extending the Internet to Cellular subscribers, On to 4G:LTE,Mobility management: Principles, Addressing, Routing to a mobile node, Mobile IP, Managing mobility in cellular Networks, Routing calls to a Mobile user, Handoffs in GSM, Wireless and Mobility: Impact on Higher-layer protocols. **T1: Chap: 6 : 6.4-6.8**	**10 Hours**
Module – 5	
Multimedia Networking Applications: Properties of video, properties of Audio, Types of multimedia Network Applications, Streaming stored video: UDP Streaming, HTTP Streaming, Adaptive streaming and DASH, content distribution Networks, case studies: Netflix, You Tube and Kankan. **Network Support for Multimedia:** Dimensioning Best-Effort Networks, Providing Multiple Classes of Service, Diffserv, Per-Connection Quality-of-Service (QoS) Guarantees: Resource Reservation and Call Admission **T1: Chap: 7: 7.1,7.2,7.5**	**10 Hours**

Course outcomes: The students should be able to:

- Explain principles of application layer protocols
- Recognize transport layer services and infer UDP and TCP protocols
- Classify routers, IP and Routing Algorithms in network layer
- Understand the Wireless and Mobile Networks covering IEEE 802.11 Standard
- Describe Multimedia Networking and Network Management

Question paper pattern:
The question paper will have TEN questions.
There will be TWO questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer FIVE full questions, selecting ONE full question from each module.

Text Books:

1. James F Kurose and Keith W Ross, Computer Networking, A Top-Down Approach, Sixth edition, Pearson,2017 .

Reference Books:

1. Behrouz A Forouzan, Data and Communications and Networking, Fifth Edition, McGraw Hill, Indian Edition
2. Larry L Peterson and Brusce S Davie, Computer Networks, fifth edition, ELSEVIER
3. Andrew S Tanenbaum, Computer Networks, fifth edition, Pearson
4. Mayank Dave, Computer Networks, Second edition, Cengage Learning

DATABASE MANAGEMENT SYSTEM
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2016 -2017)
SEMESTER – V

Subject Code	15CS53	IA Marks	20
Number of Lecture Hours/Week	4	Exam Marks	80
Total Number of Lecture Hours	50	Exam Hours	03

CREDITS – 04

Course objectives: This course will enable students to

- Provide a strong foundation in database concepts, technology, and practice.
- Practice SQL programming through a variety of database problems.
- Demonstrate the use of concurrency and transactions in database
- Design and build database applications for real world problems.

Module – 1	Teaching Hours
Introduction to Databases: Introduction, Characteristics of database approach, Advantages of using the DBMS approach, History of database applications. **Overview of Database Languages and Architectures:** Data Models, Schemas, and Instances. Three schema architecture and data independence, database languages, and interfaces, The Database System environment. **Conceptual Data Modelling using Entities and Relationships:** Entity types, Entity sets, attributes, roles, and structural constraints, Weak entity types, ER diagrams, examples, Specialization and Generalization. **Textbook 1: Ch 1.1 to 1.8, 2.1 to 2.6, 7.1 to 7.8, 7.10**	10 Hours
Module – 2	
Relational Model: Relational Model Concepts, Relational Model Constraints and relational database schemas, Update operations, transactions, and dealing with constraint violations. **Relational Algebra:** Unary and Binary relational operations, additional relational operations (aggregate, grouping, etc.) Examples of Queries in relational algebra. **Mapping Conceptual Design into a Logical Design:** Relational Database Design using ER-to-Relational mapping. **SQL:** SQL data definition and data types, specifying constraints in SQL, retrieval queries in SQL, INSERT, DELETE, and UPDATE statements in SQL, Additional features of SQL. **Textbook 1: Ch 3.1, 3.2, 6.1 to 6.5, 8.1; Textbook 2: 3.5; Textbook 1: 4.1 to 4.5**	10 Hours
Module – 3	
SQL : Advances Queries: More complex SQL retrieval queries, Specifying constraints as assertions and action triggers, Views in SQL, Schema change statements in SQL. **Database Application Development:** Accessing databases from applications, An introduction to JDBC, JDBC classes and interfaces, SQLJ, Stored procedures, Case study: The internet Bookshop. **Internet Applications:** The three-Tier application architecture, The presentation layer, The Middle Tier **Textbook 1: Ch 5.1 to 5.4; Textbook 2: 6.1 to 6.6, 7.5 to 7.7.**	10 Hours
Module – 4	
Normalization: Database Design Theory – Introduction to Normalization using Functional and Multivalued Dependencies: Informal design guidelines for relation schema, Functional Dependencies, Normal Forms based on Primary Keys, Second and Third Normal Forms, Boyce-Codd Normal Form, Multivalued	10 Hours

Dependency and Fourth Normal Form, Join Dependencies and Fifth Normal Form. **Normalization Algorithms:** Inference Rules, Equivalence, and Minimal Cover, Properties of Relational Decompositions, Algorithms for Relational Database Schema Design, Nulls, Dangling tuples, and alternate Relational Designs, Further discussion of Multivalued dependencies and 4NF, Other dependencies and Normal Forms **Textbook 1: Ch 15.1 to 15.6**	
Module – 5	
Transaction Processing: Introduction to Transaction Processing, Transaction and System concepts, Desirable properties of Transactions, Characterizing schedules based on recoverability, Characterizing schedules based on Serializability, Transaction support in SQL. **Concurrency Control in Databases:** Two-phase locking techniques for Concurrency control, Concurrency control based on Timestamp ordering, Multiversion Concurrency control techniques, Validation Concurrency control techniques, Granularity of Data items and Multiple Granularity Locking. **Introduction to Database Recovery Protocols:** Recovery Concepts, NO-UNDO/REDO recovery based on Deferred update, Recovery techniques based on immediate update, Shadow paging, Database backup and recovery from catastrophic failures **Textbook 1: 20.1 to 20.6, 21.1 to 21.5, 22.1 to 22.4, 22.7.**	**10 Hours**

Course outcomes: The students should be able to:

- Identify, analyze and define database objects, enforce integrity constraints on a database using RDBMS.
- Use Structured Query Language (SQL) for database manipulation.
- Design and build simple database systems
- Design and build GUI application to interact with databases.

Question paper pattern:
The question paper will have TEN questions.
There will be TWO questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer FIVE full questions, selecting ONE full question from each module.

Text Books:
1. Database systems Models, Languages, Design and Application Programming, Ramez Elmasri and Shamkant B. Navathe, 6th Edition, Pearson.
2. Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill

Reference Books:
1. Silberschatz Korth and Sudharshan: Database System Concepts, 6th Edition, Mc-Graw Hill, 2013.
2. Coronel, Morris, and Rob, Database Principles Fundamentals of Design, Implementation and Management, Cengage Learning 2012.

<table>
<tr><td colspan="5" align="center">AUTOMATA THEORY AND COMPUTABILITY
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2016 -2017)
SEMESTER – V</td></tr>
</table>

AUTOMATA THEORY AND COMPUTABILITY				

AUTOMATA THEORY AND COMPUTABILITY
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2016 -2017)
SEMESTER – V

Subject Code	15CS54	IA Marks	20
Number of Lecture Hours/Week	4	Exam Marks	80
Total Number of Lecture Hours	50	Exam Hours	03

CREDITS – 04

Course objectives: This course will enable students to

- Introduce core concepts in Automata and Theory of Computation
- Identify different Formal language Classes and their Relationships
- Design Grammars and Recognizers for different formal languages
- Prove or disprove theorems in automata theory using their properties
- Determine the decidability and intractability of Computational problems

Module – 1	Teaching Hours
Why study the Theory of Computation, Languages and Strings: Strings, Languages. A Language Hierarchy, Computation, **Finite State Machines (FSM)**: Deterministic FSM, Regular languages, Designing FSM, Nondeterministic FSMs, From FSMs to Operational Systems, Simulators for FSMs, Minimizing FSMs, Canonical form of Regular languages, Finite State Transducers, Bidirectional Transducers. **Textbook 1: Ch 1,2, 5.1 to 5.10**	**10 Hours**
Module – 2	
Regular Expressions (RE): what is a RE?, Kleene's theorem, Applications of REs, Manipulating and Simplifying REs. Regular Grammars: Definition, Regular Grammars and Regular languages. Regular Languages (RL) and Non-regular Languages: How many RLs, To show that a language is regular, Closure properties of RLs, to show some languages are not RLs. **Textbook 1: Ch 6, 7, 8, 9: 6.1 to 6.4, 7.1, 7.2, 8.1 to 8.4**	**10 Hours**
Module – 3	
Context-Free Grammars(CFG): Introduction to Rewrite Systems and Grammars, CFGs and languages, designing CFGs, simplifying CFGs, proving that a Grammar is correct, Derivation and Parse trees, Ambiguity, Normal Forms. Pushdown Automata (PDA): Definition of non-deterministic PDA, Deterministic and Non-deterministic PDAs, Non-determinism and Halting, alternative equivalent definitions of a PDA, alternatives that are not equivalent to PDA. **Textbook 1: Ch 11, 12: 11.1 to 11.8, 12.1 to 12.6**	**10 Hours**
Module – 4	
Context-Free and Non-Context-Free Languages: Where do the Context-Free Languages(CFL) fit, Showing a language is context-free, Pumping theorem for CFL, Important closure properties of CFLs, Deterministic CFLs. Algorithms and Decision Procedures for CFLs: Decidable questions, Un-decidable questions. Turing Machine: Turing machine model, Representation, Language acceptability by TM, design of TM, Techniques for TM construction. **Textbook 1: Ch 13: 13.1 to 13.5, Textbook 2: Ch 9.1 to 9.6**	**10 Hours**
Module – 5	
Variants of Turing Machines (TM), The model of Linear Bounded automata: Decidability: Definition of an algorithm, decidability, decidable languages,	**10 Hours**

Undecidable languages, halting problem of TM, Post correspondence problem. Complexity: Growth rate of functions, the classes of P and NP, Quantum Computation: quantum computers, Church-Turing thesis. **Textbook 2: Ch 9.7 to 9.8, 10.1 to 10.7, 12.1, 12.2, 12.8, 12.8.1, 12.8.2**	

Course outcomes: The students should be able to:

- Acquire fundamental understanding of the core concepts in automata theory and Theory of Computation
- Learn how to translate between different models of Computation (e.g., Deterministic and Non-deterministic and Software models).
- Design Grammars and Automata (recognizers) for different language classes and become knowledgeable about restricted models of Computation (Regular, Context Free) and their relative powers.
- Develop skills in formal reasoning and reduction of a problem to a formal model, with an emphasis on semantic precision and conciseness.
- Classify a problem with respect to different models of Computation.

Question paper pattern:

The question paper will have TEN questions.

There will be TWO questions from each module.

Each question will have questions covering all the topics under a module.

The students will have to answer FIVE full questions, selecting ONE full question from each module.

Text Books:

1. Elaine Rich, Automata, Computability and Complexity, 1st Edition, Pearson Education,2012/2013
2. K L P Mishra, N Chandrasekaran , 3rd Edition, Theory of Computer Science, McGraw hill, 2012

Reference Books:

1. John E Hopcroft, Rajeev Motwani, Jeffery D Ullman, Introduction to AutomataTheory, Languages, and Computation, 3rd Edition, Pearson Education, 2013
2. Michael Sipser : Introduction to the Theory of Computation, 3rd edition, Cengage learning,2013
3. John C Martin, Introduction to Languages and The Theory of Computation, 3rd Edition, Tata McGraw –Hill Publishing Company Limited, 2013
4. Peter Linz, "An Introduction to Formal Languages and Automata", 3rd Edition, Narosa Publishers, 1998
5. Basavaraj S. Anami, Karibasappa K G, Formal Languages and Automata theory, Wiley India, 2012

OBJECT ORIENTED MODELING AND DESIGN
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2016 -2017)
SEMESTER – V

Subject Code	15CS551	IA Marks	20
Number of Lecture Hours/Week	3	Exam Marks	80
Total Number of Lecture Hours	40	Exam Hours	03

CREDITS – 03

Course objectives: This course will enable students to

- Describe the concepts involved in Object-Oriented modelling and their benefits.
- Demonstrate concept of use-case model, sequence model and state chart model for a given problem.
- Explain the facets of the unified process approach to design and build a Software system.
- Translate the requirements into implementation for Object Oriented design.
- Choose an appropriate design pattern to facilitate development procedure.

Module – 1	Teaching Hours
Introduction, Modelling Concepts and Class Modelling: What is Object orientation? What is OO development? OO Themes; Evidence for usefulness of OO development; OO modelling history. Modelling as Design technique: Modelling; abstraction; The Three models. Class Modelling: Object and Class Concept; Link and associations concepts; Generalization and Inheritance; A sample class model; Navigation of class models; Advanced Class Modelling, Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived Data; Packages. **Text Book-1: Ch 1, 2, 3 and 4**	**8 Hours**
Module – 2	
UseCase Modelling and Detailed Requirements: Overview; Detailed object-oriented Requirements definitions; System Processes-A use case/Scenario view; Identifying Input and outputs-The System sequence diagram; Identifying Object Behaviour-The state chart Diagram; Integrated Object-oriented Models. **Text Book-2:Chapter- 6:Page 210 to 250**	**8 Hours**
Module – 3	
Process Overview, System Conception and Domain Analysis: Process Overview: Development stages; Development life Cycle; System Conception: Devising a system concept; elaborating a concept; preparing a problem statement. Domain Analysis: Overview of analysis; Domain Class model: Domain state model; Domain interaction model; Iterating the analysis. **Text Book-1:Chapter- 10,11,and 12**	**8 Hours**
Module – 4	
Use case Realization :The Design Discipline within up iterations: Object Oriented Design-The Bridge between Requirements and Implementation; Design Classes and Design within Class Diagrams; Interaction Diagrams-Realizing Use Case and defining methods; Designing with Communication Diagrams; Updating the Design Class Diagram; Package Diagrams-Structuring the Major Components; Implementation Issues for Three-Layer Design. **Text Book-2: Chapter 8: page 292 to 346**	**8 Hours**

Module – 5	
Design Patterns: Introduction; what is a design pattern?, Describing design patterns, the catalogue of design patterns, Organizing the catalogue, How design patterns solve design problems, how to select a design patterns, how to use a design pattern; Creational patterns: prototype and singleton (only); structural patterns adaptor and proxy (only). **Text Book-3: Ch-1: 1.1, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8,Ch-3,Ch-4.**	**8 Hours**

Course outcomes: The students should be able to:

- Describe the concepts of object-oriented and basic class modelling.
- Draw class diagrams, sequence diagrams and interaction diagrams to solve problems.
- Choose and apply a befitting design pattern for the given problem.

Question paper pattern:
The question paper will have TEN questions.
There will be TWO questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer FIVE full questions, selecting ONE full question from each module.

Text Books:
1. Michael Blaha, James Rumbaugh: Object Oriented Modelling and Design with UML,2nd Edition, Pearson Education,2005
2. Satzinger, Jackson and Burd: Object-Oriented Analysis & Design with the Unified Process, Cengage Learning, 2005.
3. Erich Gamma, Richard Helm, Ralph Johnson and john Vlissides: Design Patterns – Elements of Reusable Object-Oriented Software,
   Pearson Education,2007.

Reference Books:
1. Grady Booch et. al.: Object-Oriented Analysis and Design with Applications,3rd Edition,Pearson Education,2007.
2. 2.Frank Buschmann, RegineMeunier, Hans Rohnert, Peter Sommerlad, Michel Stal: Pattern –Oriented Software Architecture. A system of patterns , Volume 1, John Wiley and Sons.2007.
3. 3. Booch, Jacobson, Rambaugh : Object-Oriented Analysis and Design with Applications, 3rd edition, pearson, Reprint 2013

INTRODUCTION TO SOFTWARE TESTING [As per Choice Based Credit System (CBCS) scheme] (Effective from the academic year 2016 -2017) SEMESTER – V			
Subject Code	15CS552	IA Marks	20
Number of Lecture Hours/Week	3	Exam Marks	80
Total Number of Lecture Hours	40	Exam Hours	03
CREDITS – 03			

Course objectives: This course will enable students to

- Differentiate the various testing techniques.
- Analyze the problem and derive suitable test cases.
- Apply suitable technique for designing of flow graph.
- Explain the need for planning and monitoring a process.

Module – 1	Teaching Hours
Basics of Software Testing: Basic definitions, Software Quality , Requirements, Behaviour and Correctness, Correctness versus Reliability, Testing and Debugging, Test cases, Insights from a Venn diagram, Identifying test cases, Test-generation Strategies, Test Metrics, Error and fault taxonomies , Levels of testing, Testing and Verification, Static Testing. **Textbook 3: Ch 1:1.2 - 1.5, 3; Textbook 1: Ch 1**	8 Hours
Module – 2	
Problem Statements: Generalized pseudo code, the triangle problem, the NextDate function, the commission problem, the SATM (Simple Automatic Teller Machine) problem, the currency converter, Saturn windshield wiper **Functional Testing:** Boundary value analysis, Robustness testing, Worst-case testing, Robust Worst testing for triangle problem, NextDate problem and commission problem, Equivalence classes, Equivalence test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations, Decision tables, Test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations. **Textbook 1: Ch 2, 5, 6 & 7, Textbook 2: Ch 3**	8 Hours
Module – 3	
Fault Based Testing: Overview, Assumptions in fault based testing, Mutation analysis, Fault-based adequacy criteria, Variations on mutation analysis. **Structural Testing:** Overview, Statement testing, Branch testing, Condition testing, Path testing: DD paths, Test coverage metrics, Basis path testing, guidelines and observations, Data –Flow testing: Definition-Use testing, Slice-based testing, Guidelines and observations**.** **T2:Chapter 16, 12 T1:Chapter 9 & 10**	8 Hours
Module – 4	
Test Execution: Overview of test execution, from test case specification to test cases, Scaffolding, Generic versus specific scaffolding, Test oracles, Self-checks as oracles, Capture and replay **Process Framework :**Basic principles: Sensitivity, redundancy, restriction, partition, visibility, Feedback, the quality process, Planning and monitoring, Quality goals, Dependability properties ,Analysis Testing, Improving the process, Organizational factors. **Planning and Monitoring the Process:** Quality and process, Test and analysis strategies and plans, Risk planning, monitoring the process, Improving the	8 Hours

process, the quality team. **T2: Chapter 17, 20.**	
Module – 5	
Integration and Component-Based Software Testing: Overview, Integration testing strategies, Testing components and assemblies. System, Acceptance and Regression Testing: Overview, System testing, Acceptance testing, Usability, Regression testing, Regression test selection techniques, Test case prioritization and selective execution. **Levels of Testing, Integration Testing:** Traditional view of testing levels, Alternative life-cycle models, The SATM system, Separating integration and system testing, A closer look at the SATM system, Decomposition-based, call graph-based, Path-based integrations. **T2: Chapter 21 & 22, T1 : Chapter 12 & 13**	**8 Hours**

Course outcomes: The students should be able to:

- Derive test cases for any given problem
- Compare the different testing techniques
- Classify the problem into suitable testing model
- Apply the appropriate technique for the design of flow graph.
- Create appropriate document for the software artefact.

Question paper pattern:
The question paper will have TEN questions.
There will be TWO questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer FIVE full questions, selecting ONE full question from each module.

Text Books:

1. Paul C. Jorgensen: Software Testing, A Craftsman's Approach, 3rd Edition, Auerbach Publications, 2008.
2. Mauro Pezze, Michal Young: Software Testing and Analysis – Process, Principles and Techniques, Wiley India, 2009.
3. Aditya P Mathur: Foundations of Software Testing, Pearson Education, 2008.

Reference Books:

1. Software testing Principles and Practices – Gopalaswamy Ramesh, Srinivasan Desikan, 2 nd Edition, Pearson, 2007.
2. Software Testing – Ron Patton, 2nd edition, Pearson Education, 2004.
3. The Craft of Software Testing – Brian Marrick, Pearson Education, 1995.

ADVANCED JAVA AND J2EE			
[As per Choice Based Credit System (CBCS) scheme]			
(Effective from the academic year 2016 -2017)			
SEMESTER – V			
Subject Code	15CS553	IA Marks	20
Number of Lecture Hours/Week	3	Exam Marks	80
Total Number of Lecture Hours	40	Exam Hours	03
CREDITS – 03			

Course objectives: This course will enable students to

- Identify the need for advanced Java concepts like Enumerations and Collections
- Demonstrate the use of JavaBeans to develop component-based Java software
- Adapt servlets to build server side programs
- Make use of JDBC to access database through Java Programs
- Construct client-server applications using Java socket API

Module – 1	Teaching Hours
Enumerations, Autoboxing and Annotations(metadata) Enumerations, Enumeration fundamentals, the values() and valueOf() Methods, java enumerations are class types, enumerations Inherits Enum, example, type wrappers, Autoboxing, Autoboxing and Methods, Autoboxing/Unboxing occurs in Expressions, Autoboxing/Unboxing Boolean and character values , Autoboxing/Unboxing helps prevent errors, A word of Warning. Annotations, Annotation basics, specifying retention policy, Obtaining Annotations at run time by use of reflection, Annotated element Interface, Using Default values, Marker Annotations, Single Member annotations, Built-In annotations.` **Textbook 1: Ch 12**	**8 Hours**
Module – 2	
The collections and Framework Collections Overview, Recent Changes to Collections, The Collection Interfaces, The Collection Classes, Accessing a collection Via an Iterator, Storing User Defined Classes in Collections, The Random Access Interface, Working With Maps, Comparators, The Collection Algorithms, Why Generic Collections?, The legacy Classes and Interfaces, Parting Thoughts on Collections. **Textbook 1: Ch 17**	**8 Hours**
Module – 3	
Networking: Introduction, Networking Basics, Protocols, Internet Address (IPv4 and IPv6),Ports, Classes and Interfaces in java.net, class InetAddress, Constructors of class in Serverside Program InetAddress, Methods of classInet Address, class URL, Constructors of URL class, Methods of URL class, class URL Connection, TCP/IP Server Socket Programming, Constructors of classSocket, Methods of class Socket, Class ServerSocket, Constructors of class ServerSocket, Methods of class ServerSocket, Communication through Sockets, Client-side Socketp programming, Users/UnreliableDatagramProtocol, classDatagramPacket, Constructors, Methods of class Datagram Packets, DatagramSocket Class, Constructors of class DatagramSocket, Methods of class DatagramSocket, Programs for Sending and Receiving Datagram, Creating a Datagram Packet for Sending and Receiving ,Writing Code for Receiving Datagram. **Textbook 2: Ch 12 and 13**	**8 Hours**

Module – 4	
Java Beans: Introduction to Java Beans, Attributes of Beans, Benefits of Using Beans, Properties of a Bean, Java Bean API, Interfaces, Classes of java. bean package(Java SE 8),class Beans, Class Methods, class Property Change Support, JAR Files, Creating a JAR file, Viewing the contents of a JAR File, Building Java Beans with NETBEAN IDE, Building a Composite Component Bean, Changing Properties of Components, Source Code Generated by IDE, Java Beans Project 1—Construct a Bean Containing a Label and a Scrollbar, Java Beans Project 2— Construct a Bean to Illustrate MouseEvents, Java Beans Project 3— Construct a Bean to Illustrate Working of RadioButtons, Java Beans Project 4—Construct a Bean Containing ComboBox, Java Beans Project 5,Application Program, Application Project 1,Application Project 2. **Java Servlets:** Introduction, HTTP Basics, Life Cycle of a Servlet, Servlet Architecture, Setting up Web Server for Servlet Deployment, Classes Defined in Javax.servlet.http Package, Interfaces Defined in javax.servlet.http Package, Handling HTTP Request and Response, Writing Servlet, Reading Servlet Parameters, Classes in javax.servlet Package, Session Tracking and Session Management, Session Tracking Techniques, Http Redirects in Servlets, Securing Servlets, Application Program.	**8 Hours**
Textbook 2: Ch 26, 20	
Module – 5	
Java Database Connectivity: Introduction, JDBC Architecture, Two-tier Architecture for DataAccess, Three-tier Architecture for DataAccess, Installing MySQL and MySQL Connector/J,SQL Statements, JDBC Environment Setup, JDBC Connectivity Model and API, Establishing JDBC Database Connections, Load and Register the JDBC Driver ,Defining the ConnectionURL, ResultSet Interface, Navigating the ResultSet , ResultSetMetaData interface, Creating JDBC Application, JDBC Batch Processing, JDBC Transaction Management, Application Programs.	**8 Hours**
Textbook 2: Ch 22	

Course outcomes: The students should be able to:

- Interpret the need for advanced Java concepts like enumerations and collections in developing modular and efficient programs
- Build client-server applications and TCP/IP socket programs
- Develop reusable software components using Java Beans
- Describe how servlets fit into Java-based web application architecture
- Illustrate database access and details for managing information using the JDBC API

Question paper pattern:
The question paper will have TEN questions.
There will be TWO questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer FIVE full questions, selecting ONE full question from each module.

Text Books:
1. Herbert Schildt: Java The Complete Reference, 7th Edition, Tata McGraw Hill, 2007.
2. Uttam K. Roy , Advanced JAVA Programming. Oxford University Press.

Reference Books:
1. Jim Keogh: J2EE - The Complete Reference, Tata McGraw Hill, 2007.
2. Y. Daniel Liang: Introduction to JAVA Programming, 7th Edition, Pearson

Education, 2007.

3. Stephanie Bodoff et al: The J2EE Tutorial, 2nd Edition, Pearson Education, 2004.

ADVANCED ALGORITHMS
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2016 -2017)
SEMESTER – V

Subject Code	15CS554	IA Marks	20
Number of Lecture Hours/Week	3	Exam Marks	80
Total Number of Lecture Hours	40	Exam Hours	03

CREDITS – 03

Course objectives: This course will enable students to

- Explain principles of algorithms analysis approaches
- Compare and contrast a number theoretic based strategies.
- Describe complex signals and data flow in networks
- Apply the computational geometry criteria.

Module – 1	Teaching Hours
Analysis Techniques: Growth functions, Recurrences and solution of recurrence equations; Amortized analysis: Aggregate, Accounting, and Potential methods, String Matching Algorithms: Naive Algorithm; Robin-Karp Algorithm, String matching with Finite Automata, Knuth-Morris-Pratt and Boyer-Moore Algorithms	**8 Hours**
Module – 2	
Number Theoretic Algorithms: Elementary notions, GCD, Modular arithmetic, Solving modular linear equations, The Chinese remainder theorem, Powers of an element RSA Cryptosystem, Primality testing, Integer factorization, - Huffman Codes, Polynomials. FFT-Huffman codes: Concepts, construction, Proof correctness of Huffman's algorithm; Representation of polynomials	**8 Hours**
Module – 3	
DFT and FFT efficient implementation of FFT, Graph Algorithms, Bellman-Ford Algorithm Shortest paths in a DAG, Johnson's Algorithm for sparse graphs, Flow networks and the Ford-Fulkerson Algorithm, Maximum bipartite matching.	**8 Hours**
Module – 4	
Computational Geometry-I: Geometric data structures using, C, Vectors, Points, Polygons, Edges Geometric objects in space; Finding the intersection of a line and a triangle, Finding star-shaped polygons and convex hulls using,, incremental insertion, Point Enclosure: Ray-shooting and Signed angle methods.	**8 Hours**
Module – 5	
Computational Geometry-II: Clipping: Cyrus-Beck and Sutherland-Hodman Algorithms; Triangulating, monotonic polygons; Convex hulls, Gift wrapping and Graham Scan; Removing hidden surfaces, Intersection of convex polygons; convex hulls, contour of the union of rectangles. Decomposing polygons into monotone pieces.	**8 Hours**

Course outcomes: The students should be able to:

- Explain the principles of algorithms analysis approaches
- Apply different theoretic based strategies to solve problems
- Illustrate the complex signals and data flow in networks with usage of tools
- Describe the computational geometry criteria.

Question paper pattern:
The question paper will have TEN questions.

There will be TWO questions from each module.

Each question will have questions covering all the topics under a module.

The students will have to answer FIVE full questions, selecting ONE full question from each module.

Text Books:

1. Thomas H. Cormen et al: Introduction to Algorithms, Prentice Hall India, 1990
2. Michael J. Laszlo: Computational Geometry and Computer Graphics in C' Prentice Hall India, 1996

Reference Books:

1. E. Horowitz, S. Sahni and S. Rajasekaran, Fundamentals of Computer Algorithms, University Press, Second edition, 2007
2. Kenneth A Berman & Jerome L Paul, Algorithms, Cengage Learning, First Indian reprint, 2008

MOBILE APPLICATION DEVELOPMENT
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2016 -2017)
SEMESTER – V

Subject Code	15CS561	IA Marks	20
Number of Lecture Hours/Week	3	Exam Marks	80
Total Number of Lecture Hours	40	Exam Hours	03

CREDITS – 03

Course objectives: This course will enable students to

- Learn to setup Android application development environment
- Illustrate user interfaces for interacting with apps and triggering actions
- Interpret tasks used in handling multiple activities
- Identify options to save persistent application data
- Appraise the role of security and performance in Android applications

Module – 1	Teaching Hours
Get started, Build your first app, Activities, Testing, debugging and using support libraries	8 Hours
Module – 2	
User Interaction, Delightful user experience, Testing your UI	8 Hours
Module – 3	
Background Tasks, Triggering, scheduling and optimizing background tasks	8 Hours
Module – 4	
All about data, Preferences and Settings, Storing data using SQLite, Sharing data with content providers, Loading data using Loaders	8 Hours
Module – 5	
Permissions, Performance and Security, Firebase and AdMob, Publish	8 Hours

Course outcomes: The students should be able to:

- Create, test and debug Android application by setting up Android development environment
- Implement adaptive, responsive user interfaces that work across a wide range of devices.
- Infer long running tasks and background work in Android applications
- Demonstrate methods in storing, sharing and retrieving data in Android applications
- Analyze performance of android applications and understand the role of permissions and security
- Describe the steps involved in publishing Android application to share with the world

Question paper pattern:

The question paper will have TEN questions.

There will be TWO questions from each module.

Each question will have questions covering all the topics under a module.

The students will have to answer FIVE full questions, selecting ONE full question from each module.

Text Books:

1. Google Developer Training, "Android Developer Fundamentals Course – Concept Reference", Google Developer Training Team, 2017.
   https://www.gitbook.com/book/google-developer-training/android-developer-fundamentals-course-concepts/details (Download pdf file from the above link)

Reference Books:

1. Erik Hellman, "Android Programming – Pushing the Limits", 1st Edition, Wiley India Pvt Ltd, 2014.
2. Dawn Griffiths and David Griffiths, "Head First Android Development", 1st Edition, O'Reilly SPD Publishers, 2015.
3. J F DiMarzio, "Beginning Android Programming with Android Studio", 4th Edition, Wiley India Pvt Ltd, 2016. ISBN-13: 978-8126565580
4. Anubhav Pradhan, Anil V Deshpande, " Composing Mobile Apps" using Android, Wiley 2014, ISBN: 978-81-265-4660-2

ARTIFICIAL INTELLIGENCE
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2016 -2017)
SEMESTER – V

Subject Code	15CS562	IA Marks	20
Number of Lecture Hours/Week	3	Exam Marks	80
Total Number of Lecture Hours	40	Exam Hours	03

CREDITS – 03

Course objectives: This course will enable students to

- Identify the problems where AI is required and the different methods available
- Compare and contrast different AI techniques available.
- Define and explain learning algorithms

Module – 1	Teaching Hours
Overview of Artificial Intelligence: Definition & Importance of AI., Problems, Problem Spaces and search, **Knowledge Representation Reasoning, Issues and Acquisition.:** Predicate Logic, Representing knowledge using Rules, Symbolic Reasoning under Uncertainty, Statistical reasoning, Weak Slot and Filter Structures	8 Hours
Module – 2	
Heuristic Search Techniques. Defining the problem as a state space search, production system, problem Characteristics, Heuristic search techniques-Generate and test, Hill Climbing, Best-First Search, Problem Reduction, Constraint Satisfaction and Means-ends Analysis, State of Art Game programs.	8 Hours
Module – 3	
Game Playing: Minimax search procedure, Adding alpha-beta cutoffs, additional refinement. Iterative Deepening and references on specific games **Natural Language Processing:** Syntactic Processing, Semantic Analysis, Discourse and Pragmatic processing, Statistical Natural language processing and Spell checking.	8 Hours
Module – 4	
Learning: What is learning?, Forms of learning, Rote learning, learning by taking advice, Learning in problem solving, Induction leaning, Explanation based learning, Discovery, Analogy, Formal learning Theory, Neural Network Learning and Genetic Learning	8 Hours
Module – 5	
Expert Systems: Basic Concepts of Expert Systems, Application of Expert Systems, Structure of Expert Systems, Knowledge Engineering, Problem Area suitable for Expert Systems, Developments ,Benefits of Expert Systems and Expert Systems on Web. **T2: Chapter 12**	8 Hours

Course outcomes: The students should be able to:

- Identify the AI based problems
- Apply techniques to solve the AI problems
- Define learning and explain various learning techniques
- Discuss on expert systems

Question paper pattern:

The question paper will have TEN questions.
There will be TWO questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer FIVE full questions, selecting ONE full question from each module.

Text Books:

1. E. Rich , K. Knight & S. B. Nair - Artificial Intelligence, 3/e, McGraw Hill. **Chapters 1,2,3,4,5.5.7,8,9,10,12,15,17**
2. Decision Support and Business Intelligence Systems E Turban, R Sharda and D. Delen Pearson 9-Edition **Chapter 12**

Reference Books:

1. Artificial Intelligence: A Modern Approach, Stuart Rusell, Peter Norving, Pearson Education 2nd Edition.
1. Dan W. Patterson, Introduction to Artificial Intelligence and Expert Systems – Prentice Hal of India.
2. G. Luger, "Artificial Intelligence: Structures and Strategies for complex problem Solving", Fourth Edition, Pearson Education, 2002.
3. Artificial Intelligence and Expert Systems Development by D W Rolston-Mc Graw hill.
4. N.P. Padhy "Artificial Intelligence and Intelligent Systems" , Oxford University Press-2015

EMBEDDED COMPUTING SYSTEM				
[As per Choice Based Credit System (CBCS) scheme]				
(Effective from the academic year 2016 -2017)				
SEMESTER – V				
Subject Code	15CS563	IA Marks	20	
Number of Lecture Hours/Week	3	Exam Marks	80	
Total Number of Lecture Hours	40	Exam Hours	03	
CREDITS – 03				

Course objectives: This course will enable students to

- Provide a general overview of Embedded Systems
- Show current statistics of Embedded Systems
- Design, code, compile, and test real-time software
- Integrate a fully functional system including hardware and software.

Module – 1	**Teaching Hours**
Introduction to embedded systems: Embedded systems, Processor embedded into a system, Embedded hardware units and device in a system, Embedded software in a system, Examples of embedded systems, Design process in embedded system, Formalization of system design, Design process and design examples, Classification of embedded systems, skills required for an embedded system designer.	**8 Hours**
Module – 2	
Devices and communication buses for devices network: IO types and example, Serial communication devices, Parallel device ports, Sophisticated interfacing features in device ports, Wireless devices, Timer and counting devices, Watchdog timer, Real time clock, Networked embedded systems, Serial bus communication protocols, Parallel bus device protocols-parallel communication internet using ISA, PCI, PCI-X and advanced buses, Internet enabled systems-network protocols, Wireless and mobile system protocols.	**8 Hours**
Module – 3	
Device drivers and interrupts and service mechanism: Programming-I/O busy-wait approach without interrupt service mechanism, ISR concept, Interrupt sources, Interrupt servicing (Handling) Mechanism, Multiple interrupts, Context and the periods for context switching, interrupt latency and deadline, Classification of processors interrupt service mechanism from Context-saving angle, Direct memory access, Device driver programming.	**8 Hours**
Module – 4	
Inter process communication and synchronization of processes, Threads and tasks: Multiple process in an application, Multiple threads in an application, Tasks, Task states, Task and Data, Clear-cut distinction between functions. ISRS and tasks by their characteristics, concept and semaphores, Shared data, Inter-process communication, Signal function, Semaphore functions, Message Queue functions, Mailbox functions, Pipe functions, Socket functions, RPC functions.	**8 Hours**
Module – 5	
Real-time operating systems: OS Services, Process management, Timer functions, Event functions, Memory management, Device, file and IO subsystems management, Interrupt routines in RTOS environment and handling of interrupt source calls, Real-time operating systems, Basic design using an RTOS, RTOS task scheduling models, interrupt latency and response of the tasks	**8 Hours**

as performance metrics, OS security issues. Introduction to embedded software development process and tools, Host and target machines, Linking and location software.

Course outcomes: The students should be able to:
• Distinguish the characteristics of embedded computer systems.
• Examine the various vulnerabilities of embedded computer systems.
• Design and develop modules using RTOS.
• Implement RPC, threads and tasks

Question paper pattern:

The question paper will have TEN questions.

There will be TWO questions from each module.

Each question will have questions covering all the topics under a module.

The students will have to answer FIVE full questions, selecting ONE full question from each module.

Text Books:

1. Raj Kamal, "Embedded Systems: Architecture, Programming, and Design" 2nd edition , Tata McGraw hill-2013. **Chapter 1.1 to 1.5, 1.8 to 1.12, Chapter 3, 4, 7, 8 and 13.1 to 13.3.**

Reference Books:

1. Marilyn Wolf, "Computer as Components, Principles of Embedded Computing System Design" 3rd edition, Elsevier-2014.

DOT NET FRAMEWORK FOR APPLICATION DEVELOPMENT
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2016 -2017)
SEMESTER – V

Subject Code	15CS564	IA Marks	20
Number of Lecture Hours/Week	3	Exam Marks	80
Total Number of Lecture Hours	40	Exam Hours	03

CREDITS – 03

Course objectives: This course will enable students to

- Inspect Visual Studio programming environment and toolset designed to build applications for Microsoft Windows
- Understand Object Oriented Programming concepts in C# programming language.
- Interpret Interfaces and define custom interfaces for application.
- Build custom collections and generics in C#
- Construct events and query data using query expressions

Module – 1	Teaching Hours
Introducing Microsoft Visual C# and Microsoft Visual Studio 2015: Welcome to C#, Working with variables, operators and expressions, Writing methods and applying scope, Using decision statements, Using compound assignment and iteration statements, Managing errors and exceptions **T1: Chapter 1 – Chapter 6**	8 Hours
Module – 2	
Understanding the C# object model: Creating and Managing classes and objects, Understanding values and references, Creating value types with enumerations and structures, Using arrays **Textbook 1: Ch 7 to 10**	8 Hours
Module – 3	
Understanding parameter arrays, Working with inheritance, Creating interfaces and defining abstract classes, Using garbage collection and resource management **Textbook 1: Ch 11 to 14**	8 Hours
Module – 4	
Defining Extensible Types with C#: Implementing properties to access fields, Using indexers, Introducing generics, Using collections **Textbook 1: Ch 15 to 18**	8 Hours
Module – 5	
Enumerating Collections, Decoupling application logic and handling events, Querying in-memory data by using query expressions, Operator overloading **Textbook 1: Ch 19 to 22**	8 Hours

Course outcomes: The students should be able to:

- Build applications on Visual Studio .NET platform by understanding the syntax and semantics of C#
- Demonstrate Object Oriented Programming concepts in C# programming language
- Design custom interfaces for applications and leverage the available built-in interfaces in building complex applications.
- Illustrate the use of generics and collections in C#
- Compose queries to query in-memory data and define own operator behaviour

Question paper pattern:

The question paper will have TEN questions.

There will be TWO questions from each module.

Each question will have questions covering all the topics under a module.

The students will have to answer FIVE full questions, selecting ONE full question from each module.

Text Books:

1. John Sharp, Microsoft Visual C# Step by Step, 8th Edition, PHI Learning Pvt. Ltd. 2016

Reference Books:

1. Christian Nagel, "C# 6 and .NET Core 1.0", 1st Edition, Wiley India Pvt Ltd, 2016. Andrew Stellman and Jennifer Greene, "Head First C#", 3rd Edition, O'Reilly Publications, 2013.
2. Mark Michaelis, "Essential C# 6.0", 5th Edition, Pearson Education India, 2016.
3. Andrew Troelsen, "Prof C# 5.0 and the .NET 4.5 Framework", 6th Edition, Apress and Dreamtech Press, 2012.

COMPUTER NETWORK LABORATORY			
[As per Choice Based Credit System (CBCS) scheme]			
(Effective from the academic year 2016 -2017)			
SEMESTER – V			
Subject Code	15CSL57	IA Marks	20
Number of Lecture Hours/Week	01I + 02P	Exam Marks	80
Total Number of Lecture Hours	40	Exam Hours	03
CREDITS – 02			
Course objectives: This course will enable students to			
• Demonstrate operation of network and its management commands • Simulate and demonstrate the performance of GSM and CDMA • Implement data link layer and transport layer protocols.			
Description (If any):			
For the experiments below modify the topology and parameters set for the experiment and take multiple rounds of reading and analyze the results available in log files. Plot necessary graphs and conclude using any **suitable** tool.			
Lab Experiments:			
PART A			
1. Implement three nodes point – to – point network with duplex links between them. Set the queue size, vary the bandwidth and find the number of packets dropped. 2. Implement transmission of ping messages/trace route over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion. 3. Implement an Ethernet LAN using n nodes and set multiple traffic nodes and plot congestion window for different source / destination. 4. Implement simple ESS and with transmitting nodes in wire-less LAN by simulation and determine the performance with respect to transmission of packets. 5. Implement and study the performance of GSM on NS2/NS3 (Using MAC layer) or equivalent environment. 6. Implement and study the performance of CDMA on NS2/NS3 (Using stack called Call net) or equivalent environment.			
PART B			
Implement the following in Java: 7. Write a program for error detecting code using CRC-CCITT (16- bits). 8. Write a program to find the shortest path between vertices using bellman-ford algorithm. 9. Using TCP/IP sockets, write a client – server program to make the client send the file name and to make the server send back the contents of the requested file if present. Implement the above program using as message queues or FIFOs as IPC channels. 10. Write a program on datagram socket for client/server to display the messages on client side, typed at the server side. 11. Write a program for simple RSA algorithm to encrypt and decrypt the data. 12. Write a program for congestion control using leaky bucket algorithm.			
Study Experiment / Project:			
NIL			
Course outcomes: The students should be able to:			
• Analyze and Compare various networking protocols.			

- Demonstrate the working of different concepts of networking.
- Implement, analyze and evaluate networking protocols in NS2 / NS3

Conduction of Practical Examination:

1. All laboratory experiments are to be included for practical examination.
2. Students are allowed to pick one experiment from part A and part B with lot.
3. Strictly follow the instructions as printed on the cover page of answer script
4. Marks distribution: Procedure + Conduction + Viva: 80

                                               Part A: 5+30+5   =40

                                               Part B: 5+30+5   =40

5. Change of experiment is allowed only once and marks allotted to the procedure part to be made zero.

<div align="center">

DBMS LABORATORY WITH MINI PROJECT
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2016 -2017)
SEMESTER – V

</div>

Subject Code	15CSL58	IA Marks	20
Number of Lecture Hours/Week	01I + 02P	Exam Marks	80
Total Number of Lecture Hours	40	Exam Hours	03

<div align="center">

CREDITS – 02

</div>

Course objectives: This course will enable students to

- Foundation knowledge in database concepts, technology and practice to groom students into well-informed database application developers.
- Strong practice in SQL programming through a variety of database problems.
- Develop database applications using front-end tools and back-end DBMS.

Description (If any):

 PART-A: SQL Programming (Max. Exam Mks. 50)
- Design, develop, and implement the specified queries for the following problems using Oracle, MySQL, MS SQL Server, or any other DBMS under LINUX/Windows environment.
- Create Schema and insert at least 5 records for each table. Add appropriate database constraints.

 PART-B: Mini Project (Max. Exam Mks. 30)
- Use Java, C#, PHP, Python, or any other similar front-end tool. All applications must be demonstrated on desktop/laptop as a stand-alone or web based application (Mobile apps on Android/IOS are not permitted.)

Lab Experiments:

Part A: SQL Programming

1	Consider the following schema for a Library Database:
	BOOK(Book_id, Title, Publisher_Name, Pub_Year)
	BOOK_AUTHORS(Book_id, Author_Name)
	PUBLISHER(Name, Address, Phone)
	BOOK_COPIES(Book_id, Branch_id, No-of_Copies)
	BOOK_LENDING(Book_id, Branch_id, Card_No, Date_Out, Due_Date)
	LIBRARY_BRANCH(Branch_id, Branch_Name, Address)
	Write SQL queries to
	    1. Retrieve details of all books in the library – id, title, name of publisher, authors, number of copies in each branch, etc.
	    2. Get the particulars of borrowers who have borrowed more than 3 books, but from Jan 2017 to Jun 2017.
	    3. Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulation operation.
	    4. Partition the BOOK table based on year of publication. Demonstrate its working with a simple query.
	    5. Create a view of all books and its number of copies that are currently available in the Library.

2	Consider the following schema for Order Database: SALESMAN(Salesman_id, Name, City, Commission) CUSTOMER(Customer_id, Cust_Name, City, Grade, Salesman_id) ORDERS(Ord_No, Purchase_Amt, Ord_Date, Customer_id, Salesman_id) Write SQL queries to 1. Count the customers with grades above Bangalore's average.

	2. Find the name and numbers of all salesman who had more than one customer. 3. List all the salesman and indicate those who have and don't have customers in their cities (Use UNION operation.) 4. Create a view that finds the salesman who has the customer with the highest order of a day. 5. Demonstrate the DELETE operation by removing salesman with id 1000. All his orders must also be deleted.

3	Consider the schema for Movie Database:
	ACTOR(Act_id, Act_Name, Act_Gender)
	DIRECTOR(Dir_id, Dir_Name, Dir_Phone)
	MOVIES(Mov_id, Mov_Title, Mov_Year, Mov_Lang, Dir_id)
	MOVIE_CAST(Act_id, Mov_id, Role)
	RATING(Mov_id, Rev_Stars)
	Write SQL queries to
	    1. List the titles of all movies directed by 'Hitchcock'.
	    2. Find the movie names where one or more actors acted in two or more movies.
	    3. List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOIN operation).
	    4. Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by movie title.
	    5. Update rating of all movies directed by 'Steven Spielberg' to 5.

4	Consider the schema for College Database:
	STUDENT(USN, SName, Address, Phone, Gender)
	SEMSEC(SSID, Sem, Sec)
	CLASS(USN, SSID)
	SUBJECT(Subcode, Title, Sem, Credits)
	IAMARKS(USN, Subcode, SSID, Test1, Test2, Test3, FinalIA)
	Write SQL queries to
	    1. List all the student details studying in fourth semester 'C' section.
	    2. Compute the total number of male and female students in each semester and in each section.
	    3. Create a view of Test1 marks of student USN '1BI15CS101' in all subjects.
	    4. Calculate the FinalIA (average of best two test marks) and update the corresponding table for all students.
	    5. Categorize students based on the following criterion:
	    If FinalIA = 17 to 20 then CAT = 'Outstanding'
	    If FinalIA = 12 to 16 then CAT = 'Average'
	    If FinalIA< 12 then CAT = 'Weak'
	    Give these details only for 8th semester A, B, and C section students.

5	Consider the schema for Company Database: EMPLOYEE(SSN, Name, Address, Sex, Salary, SuperSSN, DNo) DEPARTMENT(DNo, DName, MgrSSN, MgrStartDate) DLOCATION(DNo,DLoc) PROJECT(PNo, PName, PLocation, DNo) WORKS_ON(SSN, PNo, Hours) Write SQL queries to 1. Make a list of all project numbers for projects that involve an employee whose last name is 'Scott', either as a worker or as a manager of the department that controls the project.

	2.	Show the resulting salaries if every employee working on the 'IoT' project is given a 10 percent raise.
	3.	Find the sum of the salaries of all employees of the 'Accounts' department, as well as the maximum salary, the minimum salary, and the average salary in this department
	4.	Retrieve the name of each employee who works on all the projects controlledby department number 5 (use NOT EXISTS operator).
	5.	For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than Rs. 6,00,000.

Part B: Mini project

- For any problem selected, write the ER Diagram, apply ER-mapping rules, normalize the relations, and follow the application development process.
- Make sure that the application should have five or more tables, at least one trigger and one stored procedure, using suitable frontend tool.
- Indicative areas include; health care, education, industry, transport, supply chain, etc.

Course outcomes: The students should be able to:

- Create, Update and query on the database.
- Demonstrate the working of different concepts of DBMS
- Implement, analyze and evaluate the project developed for an application.

Conduction of Practical Examination:

1. All laboratory experiments from part A are to be included for practical examination.
2. Mini project has to be evaluated for 30 Marks as per 6(b).
3. Report should be prepared in a standard format prescribed for project work.
4. Students are allowed to pick one experiment from the lot.
5. Strictly follow the instructions as printed on the cover page of answer script.
6. Marks distribution:
   a) Part A: Procedure + Conduction + Viva:10 + 35 +5 =50 Marks
   b) Part B: Demonstration + Report + Viva voce = 15+10+05 = 30 Marks
7. Change of experiment is allowed only once and marks allotted to the procedure part to be made zero.