JECRC™
UNIVERSITY
BUILD YOUR WORLD

School of Engineering

Syllabi and Course Structure

B. Tech. (Computer Science & Engineering) (2019-2023)

Academic Programmes

April 2019

The curriculum and syllabus for B.Tech. Program conforms to outcome based teaching learning process. In general, several outcomes have been identified and the curriculum and syllabus have been planned in such a way that each of the courses meets one or more of these outcomes. Student outcomes illustrate the students are expected to know and be able to do by the time of graduation. These relate to the skills, understanding, and behaviours that students acquire as they progress through the program. Further each course in the program brings out clear instructional objectives which are mapped to the student outcomes.

B.Tech. (CSE) Program Educational Objective (PEO's):
A graduate of the Computer Science and Engineering Program should:

PEO- I

Students will develop themselves as effective professionals by solving real problems through the use of computer science knowledge and with attention to team work, effective communication, critical thinking and problem solving skills.

PEO- II

Students will develop professional skills that prepare them for immediate employment and for life-long learning in advanced areas of computer science and related fields.

PEO- III

Students will demonstrate their ability to adapt to a rapidly changing environment by having learned and applied new skills and new technologies.

PEO- IV

Students will be provided with an educational foundation that prepares them for excellence, leadership roles along diverse career paths with encouragement to professional ethics and active participation needed for a successful career.

Program Outcome (PO's)

A graduate of the Computer Science and Engineering Program will demonstrate:

PO1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2. Problem analysis: Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4. Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11. Project management and finance: Demonstrate knowledge and understanding of the

engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcome:

PSO1: The ability to understand, analyze and develop computer programs in the areas related to algorithms, system software, multimedia, web design, big data analytics, and networking for efficient design of computer-based systems of varying complexity.(Professional Skills)

PSO2: The ability to apply standard practices and strategies in software project development using open-ended programming environments to deliver a quality product for business success. (Problem-Solving Skills)

PSO3: The ability to employ modern computer languages, environments, and platforms in creating innovative career paths to be an entrepreneur, and a zest for higher studies.(Successful Career and Entrepreneurship)

B.Tech CSE III Semester

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	Type
1	BAS 003A	Multivariate Analysis, Linear Algebra and Special Functions	3	0	0	3	3	F
2	BAS 007A	Discrete Mathematics	2	0	0	2	2	F
3	BCO 001B	Software Engineering	3	0	0	3	3	S
4	BCO 002A/ BCO 002B	Data Structures and Algorithms	3	0	0	3	3	C
5	BCO 003B	Object Oriented Programming with C++	3	0	0	3	3	C
6	BEE009A	Digital Systems	4	0	0	4	4	F
7	BCO 004A	Object Oriented Programming Lab	0	0	2	2	2	C
8	BCO 005A	Data Structure and Algorithms Lab	0	0	2	2	2	C
9	BEE010A	Digital Systems Lab	0	0	2	2	2	F
10	BCO 080B	Linux Programming Lab	0	0	2	2	3	S
11	BCO 011A	Computer Networks	4	0	0	4	4	C
		Total	**22**	**0**	**8**	**30**	**30**	

B.Tech CSE Semester IV

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	TYPE
1	BAS 005A	Complex Analysis	2	0	0	2	2	F
2	BCO 007A	Computer Graphics	3	0	0	3	3	C
3	BCO 008B	Operating Systems	3	0	0	3	3	C
4	BCO 009B	Computer Organization and Design	3	1	0	4	4	C
5	BCO 010C	Database Management Systems	4	0	0	4	4	C
6	BCO 081B	Programming with Python	2	0	0	2	2	S
7	BCO 012A	Software Project Management	3	0	0	3	3	S
8	BCO 013A	Database Management Systems Lab	0	0	2	2	2	C
9	BCO 014A	Operating Systems (Unix Programming) Lab	0	0	2	2	2	C
10	BCO 082A	Programming with Python Lab	0	0	2	2	2	S
11	BCO 015A	Computer Graphics Lab	0	0	2	2	2	C
12	BCO 016A	Seminar	0	0	1	1	1	S
		Total	20	1	9	30	30	

B.Tech. CSE Semester V

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	Type
1	BAS 004A	Optimization and Calculus of Variations	2	0	0	2	2	S
2	BCO 017A	Formal Languages & Automation Theory	3	1	0	4	4	C
3	BCO 018A	Object Oriented Analysis and Design	2	0	0	2	2	S
4	BCO 019A	Artificial Intelligence	4	0	0	4	4	S
5		Program Elective – I	4	0	0	4	4	S
6		Program Elective II	4	0	0	4	4	S
7		Open Elective-I	3	0	0	3	3	ID
8		Program Elective –I Lab	0	0	2	2	2	S
9		Program Elective –II Lab	0	0	2	2	2	S
10	BCO 021A	Object Oriented Analysis and Design Lab	0	0	2	2	2	S
11	BCO 022A	Seminar	0	0	1	1	1	S
		Total	**22**	**1**	**7**	**30**	**30**	

B.Tech CSE Semester VI

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	TYPE
1	BCO 090A	Internet of Things	2	0	0	2	2	S
2	BCO 023A	Design & Analysis of Algorithms	3	0	0	3	3	C
3	BCO 024A	Advanced Computer Architecture	3	0	0	3	3	S
4	BEE 047B	Embedded Computing System	3	0	0	3	3	S
5		Program Elective-III	4	0	0	4	4	S
6		Program Elective –IV	4	0	0	4	4	S
7		Open Elective-II	3	0	0	3	3	ID
8	BCO 025A	Design & Analysis of Algorithms Lab	0	0	2	2	2	C
9		Program Elective –III Lab	0	0	2	2	2	S
10	BEE 048A	Embedded Computing Lab	0	0	2	2	2	S
11	BCO 026A	Seminar	0	0	1	1	1	S
12	BCO 074A	Minor Project	0	0	1	0	1	C
		Total	**22**	**0**	**8**	**30**	**30**	

B.Tech. CSE Semester VII

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	Type
1	BCO 027A	Mobile Computing	3	0	0	3	3	S
2	BCO 028A	Compiler Construction	3	0	0	3	3	C
3	BCO 029A	Data Mining & Warehousing	3	0	0	3	3	S
4	BCO 030A	Principles of Information System Security	3	0	0	3	3	S
5		Program Elective-V	4	0	0	4	4	S
6		Program Elective-VI	4	0	0	4	4	S
7		Open Elective-III	3	0	0	3	3	ID
8		Program Elective –V lab	0	0	2	2	2	S
9	BCO 031A	Compiler Design Lab	0	0	2	2	2	C
10	BCO 032A	Project	0	0	2	2	2	C
11	BCO 033A	Seminar	0	0	1	1	1	S
		Total	**23**	**0**	**7**	**30**	**30**	

B.Tech. CSE Semester VIII

S. No.	Code	Subject	L	T	P	Contact Hrs.	Credits	Type
1	BCO 034A	Industrial Project/Dissertation	0	0	28	28	28	C
		TOTAL	**0**	**0**	**28**	**28**	**28**	

Program Elective-I

Prerequisite: Nil

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	Type
V Semester								
1	BCO 035A	**Programming in Java**	4	0	0	4	4	S
2	BCO 036A	**Programming in .NET**	4	0	0	4	4	S

1	**BCO 068A**	**Programming in Java Lab**	**0-0-2**

List of Program Electives-II

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	Type
V Semester								
1	BCO 041A	**Multimedia Computing**	4	0	0	4	4	S
2	BCO 042A	**Information Retrieval**	4	0	0	4	4	S
3	BCO 043A	**Software Architecture**	4	0	0	4	4	S
4	BCO 044A	**High Speed Networks**	4	0	0	4	4	S
5	BCO 045A	**Operations Research**	4	0	0	4	4	S
6	BCO 046A	**Simulation and Modelling**	4	0	0	4	4	S
7.	**BCO 092A**	**Robotic Process Automation**	4	0	0	4	4	S
8	**MOOC 003**	**Animations**	0	0	0	0	4	MOOC

1	**BCO 083A**	**Web Technology Lab**	**0-0-2**
2	**BCO 093 A**	**Robotic Process Automation Lab**	**0-0-2**

List of Program Electives-III

Prerequisite: Students have to take Program Elective-I in respective area.

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	Type
1	BCO 037A	**Advance Programming in Java**	4	0	0	4	4	S
2	BCO 038A	**Advance Programming in .NET**	4	0	0	4	4	S

1	**BCO 069A**	**Advance Programming in Java Lab**	**0-0-2**

List of Program Electives-IV

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	Type
1	BCO 047A	**Graph Theory**	4	0	0	4	4	S
2	BCO 048A	**Real Time Systems**	4	0	0	4	4	S
3	BCO 049A	**Distributed Computing**	4	0	0	4	4	S
4	BCO 050A	**Human Computer Interface**	4	0	0	4	4	S
5	BCO 051A	**Approximation of Algorithms**	4	0	0	4	4	S
6	BCO 052A	**Wireless AdHoc Networks**	4	0	0	4	4	S
7	BCO 070 B	**Big Data Analytics**	4	0	0	4	4	S
8	BCO 091 A	**DevOps**	4	0	0	4	4	S
9	MOOC 037	**Society and Media**	0	0	0	0	4	MOOC

List of Program Electives-V

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	Type
1	BCO 084A	**Cyber Security**	4	0	0	4	4	S
2	BCO 085A	**Digital Image Processing**	4	0	0	4	4	S
3	BCO 086A	**Machine Learning**	4	0	0	4	4	S

Program Elective V Lab

Prerequisite: Students have to take Program Elective-V in respective area.

1	BCO 087A	**Cyber Security Lab**	**0-0-2**
2	BCO 088A	**Digital Image Processing Lab**	**0-0-2**
3	BCO 089A	**Machine Learning Lab**	**0-0-2**

List of Program Electives-VI

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	Type
1	BCO 053A	**Pattern Recognition**	4	0	0	4	4	S
2	BCO 054A	**Soft Computing**	4	0	0	4	4	S
3	BCO 055A	**Game Theory**	4	0	0	4	4	S
4	BCO 056A	**Wireless Sensor Networks**	4	0	0	4	4	S
5	BCO 057A	**Bioinformatics**	4	0	0	4	4	S
6	BCO 058A	**Computer Vision**	4	0	0	4	4	S

List of Open Electives

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	Type
Open Elective-I								
1	BCO 010A	**Database Management System**	4	0	0	4	4	S
2	BCO 059A	**Data Mining**	4	0	0	4	4	S
3	BCO 060A	**Object Oriented Databases**	4	0	0	4	4	S
4	BCO 003A	**Object Oriented Programming With C++**	4	0	0	4	4	S
5	BCO 061A	**Web Designing Techniques**	4	0	0	4	4	S
Open Elective-II								
1	BCO 012A	**Software Project Management**	4	0	0	4	4	S
2	BCO 062A	**Software Testing**	4	0	0	4	4	S
3	BCO 063A	**Advanced Topics in Database Management Systems**	4	0	0	4	4	S
4	BCO 035A	**Programming in Java**	4	0	0	4	4	S
Open Elective-III								
1	BCO 001A	**Software Engineering**	4	0	0	4	4	S
2	BCO 041A	**Multimedia Computing**	4	0	0	4	4	S
3	BCO 042A	**Information Retrieval**	4	0	0	4	4	S
4	BCO 064A	**Cloud Computing**	4	0	0	4	4	S
5	BCO 065A	**Engineering System Analysis and Design**	4	0	0	4	4	S

BAS 003A	MULTIVARIATE ANALYSIS, LINEAR ALGEBRA AND SPECIAL FUNCTIONS	3-0-0 [3]

B.Tech CSE Semester III

Objectives:

1. The objective of this course is to familiarize the prospective engineers with techniques in multivariate analysis, linear algebra and some useful special functions.
2. It deals with acquainting the students with standard concepts and tools at an intermediate to advanced level that will serve them well towards tackling more advanced level of mathematics and applications that they would find useful in their profession.
3. This course introduces vector calculus and its applications, in both differential and integral forms.

UNIT 1	*Multivariate functions* covering, limits, continuity and differentials, partial derivatives, maximum-minimum problems, Laangians, Chain rule.
UNIT 2	Double integrals, iterated integrals, triple integrals, line integrals, simple connected regions, Green's theorem; Path independence, surface integrals.
UNIT 3	Stokes theorem; Fourier series and integral, Dirichlet conditions, Parseval's identity. The convolution theorem.
UNIT 4	*Vectors* covering, laws of vector algebra, operations- dot, cross, triple products;Vector function – limits, continuity and derivatives, geometric interpretation; Gradient,divergence and curl – formulae; Orthogonal curvilinear coordinates; Jacobians, gradient,divergence, curl and Laplacian in curvilinear coordinates; Special curvilinear coordinates.
UNIT 5	*Gama Beta and other Special Functions* covering, the Gama function, values and graph, asymptotic formula for T(n)l The Beta function – Dirichlet integral; Other special functions – Error function, exponential integral, sine and cosine integrals, Bessel's differential equation and function (first and second kind), Legendre differential equation andpolynomials; Some applications.

Course Outcome (CO):

At the end of this course, students will demonstrate ability to:

CO1: Compute dot and cross product of vectors. Use Calculus to compute quantities from physics such as: motion of a particle (velocity, acceleration, distance travelled). Find derivation of vector or scalar point function, gradient, divergence and curl.
CO2: Apply Fundamental Theorem of Line Integrals, Green's Theorem, Stokes' Theorem, or Divergence Theorem to evaluate integrals.
CO3: Familiar with Orthogonal curvilinear coordinates, polar spherical coordinates and cylindrical coordinates, change of variables (Jacobian).
CO4: Use the gamma function, beta function and special functions to evaluate different types of integral calculus problems.
CO5: To approximate polynomials in terms of Legendre's and Bessel's Functions and able to solve Linear differential equations using power series method.

Recommended Books:

1. Erwin Kreyszig , Advanced Engineering Mathematics, Wiley 9th Edition, 2008
2. Thomas and Finney, Calculus and Analytical Geometry, Narosa Publishing House. New Delhi, 2002.
3. M.Ray and Chaturvedi, A Text Book of Differential Equations, Students Friends & Co. Publisher, Agra, 1998.
4. B.V.Ramana, Higher Engineering Mathematics, Tata McGraw Hill, 2011.
5. Maurice D. Weir and Joel Hass, Thomas Calculus, Pearson, 11th Edition, 2005.
6. Rainiville, E.D.: Special Functions. Macmillan & Co. New York (1960).
7. Labedev, W.N.: Special Functions and their Applications. Dover, (1972).

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	**Program Outcome**											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H									M		
CO2		M									M	
CO3			M					H				
CO4				M	H							M
CO5		M		H								

H = Highly Related; M = Medium; L = Low

BAS 007A	DISCRETE MATHEMATICS	2-0-0 [2]

Objective:

- To introduce a number of Discrete Mathematical Structures (DMS) found to be serving as tools even today in the development of theoretical computer science.
- Tosolve problems occurred in the development of programming languages.
- Tofamiliarize students with concepts and techniques of graph theory, and sets apart from languages of logic and proof methods.

UNIT 1	**Sets:** Definition and types, Set operations, Partition of set, Cardinality (Inclusion-Exclusion & Addition Principles), Recursive definition of set. Functions: Concept, Some Special Functions (Polynomial, Exponential & Logarithmic, Absolute Value, Floor & Ceiling, Mod &Div Functions), Properties of Functions, Cardinality of Infinite Set, Countable & Uncountable Sets,
UNIT 2	**Graph Theory**: Graphs – Directed, Undirected, Simple,. Adjacency & Incidence, Degree of Vertex, Subgraph, Complete graph, Cycle & Wheel Graph, Bipartite & Complete Bipartite Graph, Weighed Graph, Union of Simple Graphs. Complete Graphs. Isomorphic Graphs, Path, Cycles & Circuits Euclerian& Hamiltonian Graphs. Planar Graph: Kuratowski's Two Graphs, Euler's Formula, Kuratowski's Theorem. Trees: Spanning trees- Kruskal'sAlgo, Finding Spanning Tree using Depth First Search, Breadth First Search, Complexity of Graph, Minimal Spanning Tree.
UNIT 3	**Semigroups, Groups and Coding**: Binary Operations, Semigroups, Products and Quotients of Semigroups, Groups, Product and Quotients of Groups, Coding of Binary Information and Error Correction, Decoding and Error Correction. **Language of Logic**: Proposition, Compound Proposition, Conjunction, Disjunction, Implication, Converse, Inverse &Contrapositive, Biconditional Statements, tautology, Contradiction & Contingency, Logical Equivalences, Quantifiers, Arguments.
UNIT 4	**Proof Methods**: Vacuous, Trivial, Direct, Indirect by Contrapositive and Contradiction, Constructive & Non-constructive proof, Counterexample. The Division Algorithm, Divisibility Properties (Prime Numbers & Composite Numbers), Principle of Mathematical Induction, The Second Principle of Mathematical Induction, Fundamental Theorem of Arithmetic. Algorithm Correctness: Partial Correctness, Loop Invariant. Testing the partial correctness of linear & binary search, bubble & selection sorting.
UNIT 5	Relations: Boolean Matrices, Binary Relation, Adjacency Matrix of Relation, Properties of Relations, Operations on Relations, The Connectivity Relations, Transitive Closure-Warshall's Algorithm, Equivalence relations- Congruence Relations, Equivalence Class, Number of Partitions of a Finite Set, Partial & Total Orderings.

Course Outcome (CO):

At the end of this course, students will demonstrate ability to:

CO1: Demonstrate complete knowledge on various discrete structures available in literature.
CO2: Realization of some satisfaction of having learnt that discrete structures are indeed useful in computer science and engineering and thereby concluding that no mistake has been done in studying this course.
CO3: Gaining of some confidence on how to deal with problems which may arrive in computer science and engineering in near future.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	L		H		M						L				
CO2		H		H								M	M		
CO3								H	M	L				H	

-
- H = Highly Related; M = Medium L = Low

Text Books

1. B.Kolman et.al- Discrete mathematical Structures, 5th Edn, Pearson Education, New Delhi - 2004.

Reference Books

1. K.H. Rosen – Discrete Mathematics and Its Applications – 4th Edn, Tata McGraw Hill, New Delhi – 2001
2. J.P. Tremblay et.al – Discrete Mathematical Structures with Applications to Computer Science, TMH, New Delhi – 2004.
3. Mott. J.L., Kandel A. and Baker, T.P. "Discrete mathematics", for computer scientists and Mathematicians", Second Edition, Prentice Hall 1986.
4. Tremblay J.P. and Manohar, R. "Discrete Mathematical Structures with Applications to Computer Science", McGraw Hill, 1975.

BCO 001B	SOFTWARE ENGINEERING	3-0-0 [3]

Objective

- To learn about generic models of software development process.
- To understand fundamental concepts of requirements engineering and Analysis Modeling.
- To understand the different design techniques and their implementation.
- To learn various testing and maintenance measures

UNIT 1	Introduction- Introduction to Software Engineering, Software Components, Software Characteristics, Software Crisis, Engineering aspects of Software production – necessity of automation .Job responsibilities of Programmers and Software Engineers as Software developers.Software Development Life Cycle (SDLC)
UNIT 2	Process Models and Program Design Techniques- Software Development Process Models – Code & Fix model, Waterfall model, Incremental model, Rapid Prototyping model, Spiral (Evolutionary) model.Software Requirement Specifications (SRS), Management of User Needs, Data Flow Diagrams, Entity Relationship Diagrams, Decision Tables, SRS Document, Design Techniques – Structured Programming, Coupling and Cohesion, Abstraction and Information Hiding, SoftwareModeling Tools –Data flow Diagrams, UML and XML.
UNIT 3	Software Testing: Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Verification and Validation: Testing of Software Products – Black-Box Testing and White-Box Testing, Static Analysis, Symbolic Execution and Control Flow Graphs –Cyclomatic Complexity.
UNIT 4	Software Project Management: Management Functions and Processes, Project Planning and Control, Organization and Intra-team Communication, Risk Management. Software Cost Estimation – underlying factors of critical concern. Metrics for estimating costs of software products – Function Points. Techniques for software cost estimation –Expert judgment, Work break-down structure and Process breakdown structure, COCOMO and COCOMO-II.
UNIT 5	Software Maintenance,Need for Maintenance, Categories of Maintenance, An Overview of CASE Tools.

Course Outcome (CO):

At the end of this course students will have:

CO1: An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.
CO2: An ability to identify, formulates, and solve engineering problems.
CO3: An understanding of professional and ethical responsibility.
CO4: The broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	L		H		M						L				
CO2		H		H								M	M		
CO3								H	M	L				H	
CO4	L		L			L	M								L

H = Highly Related; M = Medium L = Low

Text Books:

1. Fundamentals of Software Engineering – Carlo Ghezziet. Et.al.
2. Software Engineering – Design, Reliability Management – Pressman.

Reference Books:

1. Software Engineering – Ian Sommerville.
2. Software Engineering - Shoeman.
3. Software Engineering with Abstraction – Berzins and Luqi
4. Pankaj Jalote, Software Engineering, Wiley

BCO 002A	**DATA STRUCTURES AND ALGORITHMS**	**4-0-0 [4]**

OBJECTIVE:

- To study various data structure concepts like Stacks, Queues, Linked List, Trees and Files
- To overview the applications of data structures.
- To be familiar with utilization of data structure techniques in problem solving.
- To have a comprehensive knowledge of data structures and algorithm.
- To carry out asymptotic analysis of algorithm.

UNIT 1	Introduction: Notions of data type, abstract data type and data structures. Importance of algorithms and data structures in programming. Notion of Complexity covering time complexity, space complexity, Worst case complexity & Average case complexity. BigOh Notation, Omega notation, Theta notation. Examples of simple algorithms and illustration of their complexity. Sorting- Bubble sort, selection sort, insertion sort, Quick sort; Heap sort; Merge sort; Analysis of the sorting methods. Selecting the top k elements. Lower bound on sorting.
UNIT 2	Stack ADT, Infix Notation, Prefix Notation and Postfix Notation. Evaluation of Postfix Expression, conversion of Infix to Prefix and Postfix Iteration and Recursion- Problem solving using iteration and recursion with examples such as binary search, Fibonacci numbers, and Hanoi towers. Tradeoffs between iteration and recursion.
UNIT 3	List ADT. Implementation of lists using arrays and pointers. Stack ADT. Queue ADT. Implementation of stacks and queues. Dictionaries, Hash tables: open tables and closed tables. Searching technique- Binary search and linear search, link list- single link list, double link list, Insertion and deletion in link list.
UNIT 4	Binary Trees- Definition and traversals: preorder, post order, in order. Common types and properties of binary trees. Binary search trees: insertion and deletion in binary search tree worst case analysis and average case analysis. AVL trees. Priority Queues -Binary heaps: insert and delete min operations and analysis.
UNIT 5	Graph: Basic definitions, Directed Graphs- Data structures for graph representation. Shortest path algorithms: Dijkstra (greedy algorithm) and Operations on graph, Worshall's algorithm , Depth first search and Breadth-first search. Directed acyclic graphs. Undirected Graphs, Minimal spanning trees and algorithms (Prims and Kruskal) and implementation. Application to the travelling salesman problem.

Course OUTCOME (CO):

CO1: Show the understanding of various data structure concepts like Stacks, Queues, Linked List, Trees and Files
CO2: Understand the applications of data structures.
CO3: Understand with utilization of data structure techniques in problem solving.
CO4: Use comprehensive knowledge of data structures and algorithm.
CO5: Use asymptotic analysis of algorithm.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H												M		
CO2			H		M								M		
CO3		H							M			L		H	
CO4	H	M											L		L
CO5		M		H											L

H = Highly Related; M = Medium; L = Low

Text Books:

1. Data Structures and Algorithms by Alfred V. Aho, Jeffrey D. Ullman and John E. Hopcroft , Addison-Wesley Series (1983)

Reference Books:

1. T.H. Cormen, C.E. Leiserson, and R.L. Rivest. Introduction to Algorithms.The MIT Press and
2. McGraw-Hill Book Company, Cambridge, Massacusetts, 1990 (Available in Indian Edition).
3. Steven S. Skiena. The Algorithm Design Manual.Springer, Second Edition, 2008.
4. Data Structures and Algorithm Analysis in Java (3rd Edition) by Mark Allen Weiss, Addison Wesley(2011).

BCO 003B	**OBJECT ORIENTED PROGRAMMING WITH C++**	**3-0-0 [3]**

OBJECTIVE:

- To perform object oriented programming solution and develop solutions to problems demonstrating usage of control structure, modularity, classes, I/O and the scope of the class members
- To demonstrate adeptness of object oriented programming in developing solution to problems demonstrating usage of data abstraction, encapsulation and inheritance
- To demonstrate ability to implement one or more patterns involving dynamic binding and utilization of polymorphism in the solution of problems
- To learn syntax and features of exception handling
- To demonstrate the ability to implement solution to various I/O manipulation operations and the ability to create two-dimensional graphic components using applets
-

UNIT 1	C++ Overview, C++ Characteristics, Object-Oriented Terminology, Polymorphism, encapsulation ,inheritance, Object-Oriented Paradigm, Abstract Data Types, I/O Services, Standard Template Library, Standards Compliance, Functions and Variables. Declaration and Definition
UNIT 2	Variables: Dynamic Creation and Derived Data, Arrays and Strings in C++,Classes in C++, Defining Classes in C++, Classes and Encapsulation, Member Functions, Instantiating and Using Classes. Friend function ,Inline function
UNIT 3	Using Constructors, Multiple Constructors and Initialization Lists, Using Destructors to Destroy Instances, Using Destructors to Destroy Instances, Operator Overloading: operator overloading of unary and binary operator, Function Overloading, Working with Overloaded Operator Methods, Initialization and Assignment, Initialization vs. Assignment
UNIT 4	Constant and Static Class Members, Inheritance, Overview of Inheritance, Defining Base and Derived Classes, Single, Multiple, multilevel, hybrid hierarchical inheritance. Constructor and Destructor Calls in inheritance, virtual function, virtual base class,
UNIT 5	Input and Output in C++ Programs, Standard Streams, Manipulators, Unformatted Input and Output. Working with files.

Course Outcome (CO):

At the end of this course, students will demonstrate ability to:

CO1: Understand object-oriented programming features in C++,

CO2: Apply these features to program design and implementation,

CO3: Develop applications using Object Oriented Programming Concepts.

CO4: Implement features of object oriented programming to solve real world problems.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M												M		
CO2			H		H									H	L
CO3			H		M				M		M		H		
CO4				L								M		L	

Text Books

1. Let Us C: BalaGuruswamy, TATA McGraw Hill.
2. Programming with C, C++: Yashwant Kanetkar

Reference Books

1. C++:The Complete Reference.
2. The C++ Programming Language:Bjarne Stroustrup

BEE009A	DIGITAL SYSTEMS	4-0-0 [4]

OBJECTIVE:

1. To provide a comprehensive introduction to digital logic design leading to the ability to understand number system representations, binary codes, binary arithmetic and Boolean algebra, its axioms and theorems, and its relevance to digital logic design.
2. To provide introduction to combinational circuits(such as Karnaugh maps),synchronous sequential logic and Asynchrnous sequential logic.

UNIT1	IC Digital Logic Families - Characteristics of digital IC's, Transistor – TransistorLogic family, Standard TTL characteristics, Other TTL series, Open collector TTL, WiredOR/AND connection, Tristate TTL, Emitter-Coupled Logic family, ECL NOR/OR gate
UNIT2	Simplification of Boolean Functions - Using Karnaugh map and Quine-Mccluskey methods, SOP, POS simplification, NAND and NOR implementations,other two-level implementation (AND-OR-INVERT).
UNIT 3	Combinational Logic Design- Design procedure, Adder : Half adder, Full adder,Serial adder, Parallel adder & Carry look-ahead adder, Subtractors : Half subtractor&Fullsubtractor, BCD to Excess-3 code convertor, BCD to 7-segment decoder, Parity generator and checker .
UNIT 4	Combinational Logic Design using MSI Circuits - Application of typical IC‟s like4-bit parallel adder (ex : 7483), Encoders (ex :74148), Multiplexers (ex: 74151, 74153,74157) and their use in realizing boolean functions, Multiplexer trees, Demultiplexer /Decoders (e.g.: 74138, 74154) and their use in realizing a boolean function and demultiplexertrees, 4- it magnitude comparator (ex:7485).
UNIT 5	Synchronous Sequential Logic- Analysis of clocked sequential logic, Statereduction and assignment, Flip-flop excitation tables, Design procedure, Design ofsequentialciruits ex : 3-bit up/down counter (mod < 8), 3-bit up/down gray code counter,Serial adder.

Text Books:

1. M Morris Mano, Digital Design, 3rd Edition, 2006, PHI
2. R. P Jain, Modern Digital Electronics, Second Edition, TMH

Reference Books:

1. Tocci : Digital Systems PHI , 6e, 2001
2. Bignell&Donovan Digital Electronics, 4th Edition, 2007, Thomson Learning.

BCO 004A	OBJECT ORIENTED PROGRAMMING LAB	0-0-2 [2]

List of Experiments

1. Write a program for understanding of C++ program structure without any CLASS declaration. Program may be based on simple input output, understanding of keyword using.

2. Write a Program to Understand Structure & Unions.

3. Write a C++ program to demonstrate concept of declaration of class with public & private member, constructors, object creation using constructors, access restrictions, defining member functions within and outside a class. Scope resolution operators, accessing an object's data members and functions through different type of object handle name of object, reference to object, pointer to object, assigning class objects to each other.

4. Write a Program, involving multiple classes (without inheritance) to accomplish a task &demonstrate composition of class.

5. Write a Program to Demonstrate Friend function, classes and this pointer.

6. Write a Program to Demonstrate Inline functions.

7. Write a Program to Demonstrate pointers to derived classes.

8. Write a Program to demonstrate dynamic memory management using new & delete & static class members.

9. Write a Program to demonstrate an operator overloading, operator functions as member function and/ or friend function, overloading stream insertion and stream extraction, operators, overloading operators etc.

10. Write a Program to demonstrate use of protected members, public & private protected classes, multilevel inheritance etc.

11. Write a Program for multiple inheritance, virtual functions, virtual base classes, abstract classes

12. Write a Program to Demonstrate use of Constructors and Destructors.

13. Write a Program to Develop with suitable hierarchy, classes for Point, Shape, Rectangle, Square, Circle, Ellipse, Triangle, Polygon, etc. Design a simple test application to demonstrate dynamic polymorphism.

14. Write a Program to Show how file management is done in C++.

15. Write a Program to demonstrate class templates.

Course Outcome (CO):

At the end of this course, students will demonstrate ability to:

CO1: Be familiar with language environment

CO2: Implement object oriented concepts to solve problems

CO3: Develop applications using object oriented concepts

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M												M		
CO2			H		H									H	L
CO3			H		M				M		M		H		
CO4				L								M		L	

BCO 005A	DATA STRUCTURE AND ALGORITHMS LAB	0-0-2 [2]

List of Experiments

1.Write a program to implement following searching algorithms using array data structure
- 1.1 Matrix Addition and Subtraction
- 1.2 Matrix Multiplication and Transpose

2.Write a program to implement following searching algorithms using array data structure
- 2.1. Linear Search
- 2.2. Binary Search

3. Write a program to implement following searching algorithms using array data structure
- 3.1. Insertion Sort
- 3.2 Bubble Sort

4. Write a program to implement following searching algorithms using array data structure
- 4.1. Selection Sort
- 4.2 Quick Sort

5. Write a program to implement following operations on stack using array data structure.
- 5.1 Traversing
- 5.2 Push
- 5.3 POP

6. Write a program to implement following examples of recursion
- 6.1 Fibonacci Series
- 6.2 Factorial Function
- 6.3 Tower of Hanoi

7. Write a program to implement Merge Sort.

8. Write a program to implement following operations on Queue using array data structure.
- 8.1 Insertion8.2 Deletion8.3 Traversing

9. Write a program to implement Postfix evaluation.

10. Write a program to implement Infix to Postfix Notation.

11. Write a program to implement following operations on Link List data structure.
- 11.1 Insertion at beginning
- 11.2 Insertion at last
- 11.3 Insertion at any location

12. Write a program to implement following operations on Link List data structure.
- 12.1 Deletion at beginning
- 12.2 Deletion at last
- 12.3 Deletion at any location

13. Write a program to implement Doubly Link List
- 13.1 Insertion13.2 Traversing

14. Write a program to implement Breadth First Search Algorithm.

15. Write a program to implement Depth First Search Algorithm.

Course Outcomes:

Having successfully completed this course, the student will be able to:

CO1: Apply knowledge of computing and mathematics to choose the data structures that effectively model the information in a problem.

CO2: Solve problems by using iterative and recursive methods

CO3: Write various operations like searching, sorting, insertion, deletion, traversing etc. on different data structure.

CO4: Apply programming concepts to solve different problems based on data structures.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO
CO1	H												M		
CO2			H		M								M		
CO3		H							M			L		H	
CO4	H	M											L		L
CO5		M		H											L

H = Highly Related; M = Medium; L = Low

BEE010A	DIGITAL SYSTEMS LAB	0-0-2(2)

List of Experiments

1. Truth Table verification – NAND gate, NOR gate, OR gate, AND gate, NOT gate.
2. Verifying if NAND gate is a universal gate.
3. Verifying if NOR gate is a universal gate.
4. Realizing given truth table using SOP form.
5. Realizing given truth table using POS form.
6. Design and Implementation of Adder and Subtractor.
7. Design and Implementation of Multiplexer and Demultiplexer.
8. Design and Implementation of Binary to gray code converters and vice-versa.
9. Design and Implementation of BCD Adder.
10. Design and Implementation of encoder and decoder.
11. Design and Implementation of parity generator and detector.
12. Design and Implementation of Magnitude Comparator.
13. Design and Implementation of flip flops – RS, JK, D and T flip flops.
14. Design and Implementation of 3-bit synchronous up/down counter.
15. Design and Implementation of SISO, SIPO, PISO and PIPO shift registers using Flip-flops

BCO 080B	LINUX PROGRAMMING LAB	0-0-2 [2]

Lab Experiments:

1.a) Study of Unix/Linux general purpose utility command list: man,who,cat, cd, cp, ps, ls, mv, rm, mkdir, rmdir, echo, more, date, time, kill, history, chmod, chown, finger, pwd, cal, logout, shutdown.

b) Study of vi editor.

c) Study of Bash shell, Bourne shell and C shell in Unix/Linux operating system.

d) Study of Unix/Linux file system (tree structure).

e) Study of .bashrc, /etc/bashrc and Environment variables.

2. Write a C program that makes a copy of a file using standard I/O, and system calls

3. Write a C program to emulate the UNIX ls –l command.

4. Write a C program that illustrates how to execute two commands concurrently with a command pipe. Ex: - ls –l | sort

5. Write a C program that illustrates two processes communicating using shared memory

6. Write a C program to simulate producer and consumer problem using semaphores

7. Write C program to create a thread using pthreads library and let it run its function.

8. Write a C program to illustrate concurrent execution of threads using pthreads library.

Extra Programs

1. Write a shell script that accepts a file name, starting and ending numbers as arguments and displays all the lines between the given line numbers.

2. Write a shell script that displays a list of all files in the current directory to which the user has read, write and execute permissions.

Course Outcome (CO):

At the ends of this course studentswill have:

CO1: The practical knowledge of UNIX/Linux Operating System commands.
CO2: Be able to work confidently in Unix/Linux environment
CO3: Be able to write shell scripts to automate various tasks.
CO4: Be able to learn the important Linux/UNIX library functions and system calls.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H		L										L		
CO2								M					M	M	
CO3				M	M									L	
CO4							M					H			L

H = Highly Related; M = Medium L = Low

BCO 011A	COMPUTER NETWORKS	4-0-0 [4]

OBJECTIVES:

- To build an understanding of the fundamental concepts of computer networking.
- To familiarize the student with the basic taxonomy and terminology of the computer networking area.
- To introduce the student to advanced networking concepts, preparing the student for entry Advanced courses in computer networking.
- To allow the student to gain expertise in some specific areas of networking such as the design and maintenance of individual networks.

UNIT 1	Introduction -Hardware and software, Data communication, Networking, Protocols and standards. Data transmission concepts. Analog and digital transmission. Transmission impairments. Layered Architecture of Computer Networks, OSI and TCP/IP architectures Physical Layer- Guided transmission media and wireless transmission, Data encoding - Digital and analog data. Data communication interface - asynchronous and synchronous transmission, Data link layer - Flow control. Error detection and error control. HDLC and other data link protocols. Multiplexing – Frequency-division, synchronous time-division, and statistical time-division multiplexing
UNIT 2	Link Layer :Medium Access Control: CDMA, ALOHA, and Ethernet; Link Layer Addressing and Forwarding; Spanning Trees; The Channel Allocation Problem, Multiple Access Protocols, Ethernet, Wireless LANs, Broadband Wireless, Bluetooth, Data Link Layer Switching, Switched networks. Circuit-switched networks, switching concepts, Routing incircuit-switched networks. Control signaling. Packet switching principles. Routing and congestion control
UNIT 3	Network Layer: Network layer design issues. Routing algorithms , Flooding, Shortest path routing, Link Sate routing, Hierarchical routing, Broadcast and multicast routings, Routing in the Internet, Path Vector routing, OSPF routing. The network layer in the Internet: IP protocol: ARP and RARP, BOOTP, ICMP, DHCP, Network Address Translation(NAT) Internetworking
UNIT 4	Transport Layer:TCP introduction, Reliable/Un- Reliable Transport, TCP, UDP, Congestion Control, Intra-Domain Routing: Distance-Vector, Intra-Domain Routing: Link- State, Wireless Networks: 802.11 MAC, Efficiency considerations
UNIT 5	Application Layer: DNS-The Domain Name System, Electronic Mail, HTTP, FTP, Simple network management protocol (SNMP), The World Wide Web

***Course Outcome (CO) of* Computer Network**

At the end of this course studentswill have:

CO1: To provide an in-depth understanding of the terminology of network and concepts of OSI reference model and TCP/IP model.
CO2: To equip our students with technical concept of protocols, network interfaces, and design/performance issues in networks.
CO3: To be familiar with contemporary issues in networking technologies.
CO4: To be familiar with network tools and to enhance analytical skills to develop innovative solutions.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H			L									H		
CO2	M		H									L		L	
CO3		M							M				M		L
CO4					H										L

H = Highly Related; M = Medium L = Low

Text Books:

1. Computer Networks, by Andrew S Tanenbaum, PHI. (2010)

Reference Books:

- Data Communications, Computer networking on OSI , by Fred Halsall, Addison Wesley Publishing Co.1998
- Computer Networking -A Top-Down Approach Featuring the Internet ,James F. Kurose and Keith W. Ross ,Addison Wesley Publishing Co. 2004
- Computer Networks: Protocols standards and interfaces , by Uyless Black, Prentice Hall.2002
- Data communication & Networks , by Behrou A. Forouzan, Tata McGraw Hill. 2002
- Data and Computer Communications, by Walliam Stallings, PHI. (2002)

B.Tech CSE Semester IV

BAS 005A	COMPLEX ANALYSIS	2-0-0 [2]

OBJECTIVE:.

- To perform algebra with complex numbers.
- To identify complex-differentiable functions.
- To compute complex line integrals
- To use the residue theorem.

UNIT 1	*Complex Numbers* covering, Functions Analysis including limits and continuity, derivatives; Cauchy Riemann Equations; Integrals, Cauchy theorem and Cauchy integral formulae; Analytic Functions.
UNIT 2	Taylor's series, Singular points and poles; Laurent's Series, Residues, Residue Theorem. Evaluation of definite integrals.
UNIT 3	Conformal mapping, Riemann's mapping theorem; Some general transformations, mapping a half plane into a circle.
UNIT 4	The Schwarz-Christoffel transformation; The solution of Laplace equation by conformal mapping.
UNIT 5	*The complex inverse formula* covering, the Bromwich contour, the use of Residue theorem in finding Laplace transforms; A sufficient condition for the integral around T to approach zero; The case of infinitely many singularities; Application to boundary value problems.

- *Compute sums, products, quotients, conjugate, modulus, and argument of complex numbers.*
- *Determine whether a given function is differentiable, and if so find its derivative.*
- *Identify the isolated singularities of a function and determine whether they are removable,poles, or essential.*

Text Books:

1. Shanti Narayan, Theory of Functions of a Complex Variable, S. Chand &Co.NewDelhi.
2. R.V. Churchil& J.W. Brown, Complex Variables and Applications, 5th Edition, McGraw-Hill, New York, 1990.

Reference Books:

1. S. Ponnuswamy,Introduction To Complex Analysis. Narosa publishers 1993.
2. D. Sarason: Notes on Complex Function Theory.
3. T. W. Gamelin: Complex Analysis.
4. J.B.Conway: Functions of one complex Variable.
5. Mark J., Ablowitz& A.S. Fokas, Complex Variables: Introduction and Applications,Cambridge University Press. South Asian Edition, 1998.
6. Murray.R.Spiegel,Theory and Problems of Complex Variables-,Schaum outline series.
7. P. Duraipandian,Complex Analysis-.

BCO 007A	COMPUTER GRAPHICS	3-0-0 [3]

OBJECTIVE:

- To provide students with a foundation in graphical applications programming
- To introduce students with fundamental concepts and theory of computer graphics
- To give basics of application programming interface (API) implementation based on graphics pipeline approach

UNIT 1	**Introduction to Computer Graphics:**Overview of Computer Graphics, Computer Graphics Application and Software, Description of some graphics devices, Input Devices for Operator Interaction, Active and Passive Graphics Devices,Storage Tube Graphics Displays, Calligraphic Refresh Graphics Displays, Raster Refresh (Raster-Scan) Graphics Displays, Cathode Ray Tube Basics, Color CRT Raster Scan Basics, Video Basics, The Video Controller, Random-Scan Display Processor, LCD displays.
UNIT 2	**Scan conversion – lines, circles and Ellipses; Filling polygons and clipping algorithms:**Scan Converting Lines, Mid-point criteria, Problems of Aliasing, end-point ordering and clipping lines, Scan Converting Circles, Scan Converting Ellipses, Filling Polygons, edge data structure, Clipping Lines algorithms Cohen-Sutherland and Liang-Barsky, Clipping Polygons, problem with multiple components.
UNIT 3	.**Two-Dimensional Transformations:**Transformations and Matrices, Transformation Conventions, 2D Transformations, Homogeneous Coordinates and Matrix Representation of 2D Transformations, Translations and Homogeneous Coordinates, Rotation, Reflection, Scaling, Combined Transformation, Transformation of Points, Transformation of The Unit Square, Solid Body Transformations, Rotation About an Arbitrary Point, Reflection through an Arbitrary Line, A Geometric Interpretation of Homogeneous Coordinates, The Window-to-Viewport Transformations.
UNIT 4	**Three-Dimensional Transformations:**Introduction, Three-Dimensional Scaling, Three-Dimensional Shearing, Three-Dimensional Rotation, Three-Dimensional Reflection, Three-Dimensional Translation, Multiple Transformation, Rotation about an Arbitrary Axis in Space, Reflection through an Arbitrary Plane, Matrix Representation of 3D Transformations, Composition of 3D Transformations, Affine and Perspective Geometry, Perspective Transformations, Techniques for Generating Perspective Views, the Perspective Geometry and camera models, Orthographic Projections, Axonometric Projections, Oblique Projections, View volumes for projections
UNIT 5	**Visible-Surface Determination :**Techniques for efficient Visible-Surface Algorithm Categories of algorithms, Back face removal, The z-Buffer Algorithm, Scan-line metho Painter's algorithms (depth sorting), Area sub-division method, BSP trees, Visib Surface Ray Tracing, comparison of the methods. **Illumination and Shading** Illumination and Shading Models for Polygons, Reflectan properties of surfaces, Ambient, Specular and Diffuse reflections, Atmospher attenuation, Phong's model, Gouraud shading, some examples.

Course Outcome (CO):

At the ends of this course studentswill have:

CO1: Understand the structure of modern computer graphics system
CO2: Understand the basic principles of implementing computer graphics primitives.
CO3: Familiarity with key algorithms for modeling and rendering graphical data
CO4: Develop design and problem solving skills with application to computer graphics

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H			M	L								M		
CO2			M	L					L		L			L	
CO3			L		L						M	L	L		M
CO4			H							L					L

H = Highly Related; M = Medium L = Low

Text Books:

1. Donald Hearn and Pauline Baker, Computer Graphics with OpenGL (third edition), Prentice Hall, 2003

Reference Books:

1.F. S. Hill Jr. and S. M. Kelley, Computer Graphics using OpenGL (third edition), Prentice Hall, 2006
2. Peter Shirley and Steve Marschner, Computer Graphics(first edition), A. K. Peters, 2010
3. Edward Angel, Interactive Computer Graphics. A Top-Down Approach Using OpenGL (fifth Edition), PearsonEducation, 2008

BCO 008B	OPERATING SYSTEMS(UNIX PROGRAMMING)	3-0-0 [3]

OJECTIVE:

- To understand the structure and functions of OS
- To learn about Processes, Threads and Scheduling algorithms
- To understand the principles of concurrency and Deadlocks
- To learn various memory management schemes
- To study I/O management and File systems

UNIT 1	Introduction : Operating system and functions, Classification of Operating systems- Batch, Interactive, Time sharing, Real Time System, Multiprocessor Systems, Multiuser Systems, Multiprocess Systems, Multithreaded Systems, Operating System Structure- Layered structure, System Components, Operating System services, Monolithic and Microkernel Systems.
UNIT 2	Process Management-Process & Threads – Process States - Process Control Block – Process Scheduling – Operations on Processes, Threads, CPU Scheduler – Preemptive and Non- Preemptive; Dispatcher, Scheduling Criteria, Scheduling Algorithms – Process Management in UNIX
UNIT 3	Process Synchronization & Inter process Communication-Concurrent Processes, Co-operating Processes, Precedence Graph, Hierarchy of Processes, Critical Section Problem – Two process solution, Synchronization Hardware, Semaphores – Deadlock- detection, handling, prevention, avoidance, recovery, Starvation, Critical Regions, Monitors, Inter process communication
UNIT 4	Memory Management-Objectives and functions, Simple Resident Monitor Program (No design), Overlays – Swapping; Schemes – Paging – Simple, Multi-level Paging; Internal and External Fragmentation; Virtual Memory Concept, Demand Paging – Page Interrupt Fault, Page Replacement Algorithms; Segmentation – Simple, Multi-level, Segmentation with Paging, Memory Management in UNIX.
UNIT 5	I/O Management and Disk Scheduling: I/O devices, and I/O subsystems, I/O buffering, Disk storage and disk scheduling. File System: File concept, File organization and access mechanism, File directories, and File sharing, File system implementation issues, File system protection and security.

Course Outcome (CO):

At the ends of this course studentswill have:

CO1: Classify Unix Kernel mode with user mode & contrast between Kernel structures.
CO2: Identify and estimate process management & thread management strategies along with their different operations
CO3:Implement different system calls for various file handling operations.
CO4:determine paging and Caching techniques related to Virtual Memory.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H			M				L			L		H		L
CO2		M	L						M					M	
CO3			M		M									M	M
CO4	M		L									L			

H = Highly Related; M = Medium L = Low

Text Books:

1. Operating Systems Concepts – Silberschatz, Galvin, Wiley Publications (2008)
2. Modern Operating Systems - Andrew S. Tanenbaum, Pearson Education Asia / PHI(2005)

Reference Books:

1. Operating Systems – William Stallings, Pearson Education Asia (2002)
2. UNIX System Programming Using C++, by Terrence Chan: Prentice Hall India, 1999.
3. Advanced Programming in UNIX Environment, by W. Richard Stevens: 2nd Ed, Pearson Education, 2005

BCO 009B	COMPUTER ORGANIZATION AND DESIGN	3-1-0 [4]

OJECTIVE:

- To understand the basic structure and operation of digital computer
- To study the design of arithmetic and logic unit and implementation of fixedpointand floating-point arithmetic operations
- To study the two types of control unit techniques and the concept ofpipelining
- To study the hierarchical memory system including cache memories andvirtual memory
- To study the different ways of communicating with I/O devices and standardI/O interfaces

UNIT 1	Basic organization of computers, Register transfer language (RTL), Bus and memory transfer, Arithmetic, logic, shift -micro operations, Types of registers and machine instructions, Fetch, decode and execute cycle.
UNIT 2	Assembly language programming, Instruction format, addressing modes , RISC vs CISC architectures.
UNIT 3	Information representation, Floating point representation (IEEE 754), computer arithmetic and their implementation; Fixed-Point, Signed and 2's complement Arithmetic: Addition, Subtraction, Multiplication and Division, Hardwired and Micro programmed Control.
UNIT 4	Memory Technology, static and dynamic memory, Memory address mapping and cache memory mapping techniques, Memory Hierarchy, Virtual memory and memory management unit
UNIT 5	I/O subsystems: Input-Output devices such as Disk, CD-ROM, Printer etc. Interfacing with IO devices, keyboard and display interfaces; Basic concepts Bus Control, Read Write operations, Programmed IO, Concept of handshaking, Polled and Interrupt-driven I/O, DMA data transfer.

Course Outcome (CO):

At the ends of this course studentswill have:

CO1: Awareness of computer organization.
CO2: Design and architecture of machine.
CO3:Implement different system calls for various units.
CO4: Logical representation of storage, representation and management.
CO5: Analysis of I/O subsystem.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H											M	H		
CO2			H		M									M	
CO3				M					M				L		
CO4				H						M			M		L
CO5				H						M					L

H = Highly Related; M = Medium L = Low

Text Book:

1. Computer Organization by V. Carl Hamacher, Safwat G. Zaky and Zvonko G. Vranesic , McGraw-Hill series(2002)

Reference Books:

1. Computer Organization and Design, by David Patterson and John Hennessey," Elsevier. 2008.

2. Computer System Architecture by Mano, M.M.,, Prentice Hall of India, New Delhi, 1992

3. Computer Systems Design and Architecture (2nd Edition) by Vincent P. Heuring and Harry F. Jordan (Dec 6, 2003)

4. Computer Architecture and Organization, by Hayes, J.P.1998, McGraw-Hill

.BCO 010C	DATABASE MANAGEMENT SYSTEMS	4-0-0 [4]

OJECTIVE:

- To provide knowledge of relational model
- To learn about ER diagrams.
- To learn about Query Processing and Transaction Processing

UNIT 1	Introduction - Database Systems versus File Systems, View of Data, Data Models, database languages, Database Users and Administrators. Transaction Management, Components of a Database management System. Entity-Relationship Model – Basic Concepts, Constraints, Keys, Design Issues, E-R Diagrams.
UNIT 2	Relational Model- Structures of relational databases, Integrity Constraints, Logical database Design, Tables, Views, Data Dictionary. Relational Algebra, Relational Calculus. SQL – Basic Structures, Query Handling,Triggers,Nested SQL Query, Embedded SQL,
UNIT 3	Relational Database Design- Functional Dependencies, Multi-valued Dependencies, Normal Forms, Decomposition into Normalized Relations.
UNIT 4	Fundamental Concepts of Transaction Management, ACID property. Serializability and testing for serializability, concurrency control schemes, lock-based protocols, two-phase locking protocols, graph-based protocols, time stamp-based protocols, deadlocks.
UNIT 5	File System: File organization- Heap File, Sequential File, Hash File, Clustered file, file operations, indexing, B-tree, B+ tree, Introduction to Data Mining, Data Farming, Data Warehousing

Course Outcome (CO):

At the ends of this course students will have:

CO1: Awareness of database management basics and different models that we use for database.
CO2: Design and architecture of relational model, relational algebra and SQL queries.
CO3: Implement different form of normalization.
CO4: Logical representation of internet database.
CO5: Analysis and concepts of transaction, concurrency and recovery systems.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H												H	M	
CO2			H		M				M						L
CO3				H		M							M		
CO4				M								M		L	L
CO5	M	L		H					L				M		

H = Highly Related; M = Medium L = Low

Text Books:

1. Database Systems Concepts – Korthe, TMH
2. An Introduction to Database Design – Date

Reference Books:

1. Fundamentals of Database Systems – Elmasri and Navathe
2. Database Management and Design – Hansen and Hansen .
3. Object-Oriented Database Design – Harrington

BCO 081B	PROGRAMMING WITH PYTHON	2-0-0 [2]

OBJECTIVE:

- To study various core programming basics—including data types, control structures, algorithm development,
- To overview the applications of Python.
- To be familiar with program design with functions—via the Python programming language.
- Students will solve problems, explore real-world software development challenges, and create practical and contemporary applications

UNIT 1	**Introduction:** Features of Python, History of Python, installing Python; basic syntax, interactive shell, editing, saving, and running a script. The concept of data types; variables, assignments; immutable variables; numerical types; arithmetic operators and expressions; comments in the program; understanding error messages
UNIT 2	Introduction to Operators, Control statements: if-else, loops (for, while); short-circuit (lazy) evaluation. **Strings:** subscript operator, indexing, slicing a string, String methods & operations; strings and number system: converting strings to numbers and vice versa. Binary, octal, hexadecimal numbers. **Text files;** manipulating files and directories, os and sys modules; reading/writing text and numbers from/to a file; creating and reading a formatted file
UNIT 3	**Lists, tuples, and dictionaries**; basic list operators, replacing, inserting, removing an element; searching and sorting lists; dictionary literals, adding and removing keys, accessing and replacing values; traversing dictionaries. **Design with functions:** hiding redundancy, complexity; arguments and return values; formal vs actual arguments, named arguments. Program structure and design. Recursive functions.
UNIT 4	**Classes and OOP:** classes, objects, attributes and methods; defining classes; design with classes, data modeling; persistent storage of objects **OOP, continued:** inheritance, polymorphism Operator overloading (_eq_, _str_, etc); abstract classes; Exception handling, try block
UNIT 5	**Graphical user interfaces;** Event-driven programming paradigm; tkinter module,,turtle module, creating simple GUI; buttons, labels, entry fields, dialogs; widget attributes - sizes, fonts, colors layouts, nested frames Multithreading, CSV(Accesing, updating, Creating)

Course Outcome:

CO1: Various core programming basics—including data types, control structures, algorithm development,
CO2: Overview the applications of Python.
CO3: Show the program design with functions—via the Python programming language.
CO4: Students will solve problems, explore real-world software development challenges, and create practical and contemporary applications

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H		M										H		
CO2		L			M									L	L
CO3			H		M						L		M		L
CO4		M		L					L	L				M	

H = Highly Related; M = Medium L = Low

Text Book:

1. *Fundamentals of Python: First Programs* Author: Kenneth Lambert Publisher: Course Technology, Cengage Learning, 2012 ISBN-13: 978-1-111-82270-5

Reference Books:

1. Python: Real World Machine LearningBy Prateek Joshi et al.ISBN 13: 9781787123212 Packt Publishing 941 pages (November 2016)

BCO 012A	SOFTWARE PROJECT MANAGEMENT	3-0-0 [3]

OBJECTIVE:

- To understand Software Project Management Concepts.
- To understand Risk Analysis.
- To Study about Software Quality Management.
- To understand Project Evaluation.

UNIT 1	**Project Management:** Definition of the Project, Project Specification and parameters, Principles of Project Management, Project Management Life Cycle.
UNIT 2	**Software Project Planning**, Project Sequencing and Scheduling Activities, Scheduling resources, Network Planning, Work Breakdown Structure, Activity Resource Requirements, Project Management Plan , Critical path analysis PERT & CPM.
UNIT 3	**Project Scheduling and Tracking Techniques**: Why projects are delayed? Effort Estimation Techniques, Task Network and Scheduling Methods, Monitoring and Control Progress, Graphical Reporting Tools. **Monitoring & Control** : Change Control, Software Configuration Management (SCM)
UNIT 4	**Risk Analysis and Management**: Risk Mitigation and Management, Software Metrics and Project Management
UNIT 5	**Quality Management and People Management**- Introduction, Understanding Behaviour, Organizational Behaviour, Selecting The Right Person For The Job, Motivation, The Old man – Hackman Job Characteristics Model , Working in Groups, Organization and team structures, Decision Making, Leadership, Organizational Structures, Stress, Health And Safety.

Course Outcome:

CO1: Understand and practice the process of project management and its application in delivering successful IT projects;

CO2: Evaluate a project to develop the scope of work, provide accurate cost estimates and to plan the various activities

CO3: Understand and use risk management analysis techniques that identify the factors that put a project at risk and to quantify the likely effect of risk on project timescales

CO4: Identify the resources required for a project and to produce a work plan and resource Schedule

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H										M		M		
CO2		M			M	L								L	M
CO3							M		M				L		
CO4			L					M	L		L		L		M

H = Highly Related; M = Medium L = Low

Text Books:

1.Bob Hughes, Mike Cotterell, "Software Project Management", Tata McGraw Hill. (2009)

Reference Books:

1. Royce, "Software Project Management", Pearson Education. (2005).
2. Robert K. Wysocki, "Effective Software Project Management", Wiley.(2006)

BCO 013A	DATABASE MANAGEMENT SYSTEMS LAB	0-0-2 [2]

List of Experiments

1	Installation of MySQL
2	Analyze the problem and come with the entities in it. Identify what Data has to be persisted in the databases.
3	Represent all entities in a tabular fashion. Represent all relationships in a tabular fashion.
4	Creating of Tables on given problem
5	Applying Not Null, Check, Unique Constraints on database Tables.
6	Applying Primary Key, References, Foreign Key Constraints on database Tables.
7	Applying Insert, Select, Distinct Clause, Where Clause on database Tables.
8	Applying Update, Delete, Drop, on database Tables.
9	Applying table creation with select, Insert data using select, Renaming on database Tables.
10	Practice Queries using MINUS, UNION, INTERSECT, % operator.
11	Practice Queries using Group Functions.
12	Practice Queries using Group By, Having, Order By Functions.
13	Practice Queries using Arithmetic Operators, Comparison Operator.
14	Practice Queries using Logical Operator.
15	Practice Queries using any four String Functions.
16	Practice Queries using any four String Functions.
17	Practice Queries using Numeric Functions.
18	Practice Queries using Date Functions.

Course Outcome (CO):

At the ends of this course studentswill have:

CO1: Awareness of database management basics and different models that we use for database.
CO2: Design and architecture of relational model, relational algebra and SQL queries.
CO3: Implement different form of normalization.
CO4: Logical representation of internet database.
CO5: Analysis and concepts of transaction, concurrency and recovery systems.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H												H	M	
CO2			H		M				M						L
CO3				H		M							M		
CO4				M								M		L	L
CO5	M	L		H					L				M		

H = Highly Related; M = Medium L = Low

BCO 014A	OPERATING SYSTEMS LAB	0-0-2 [2]

List of Experiments

Experiment No	Aim
1	Write a C program to implement the various process scheduling mechanisms such as FCFS scheduling.
2	Write a C program to implement the various process scheduling mechanisms such as SJF Scheduling.
3	Write a C program to implement the various process scheduling mechanisms such as Round Robin Scheduling.
4	Write a C program to implement the various process scheduling mechanisms such as Priority Scheduling.
5	To implement deadlock avoidance & Prevention by using Banker's Algorithm.
6	To implement page replacement algorithms FIFO (First In First Out).
7	To implement page replacement algorithm LRU (Least Recently Used).
8	To implement page replacement algorithms Optimal (The page which is not used for longest time)
9	To implement the memory management policy- Paging.
10	To implement the memory management policy-segmentation.
11	Write a C Program to implement Sequential File Allocation method.
12	Write a C Program to implement Indexed File Allocation method.
13	Write a C Program to implement Linked File Allocation method.
14	Write a program to implement multi program variable task (MVT).
15	Write a program to implement multi program fixed task (MFT).

Course Outcome (CO):

At the ends of this course studentswill have:

CO1: Classify Unix Kernel mode with user mode & contrast between Kernel structures.
CO2: Identify and estimate process management & thread management strategies along with their different operations
CO3:Implement different system calls for various file handling operations.
CO4:determine paging and Caching techniques related to Virtual Memory.

CO5: construct shell scripts.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H			M				L			L		H		L
CO2		M	L						M					M	
CO3			M		M									M	M
CO4	M		L									L			

H = Highly Related; M = Medium L = Low

BCO 082A	PROGRAMMING WITH PYTHON LAB	0-0-2 [2]

List of Experiments

1. Write a Python program to print the documents (syntax, description etc.) of Python built-in function(s).
2. Write a Python program which accepts the radius of a circle from the user and compute the area.
3. Write a Python program to accept a filename from the user print the extension of that.
4. Write a Python program to print the calendar of a given month and year.
5. Write a Python program to calculate number of days between two dates.
6. Write a Python program to calculate the length of a string.
7. Write a Python program to multiplies all the items in a list.
8. Write a Python script to sort (ascending and descending) a dictionary by value.
9. Write a Python program to create a tuple with different data types.
10. Write a Python program to find those numbers which are divisible by 7 and multiple of 5, between 1500 and 2700 (both included).
11. Write a Python program to guess a number between 1 to 9.(User is prompted to enter a guess. If the user guesses wrong then the prompt appears again until the guess is correct, on successful guess, user will get a "Well guessed!" message, and the program will exit.)
12. Write a Python program to count the number of even and odd numbers from a series of numbers.
13. Write a Python function to find the Max of three numbers.
14. Write a Python function to sum all the numbers in a list.
15. Write a Python function that takes a list and returns a new list with unique elements of the first list.
16. Write a Python class to find validity of a string of parentheses, '(', ')', '{', '}', '[' and ']. These brackets must be close in the correct order, for example "()" and "()[]{}" are valid but "[)", "({[)]" and "{{{" are invalid.
17. Write a Python class to find a pair of elements (indices of the two numbers) from a given array whose sum equals a specific target number.
18. Write a Python class to implement pow(x, n).

Course Outcome:

CO1: Various core programming basics—including data types, control structures, algorithm development,
CO2: Overview the applications of Python.
CO3: Show the program design with functions—via the Python programming language.
CO4: Students will solve problems, explore real-world software development challenges, and create practical and contemporary applications

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H		M										H		
CO2		L			M									L	L
CO3			H		M						L		M		L
CO4		M		L					L	L				M	

H = Highly Related; M = Medium L = Low

BCO 015A	COMPUTERGRAPHICS LAB	0-0-2 [2]

List of Experiments

1 Write a Program to Show basic Transformation with OpenGL
2 Write a Menu Driven Program with OpenGL
3 Write a Program to draw a line using Bresenham's Algorithm with OpenGL
4 Write a Program to implement midpoint algorithm to draw circle
5 Write a Program to implement midpoint algorithm to draw ellipse
6 Program to implement 2d scaling about an arbitrary axis.
7 Write a program to implement DDA line Algorithm
8 Program to implement 2d rotation about an arbitrary axis.
9 Program to implement translation of a line and triangle.
10 Program to implement Cohen Sutherland line clipping.
11 Program to implement Sutherland Hodgeman polygon clipping.
12 Program to draw Bezier curve.
13 Program to draw b-spline curve.
14 Program to implement a line using slope intercept formula.
15 Write a program to implement Bresenham 's Algorithm

Course Outcome (CO):

At the ends of this course studentswill have:

CO1: Understand the structure of modern computer graphics system
CO2: Understand the basic principles of implementing computer graphics primitives.
CO3: Familiarity with key algorithms for modeling and rendering graphical data
CO4: Develop design and problem solving skills with application to computer graphics

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H			M	L								M		
CO2			M	L					L		L			L	
CO3			L		L						M	L	L		M
CO4			H							L					L

H = Highly Related; M = Medium L = Low

B.Tech CSE Semester V

BAS 004A	OPTIMIZATION AND CALCULUS OF VARIATIONS	2-0-0 [2]

OBJECTIVE:

- To understand the Linear Programming Problem formulation and solution from graphical method.
- To describe methods for solving Non linear programs.
- To develop an understanding of Transportation, assignment problem and dynamic programming.
- To develop an understanding of Calculus of Variations.

UNIT 1	*First and second order conditions* for local interior optima (concavity and uniqueness), Sufficient conditions for unique global optima; Constrained optimization with Lagrange multipliers; Sufficient conditions for optima with equality and inequality constraints; Kuhn Tucker conditions.
UNIT 2	Linear Programming (Graphical and Simplex solution); Transportation and Assignment Method.
UNIT 3	Elements of dynamic programming including Hamiltonian, Bellman's optimality principle
UNIT 4	*Calculus of Variations:* Basic definition, Simplest problem, Isoperimetric problem, Problems with higher order derivatives.
UNIT 5	Euler Lagrange equation, Weierstrass-Erdmann conditions; Pontryagin maximum principle; Transversality condition and applications.

OUTCOMES: At the end of the course, the student should be able to:

- Understanding the solutions methods for nonlinear programming problems.
- Methods for Linear programming, transportation and assignment problem.
- Develop an understanding of Calculus of Variations.

Text Books:

1. A.S. Gupta.. Calculus of Variation, Prentice Hall of India Pvt. Ltd, 2002.
2. I.M.Gelfand and S. V. Francis.. Calculus of Variation, Prentice Hall, New Jersey, 1998.
3. Sharma S. D., Operations Research : Theory, Methods & Applications, KEDAR NATH RAM NATH-MEERUT, 2011.
4. Kapoor V.K., Operations Research, Sultan Chand & Sons, 2004.

Reference Books:

1. G. Hadley, Linear Programming, Narosa Publishing House, 1995.
2. S.I. Gass, Linear Programming: Methods and Applications (4th edition) McGraw-Hill,New York, 1975

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H								M	M		M	M		
CO2		H	H								M	M	M		
CO3			H		M		M						L	M	M
CO4				M	H			L			M	M			L
CO5	H	H			M	L						M			

H = Highly Related; M = Medium L=Low

BCO 017A	FORMAL LANGUAGES & AUTOMATION THEORY	3-1-0 [4]

Objective:

- To understand various Computing models like Finite State Machine, Pushdown Automata, and Turing Machine.
- To understand Decidability and Undesirability of various problems
- To construct pushdown automata and the equivalent context free grammars.
- To prove the equivalence of languages described by pushdown automata and context free grammars.
- To construct Turing machines and Post machines and prove the equivalence of languages described by Turing machines and Post machines.

UNIT 1	Basics of Strings and Alphabets, Finite Automata – DFA, transition graphs, regular languages, non-deterministic FA, equivalence of DFA and NDFA, Mealy and Moore Machine, minimization of Finite Automata,
UNIT 2	Regular grammars, regular expressions, equivalence between regular languages, properties of regular languages, pumping lemma. Relationship between DFA and Regular expression.
UNIT 3	Context Free Languages – Leftmost and rightmost derivation, parsing and ambiguity, ambiguity in grammar and languages, simplification of CFG, Normal forms
UNIT 4	Pushdown Automata – NDPDA, DPDA, context free languages and PDA,comparison of deterministic and non-deterministic versions, closure properties, pumping lemma for CFL,
UNIT 5	Turing Machines, variations, halting problem, PCP, Chomsky Hierarchy, Recursive and Recursive enumerable language, Rice Theorem.

Course Outcomes: At the end of the course, the student should be able to:

CO1:Understand and construct finite state machines and the equivalent regular expressions.
CO2:Prove the equivalence of languages described by finite state machines and regular expressions.
CO3:Construct pushdown automata and the equivalent context free grammars.
CO4:Prove the equivalence of languages described by pushdown automata and context free grammars.
CO5:Construct Turing machines and Post machines and prove the equivalence of languages described by Turing machines and Post machines

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H		L										H		
CO2		H											L		
CO3	H		H												M
CO4		H		M											
CO5	H											H		L	

H = Highly Related; M = Medium L = Low

Text Books:

1. Hopcroft J.E., Motwani R. and Ullman J.D, "Introduction to Automata Theory,Languages and Computations", Second Edition, Pearson Education, 2008.

Reference Book:

1. Mishra K L P and Chandrasekaran N, "Theory of Computer Science – Automata, Languages and Computation", Third Edition, Prentice Hall of India, 2004.
2. Harry R Lewis and Christos H Papadimitriou, "Elements of the Theory of Computation", Second Edition, Prentice Hall of India, Pearson Education, New Delhi, 2003.
3. Peter Linz, "An Introduction to Formal Language and Automata", Third Edition, Narosa Publishers, New Delhi, 2002.
4. Kamala Krithivasan and Rama. R, "Introduction to Formal Languages, Automata Theory and Computation", Pearson Education 2009.
5. John C Martin, "Introduction to Languages and the Theory of Computation", Third Edition, Tata McGraw Hill Publishing Company, New Delhi, 2007.

BCO 018A	OBJECT ORIENTED ANALYSIS AND DESIGN	2-0-0 [2]

Objective: At the end of the course, the student should be able to:

- To provide a brief, hands-on overview of object-oriented analysis in software process
- To understand the Object Basics, Classes and Inheritance
- To make utilization of software objects to build systems that are more robust.
- To familiarize the Object-Oriented Analysis and Design (OOAD) concepts for developing Object Oriented Projects

UNIT1	Introduction to object oriented systems, Classes, Objects, Abstraction, Inheritance,Polymorphism, Encapsulation, Message Sending, Association, Aggregation, Iterativedevelopment and the Unified Process (UP), UP phases: Inception, Elaboration, Constructionand Transition, Object-oriented metrics
UNIT2	Introduction to UML, Use Cases and functional requirements, Identifying andwriting Use Cases, Decomposition of use cases, Modeling System Workflows using ActivityDiagrams, Modeling a System's Logical Structure using Classes and Class Diagrams,Modeling Interactions using Sequence Diagrams and Communication Diagrams, TimingDiagrams, Interaction Overview Diagrams, Component Diagram, Package diagram, StateMachine Diagrams, Deployment Diagrams.
UNIT3	Introduction to Patterns, GoF Patterns, Creational Patterns, Structural Patterns,Behavioral Patterns, Software Architectural patterns, The Observer Pattern, The TemplateMethod Pattern , Factory Patterns: Factory Method and Abstract Factory , The SingletonPattern , The Iterator Pattern , The Composite Pattern , The Facade Pattern , The State andStrategy patterns , Command Pattern , The Adapter Pattern , The Proxy Pattern , TheDecorator Pattern, The Visitor Pattern , AntiPatterns, Patterns for Assigning Responsibilities:GRASP Patterns
UNIT4	Domain modeling, assigning responsibility using sequence diagrams, mappingdesign to code, CASE tools, Unit, Cluster, and System-level testing of Object-orientedprograms, Aspect- oriented and Service-oriented software.
UNIT 5	SOFTWARE QUALITY AND USABILITY Designing interface objects – Software quality assurance – System usability – Measuring user satisfaction

Outcomes:

At the end of this course, students will be able to:

CO1: Explain the object- oriented software development process, including object-oriented methodologies and work flows.

CO2: Use Object-Oriented Analysis and Design (OOAD) concepts for developing Object Oriented Projects

CO3: Designs software based on design principles, patterns, and heuristics, OOAD.

CO4:Better utilization of CASE tool and testing technique to build systems that are more robust.

CO5:Analyze the software quality on different parameters design the effective user interface.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H								M	M		M	M		
CO2		H	H								M	M	H		
CO3			H		M		M							H	M
CO4				M	H			L			M	M			M
CO5	H	H			M	L						M	M		

H = Highly Related; M = Medium L=Low

Text Books:

1.Applying UML and Patterns: An Introduction to object-oriented Analysis and Design and iterativedevelopment, by Craig Larman, Pearson Education. (1998)

2. Object oriented Analysis and design using UML by Jeya mala,Tata McGraw Hill

Reference Books:

Design Patterns - Elements of Reusable Object-Oriented Software, Gamma, et. al., Addison-Wesley. (1994)

BCO 019A	ARTIFICIAL INTELLIGENCE	4-0-0 [4]

Objective:

- To explain the basic principles of artificial intelligence;
- To apply logic and structured concepts in knowledge representation;
- To discuss the applications of artificial intelligence;
- To implement heuristic search algorithms;
- To design a natural language processor and implement a simple expert system.

UNIT 1	Introduction- What is intelligence? Foundations of artificial intelligence (AI), Task of artificial intelligence, Techniques of artificial intelligence, Problem Solving- Formulating problems, problem types, states and operators, state space, Expert system and its components.
UNIT 2	*Uni*nformed Search Strategies- Breath First Search, Depth First Search, Depth Limited Search, Informed Search Strategies- Heuristic Functions, Best First Search, Hill Climbing Algorithm, Problems and solutions of Hill Climbing, Iterative Deepening (IDA), A* algorithm, AO* Algorithm.
UNIT 3	Game playing- Introduction, Types of games, Minimax game algorithm, Alpha Beta cut-off procedure. Knowledge Representation- Role of Knowledge, Declarative Knowledge, Procedural Knowledge, Knowledge representation.
UNIT 4	Logics- propositional logics, First Order Predicate Logics (FOPL), Syntax of First Order Predicate Logics, Properties of Wff, Clausal Forms, Conversion to clausal forms.
UNIT 5	Planning- Introduction, Basic representation of plans, partial order planning, planning in the blocks world, Goal Stack Planning, Non-linear planning using constraint posting (TWEAK method).

Outcomes:

Upon the end of this course, student will be able to:

CO1 : Familiar basic principles of artificial intelligence;
CO2 : Able to use logic and structured concepts in knowledge representation;
CO3 : To discuss the applications of artificial intelligence;
CO4 : To implement heuristic search algorithms;
CO5 :To design a natural language processor and implement a simple expert system.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome		
	PO 1	PO2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
CO1	H	L		H	H	L	M	M	M	M	L	H	L	M	
CO2	H	H		H	H	L	M	M	M	M	M	H		H	
CO3	H	M	M	H	H	L	M	H	M	M	L	H		H	
CO4	H	H	H	H	H	L	M	M	M	M	M	H	H	M	H
CO5	H	H	H	H	H	L	M	H	M	M	L	H	H	M	M

H = Highly Related; M = Medium L = Low

Text Books:

1.Stuart Russell and Peter Norvig. *Artificial Intelligence – A Modern Approach,* Pearson Education Press, 2001.

2.Kevin Knight, Elaine Rich, B. Nair, *Artificial Intelligence, McGraw Hill, 2008.*

Reference Books:

1.George F. Luger, *Artificial Intelligence,* Pearson Education, 2001.

2.Nils J. Nilsson, *Artificial Intelligence: A New Synthesis,* Morgan Kauffman, 2002.

BCO 083A	WEB TECHNOLOGY LAB	0-0-2 [2]

List of Experiments

1. Write programs in Java to demonstrate the use of following components:
   i. Text fields, buttons, Scrollbar, Choice, List and Check box.
2. Write Java programs to demonstrate the use of various Layouts like Flow Layout,
   i. Border Layout, Grid Layout and card layout.
3. Write programs in Java to create applets incorporating the following features:
   i. Create a color palette with matrix of buttons
   ii. Set background and foreground of the control text area by selecting a color from color palette.
   iii. In order to select Foreground or background use check box control as radio buttons
4. Write programs in Java to do the following.
   i. Set the URL of another server. ii. Download the homepage of the server.
   iii. Display the contents of homepage with date, content type, and Expiration date. Last modified and length of the home page.
5. Write programs in Java using sockets to implement the following:
   i. HTTP request ii. FTP iii. SMTP iv. POP3
6. Write a program in Java for creating simple chat application with datagram sockets and datagram packets.
7. Write programs in Java using Servlets:
   i. To invoke servlets from HTML forms ii. To invoke servlets from Applets
8. Write programs in Java to create three-tier applications using servlets for conducting on-line examination for displaying student mark list. Assume that student information is available in a database which has been stored in a database server.
9. Create a web page with the following using HTML
   i. To embed a map in a web page ii. To fix the hot spots in that map
   iii. Show all the related information when the hot spots are clicked.
10. Create a web page with the following.
    i. Cascading style sheets.
    ii. Embedded style sheets.
    iii. Inline style sheets. Use our college information for the web pages.

Course Outcome (CO):

At the end of this course studentswill have:

CO1: Able to understand the basics of computer network, various protocols
CO2: Ability to understand WWW and HTML language
CO3: Ability to develop projects by formatting HTML documents &managing images in HTML
CO4: Able to understand Hypertext and Link in HTML
CO5: Ability to understand PHP programming language

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1		H		H		L		M			H		H		
CO2	H			M			L			H				M	
CO3			M		H				M			L	L		H
CO4	H	M		L								L	M	L	
CO5	M			H		L				M				H	

TEXT BOOKS

1. Deitel H.M. and Deitel P.J., “Internet and World Wide Web How to program”, Pearson International, 2012, 4th Edition.

2. Gopalan N.P. and Akilandeswari J., “Web Technology”, Prentice Hall of India, 2011Paul Dietel and Harvey Deitel,”Java How to Program”, Prentice Hall of India, 8th Edition C

BCO 021A	Object Oriented Analysis & Design Lab	0-0-2 [2]

Objective: The student is expected to take up about five mini-projects and model them and produce Use Cases, Analysis Documents - both static & dynamic aspects, Sequence Diagrams and State-Charts, Database Design using Rational Products A sample collection of ideas is given. Numerous other ideas can be found in the pages from the list of references given below.

Mini-Project - I: A Point-of-Sale (POS) System

A POS system is a computerized application used to record sales and handle payments; it is typically used in a retail store, it includes hardware components such as a computer and bar code scanner, and software to run the system. It interfaces to various service applications, such as a third-party tax calculator and inventory control. These systems must be relatively fault tolerant; that is, even if remote services are temporarily unavailable they must still be of capturing sales and handling at least cash payments. A POS system must support multiple and varied client-side terminals and interfaces such as browser, PDAs, touch-screens.

Mini-Project - II: Online Bookshop Example

Following the model of amazon.com or bn.com, design and implement an online bookstore.

Mini-Project - III: A Simulated Company

Simulate a small manufacturing company. The resulting application will enable the user to take out a loan, purchase a machine, and over a series of monthly production runs, follow the performance of their company.

Mini-Project - IV: A Multi-Threaded Airport Simulation

Simulate the operations in an airport. Your application should support multiple aircrafts usingm several runways and gates avoiding collisions/conflicts. Landing: an aircraft uses the runway, lands, and then taxis over to the terminal. Take-Off: an aircraft taxies to the runway and then takes off

Mini-Project -V: An Automated Community Portal

Business in the 21st Century is above all BUSY. Distractions are everywhere. The current crop of "enterprise intranet portals" are often high noise and low value, despite the large capital expenditures it takes to stand them up. Email takes up 30 - 70% of an employee's time. Chat and Instant Messaging are either in the enterprise or just around the corner. Meanwhile, management is tasked with unforseen and unfunded leadership and change-agent roles as well as leadership development and succession management. What is needed is a simplified, repeatable process that enhances communications within an enterprise, while allowing management and peers to self-select future leaders and easily recognize high performance team members in a dynamic way. Additionally, the system should function as a general-purpose confent management, business intelligenceand peer-review application.

Glasscode's goal is to build that system. The software is released under a proprietary license, and will have the following features: Remote, unattended moderation of discussions However, it will have powerful discovery and business intelligence features, and be infinitely extendable, owing to a powerful API and adherence to Java platform standards. Encourages peer review and indicates for management potential leaders, strong team players and reinforces enterprise and team goals seamlessly and with zero administration.

Mini-Project-VI: An Auction Application

Several commerce models exist and are the basis for a number of companies like eBay.com, pricellne.com etc. Design and implement an auction application that provides auctioning services. It should clearly model the various auctioneers, the bidding process, auctioning etc.

Mini-Project -VII: A Notes and File Management System

In the course of one's student years and professional career one produces a 1 lot of personal notes and documents. All these documents are usually kept 1 on papers or individual files on the computer. Either way the bulk of the I information is often erased corrupted and eventually lost. The goal of this 1 project is to build a distributed software application that addresses this "| problem. The system will provide an interface tocreate, organize and manage I personal notes through the Internet for multiple users. The system will also allow users to collaborate by assigning permissions for multiple users to view and edit notes.

Mini-Project - VIII: A Customizable Program Editor

A programmer's editor which will be focused on an individual programmer's particular needs and style. The editor will act according to the specific language the current source file is in, and will perform numerous features, such as auto-completion or file summarization, on the file. These features will be able to be turned on or off by the programmer, and the programming style of the user will be used to create as efficient an editing environment as possible.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H								M	M		M	M		
CO2		H	H								M	M	H		
CO3			H		M		M							H	M
CO4				M	H			L			M	M			M
CO5	H	H			M	L						M	M		

Text Book(s):

1. "Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and the Unified Process", Craig Larman, Pearson Education Asia, 2002, 2nd Edition

Reference(s):

1. "Object Oriented Systems Analysis and Design using UML", Simon Sennet, Steve McRobb, and Ray Farmer, McGraw Hill, 2002, 2nd Edition
2. "Object-Oriented Analysis & Design," Andrew Haigh, Tata McGraw-Hill, 2001,

Program Elective-I (V Semester)

BCO 035A	**Programming in Java**	**4:0:0 [4]**

Objective

- Cover issues related to the definition, creation and usage of classes, objects and methods.
- Discuss the principles of inheritance and polymorphism and demonstrate though problem analysis assignments how they relate to the design of methods, abstract classes and interfaces.
- Provide the foundation of good programming skills by discussing keys issues to the design of object-oriented software, including programming design patterns, automatic documentation techniques and programming testing.
- Cover the basics of creating APIs as well as allow students to explore the Java Abstract Programming Interface (API) and Java Collection Framework through programming assignments.
- Discuss basic principles and tools of collaborating programming (versioning systems, code review) and study their usage through group programming projects.

UNIT 1	**Java Fundamentals:** Features of Java ,OOPs concepts , Java virtual machine , Reflection byte codes ,Byte code interpretation , Data types, variable, arrays, expressions, operators, and control structures , Objects and classes
UNIT 2	**Java Classes:** Abstract classes ,Static classes ,Inner classes ,Packages,Wrapper classes Interfaces ,This ,Super ,Access control
UNIT 3	**Exception handling:** Exception as objects ,Exception hierarchy ,Try catch finally ,Throw, throws
UNIT 4	**IO package:** Input streams ,Output streams ,Object serialization ,De serialization ,Sample programs on IO files ,Filter and pipe streams
UNIT 5	**Multi threading:** Thread Life cycle ,Multi threading advantages and issues ,Simple thread program ,Thread synchronization .GUI: Introduction to AWT programming, Layout and component managers ,Event handling ,Applet class ,Applet life-cycle ,Passing parameters embedding in HTML ,Swing components – JApplet, JButton, JFrame, etc. Sample swing programs

Course Outcome:

At the end of this course student will:

CO1:Understand how object-oriented concepts are incorporated into the Java programming language
CO2: Develop problem-solving and programming skills using OOP concept
CO3:Understand the benefits of a well structured program
CO4:Develop the ability to solve real-world problems through software development in high-level programming language like Java
CO5:Develop efficient Java applets,threading and applications using OOP concept

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1		M		M	H				M		H		M	H	
CO2	L		M		H		L	L		M		M		H	M
CO3		M		H	M	L		L		M	H		M	H	
CO4			H	M			L		M		H		M	H	
CO5			H	M		L						M	H	H	

H = Highly Related; M = Medium L = Low

References:

1. Programming with Java A Primer, E.Balaguruswamy Tata McGraw Hill Companies

2. Java Programming John P. Flynt Thomson 2nd

3. Java Programming Language Ken Arnold Pearson

4. The complete reference JAVA2, Herbert schildt. TMH

BCO 068A	Programming in Java Lab	0-0-2

LIST OF EXPERIMENTS

1. **Operators and Expressions**
   a. To write a java program to find the area of rectangle
   b. To write a java program to find the result of the following expressions
      i) (a<<2) + (b>>2)
      ii) (b>0)
      iii) (a+b*100) /10
      iv) a & b
      Assume a=10 and b=5
   c. To write a java program to print the individual digits of a 3 digit number.

2. Decision Making Statements

a. write a java program to read two integers and print the larger number followed by the words "is larger "If the numbers are equal print the message "These numbers are equal".

b. To write a java program to read an integer and find whether the number is odd or eve

c.To write a java program find the biggest of three integers.

3. Looping Statements

a. To write a java program to find the sum of digits of a given number

b. To write a java program to find the first 15 terms of Fibonacci sequence.

c. To write a java program to print the Armstrong numbers.

4. Array

a. To write a java program to find the largest and smallest number in an array.

5. Strings

a. To write a java program that creates a string object and initializes it with your name and performs the following operations

i) To find the length of the string object using appropriate String method.

ii) To find whether the character 'a' is present in the string. If yes find the number of times 'a' appear in the name and the location where it appears

6. String Buffer

a. To write a java program to create a StringBuffer object and illustrate how to append characters and to display the capacity and length of the string buffer

b. To write a java program to create a StringBuffer object and illustrate how to insert characters at the beginning

c. To write a java program to Create a StringBuffer object and illustrate the operations of the append () and reverse () methods.

7. Classes and Objects

a. To write a java program to display total marks of 5 students using student class. Given the following attributes: Regno(int), Name(string), Marks in subjects(Integer Array), Total (int).

b. To write a program in java with a class Rectangle with the data fields width, length, area and colour. The length, width and area are of double type and colour is of string type.The methods are get_length(), get_width(), get_colour() and find_area().

Create two objects of Rectangle and compare their area and colour. If the area and colour both are the same for the objects then display "Matching Rectangles", otherwise display "Non-matching Rectangle".

8. Inheritance

a. write a java program to create a Player class and inherit three classes Cricket_Player,

Football_Palyer and Hockey_Player.

9. Interfaces

a. To write a java program to show how a class implements two interfaces.

b. To write a java program to show that the variables in an interface are implicitly static and final and methods are automatically public

10. Packages

a. To write a java program to create a package for Book details giving Book name, Author name, price and year of publishing.

11. Applets & AWT

a. To write a java applet program to change the color of a rectangle using scroll bars to change the value of red, green and blue

b. To write an applet program for creating a simple calculator to perform Addition, subtraction, Multiplication and Division using Button, Label and TextField component.

12. Exception Handling

a. To write a java program to catch more than two exception

b. To write a java program to create our exception subclass that throws exception if the sum of two integers is greater than 99.

13. Multithreading

a. To write a java program for generating two threads, one for generating even number and one for generating odd number.

Course Outcome:

At the end of this course student will:

CO1:Understand how object-oriented concepts are incorporated into the Java programming language

CO2: Develop problem-solving and programming skills using OOP concept

CO3:Understand the benefits of a well structured program

CO4:Develop the ability to solve real-world problems through software development in high-level programming language like Java

CO5:Develop efficient Java applets,threading and applications using OOP concept

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1		M		M	H				M		H		M	H	
CO2	L		M		H		L	L		M		M		H	M
CO3		M		H	M	L		L		M	H		M	H	
CO4			H	M			L		M		H		M	H	
CO5			H	M		L						M	H	H	

H = Highly Related; M = Medium L = Low

BCO 036A	**Programming in .NET**	**3:0:0 [3]**

Objectives

To enable students to understand

- .NET framework and its runtime environment
- Major aspects of C# language
- Object oriented features such as classes, inheritance, interfaces and polymorphism
- New features that are unique to c# such as properties, indexers, delegates, events and namespaces

UNIT 1	Introduction to Web Programming, Client/Server Technology, Understanding Web Server IIS.NET Platform: Explore NET Framework 4.5, Understand Common Language Runtime, Understand the role of CTS and CLS, Learn about Base Class Libraries, Explain the Difference between Managed Code and Unmanaged Code, Start exploring IDE-Visual Studio 2012
UNIT 2	ASP.NET 4.5: .NET Framework, The Common Language Runtime, The Framework Class Library, Garbage Collection, MSIL, Types of Websites, Intrinsic Objects in ASP.net
UNIT 3	C# Language Syntax: Working with Console Class (Input and Output), Learn how to create C# Data Type and Variable, Learn about Data Type Conversions, Working with Operators, Creating Conditional and Looping Constructs, Implementing Methods and Parameters, Explain the Difference between Value Types and Reference Types, Learn how to implement String Handling
UNIT 4	Classes & Objects: Classes and Objects, Partial Classes, Methods, Properties and Events, Constructors, Property Procedures, Enumerations, Reference vs. Value Types, Structures, Namespaces: Dynamic IL and Dynamic Language Runtime, Abstract Classes and Interfaces, The Exception Handling in .Net 4.5
UNIT 5	Arrays & Collections: Arrays, Resizing Arrays, Array Lists & Hash Tables, Generic Collections Working with Windows Form, Working with windows controls: Web Form- Web Control Class, Creating Web Forms Application, Handling Images, Navigating between Pages, Managing Server Controls, Server Control Events, Using Data Controls, Navigation Controls:-Tree View Control Menu, Control Site Map Path, Control Wizard Control ,Validation Controls:- ASP.net validation controls, Configuring validation controls,

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

H = Highly Related; M = Medium L = Low

Text Books:

1. Sathiaseelan J. G. R, Sasikaladevi N, Programming with C# .NET PHI Learning
2. Kogent Learning Solutions Inc., .NET 4.5 Programming (6-in-1) Dreamtech Press (2013)

Reference Books:

Julia Case Bradley, Anita Millspaugh, Programming in Visual Basic .NET, Tata McGraw - Hill Education

Program Elective-II (V Semester)

BCO 041A	**Multimedia Computing**	**4:0:0 [3]**

Objective:

- To present a step-by-step approach to multimedia systems design.
- To introduce multimedia standards.
- To introduce compression and decompression technologies.
- To provide a detailed analysis of the various storage technologies.

UNIT 1	Introduction to Multimedia, Media and Data Streams: Medium, Main Properties of a Multimedia System, Traditional Data Streams Characteristics, Data Stream Characteristics for Continuous Media, Information Units; Sound/Audio: Basic Sound Concepts, MIDI, Elements of Speech, Speech Generation, Speech Analysis, Speech Transmission;
UNIT 2	Images andGraphics: Digital Image Representation, Image Format, Graphic Format, Image Synthesis, Image Analysis, Image Transmission; Video and Animation: Basic Concepts, Television: Conventional Systems, Enhanced Definition System, High Definition System, Transmission , Computer Based Animation: Animation Languages, Method of Controlling Animation Display of Animation, Transmission of Animation;
UNIT 3	Data Compression: Storage Space, Coding Requirement, Source, Entropy and Hybrid Encoding, Basic Compression Techniques, Lossy Sequential DCT-based Mode, Expanded Lossy DCT-based Mode, Lossless Mode, Hierarchical Mode, H.261, MPEG, DVI; Computer Technology: Communication Architecture, Hybrid Systems, Digital Systems;
UNIT 4	Multimedia Operating Systems: Real Time and Multimedia, Resource Management, Real Time Process Management in Conventional Operating Systems, Real Time Processing Requirement, Traditional Real Time Scheduling, Earliest Deadline First Algorithm, Rate Monotonic Algorithm, EDF and Rate Monotonic: Context Switches, EDF and Rate Monotonic: Processor Utilization, Preemptive versus Non-preemptive Task Scheduling, Scheduling of Continuous Media Tasks, Traditional File System: Disk Scheduling: Shortest-Seek-Time First, SCAN, C-SCAN, Multimedia File System: Disk Scheduling Algorithm;
UNIT 5	Synchronization: Basic Synchronization Issues, Intra and Inter-object Synchronization, Live and Synthetic Synchronization, Lip Synchronization Requirement, Pointer Synchronization Requirement, Elementary Media Synchronization, The Synchronization Reference Model, Multimedia Synchronization Specification Methods, Interval-based Specifications, Axes-based Specification, Control Flow-based Specification, Event-based Synchronization;

Course Outcome(CO):

At the end of this course studentswill have:

CO1-Students will be capable of understanding different realizations of multimedia tools and their usage.

CO2-Students will be capable of implementing various multimedia standards

CO3-Student will be capable of implementing different compression and decompression technologies.

CO4-Students will be capable of analyzing various storage technologies

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1			L			H									
CO2	H						H					M	M		
CO3		H	H	M	H				H	L	H	H	H	H	H
CO4	M							L							

BCO 042A	Information Retrieval	3:0:0 [3]

Objective:

- To provide the foundation knowledge in information retrieval.
- To equip students with sound skills to solve computational search problems.
- To appreciate how to evaluate search engines.
- To appreciate the different applications of information retrieval techniques in the Internet or Web environment.
- To provide hands-on experience in building search engines and/or hands-on experience in evaluating search engines.

UNIT 1	Introduction- Basics of Information Retrieval and Introduction to Search Engines; Boolean Retrieval-: Boolean queries, Building simple indexes, Processing Boolean queries.
UNIT 2	Term Vocabulary and Posting Lists**-** Choosing document units, Selection of terms, Stop word elimination, Stemming and lemmatization, Skip lists, Positional postings and Phrase queries; Dictionaries and Tolerant Retrieval: Data structures for dictionaries, Wildcard queries, Permuterm and K-gram indexes, Spelling correction, Phonetic correction.
UNIT 3	Index Construction- Single pass scheme, Distributed indexing, Map Reduce, Dynamic indexing; Index Compression - Statistical properties of terms, Zipf's law, Heap's law, Dictionary compression, Postings file compression, Variable byte codes, Gamma codes.
UNIT 4	Vector Space Model- Parametric and zone indexes, Learning weights, Term frequency and weighting, Tf-Idf weighting, Vector space model for scoring, variant tf-idf functions.
UNIT 5	Computing Scores in a Complete Search System- Efficient scoring and ranking, Inexact retrieval, Champion lists, Impact ordering, Cluster pruning, Tiered indexes, Query term proximity, Vector space scoring and query operations

Outcomes:

Upon completion of the subject, students will be able to:

CO1:Understand and apply the basic concepts of information retrieval.
CO2:Appreciate the limitations of different information retrieval techniques.
CO3:Write programs to implement search engines.
CO4:Evaluate search engines.
CO5: appreciate the different applications of information retrieval techniques in the Internet or Web environment.

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1		L	L								L		M	L	
CO2		L	H		L			L		L	M		M	M	
CO3		M	H								M		H	M	
CO4		M	M								M			M	
CO5		L	M	L							L	L	M	H	

Text Books:

1.C. D. Manning, P. Raghavan, and H. Schutze, *An Introduction to Information Retrieval,* Cambridge University Press, 2009.

Reference Books:

1. 1.R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval, Pearson Education*, 1999.

BCO 043A	Software Architecture	4:0:0 [4]

Objectives

- To enable students to understand the challenges of advanced software design and the issues associated with large-scale software architectures, frameworks, patterns and components.
- To develop the students' understanding of the tools and techniques that may be used for the automatic analysis and evaluation of software

UNIT 1	Introduction to Software architecture – Architectural styles – pipes and filters – data abstraction and object oriented organization – Event based, Implicit invocation, Layered systems – Repositories – Interpreters – Process control – Heterogeneous Architectures.
UNIT 2	Shared Information Systems – Integration in software Development Environment – Integration in the design of Buildings – Architectural structures for shared information systems.
UNIT 3	Guidance for user interface architecture Quantified design space – Formal models and specifications-The value of architectural formalism – Formalizing the architecture of a specific system – Formalizing the architectural style – Formalizing an architectural design space
UNIT 4	Linguistic issues - Requirements for architecture – Description languages – first class connectors – Adding implicit invocation to factorial processing languages.
UNIT 5	Tools for architectural design – Unicon – Exploiting style in architectural design environments – Architectural interconnection ; ADL – Languages for describing architectures.

Course Outcome (CO):

At the end of this course students will have:

CO1: An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.
CO2: An ability to identify, formulates, and solve engineering problems.
CO3: Understand some of the challenging design issues that software engineers face and the trade-offs associated with the solutions to these.
CO4: Understand the principles behind software patterns and be able to apply a number of the fundamental patterns

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1			H	M						L			H		
CO2	H	H			H				L				M	H	M
CO3								H				M			H
CO4	M					H	H				L			L	

Text Books:

1. Software Architecture – perspectives on an emerging discipline - Mary Shaw, David Garlan, PHI (1996)

Reference Books:

1. Hong Zhu, "Software Design Methodology – From principles to Architectural styles", Elsevier, 2006.
2. David Budgen, "Software Design", Pearson Education, 2004

3.Mary Shaw and David Garlan, "Software Architecture – Perspectives on an emerging Discipline", PHI, 2003

BCO 044A	HIGH SPEED NETWORKS	4:0:0 [4]

OBJECTIVES

- To learn High speed networks, Traffic and congestion management
- To understand resource allocation and service management approaches
- To study wireless network operations and functions
- To learn network management and its protocols

UNIT 1	ISDN and Frame Relay: Introduction to High Speed networks - Basics: OSI/ISO reference model- ISDN: Conceptual view – Standards – Transmission structure – BISDN - Frame Relay: Frame mode protocol architecture – Call control – LAPF – Congestion – Traffic rate management –Explicit congestion avoidance – Implicit congestion control.
UNIT 2	Asynchronous Transfer Mode: Asynchronous transfer mode - ATM Protocol architecture, ATM logical Connection, ATM Cell - ATM Service Categories – AAL - Traffic and Congestion controlling ATM - Requirements - Attributes - Traffic Management Frame work, Traffic Control – ABR traffic Management - ABR rate control, RM cell formats, ABR Capacity allocations - GFR traffic management.
UNIT 3	Congestion Control and QOS in IP Networks: Congestion Control in Packet Switching Networks: – The Need for Flow and Error Control – Link Control Mechanisms – ARQ Performance – TCP Flow Control – TCP Congestion Control – Performance of TCP Over ATM –Integrated Services Architecture – Queuing Discipline – Random Early Detection – DifferentiatedServices – Resource Reservation : RSVP – Multi protocol Label Switching – Real Time Transport Protocol.
UNIT 4	WDM Optical Networks: Introduction to Optical Networks – Wavelength Division Multiplexing(WDM) – Introduction to broadcast and select networks – switch architectures – channel accessing– Wavelength routed networks – switch architectures – Routing and wavelength assignment –Virtual topology design – IP over ATM over WDM – IP over WDM.
UNIT 5	SONET and SDH: High Speed LAN's: Fast Ethernet – Switched fast Ethernet - Gigabit Ethernet– FDDI: Network configuration – Physical Interface – Frame transmission and reception –SONET: Introduction – Layers – Frames – STS multiplexing – SONET networks – Virtual tributaries - Payload mappings – Packet over SONET – Generic Framing Procedure – Transport services – SONET over WDM – Traffic Grooming.

Outcome:

After completion of this syllabus. Students will be able to:

- Familiarize with high speed networks, Traffic and congestion management

- Understand the resource allocation and service management approaches
- Learn the use of wireless network operations and functions
- Learn network management and its protocols

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

H = Highly Related; M = Medium L = Low

TEXT BOOKS

1. William Stallings, "ISDN and Broadband ISDN with Frame Relay and ATM", Prentice-Hall of India, Fourth edition, 2004.
2. William Stallings, "High Speed Networks and Internets", Pearson Education, Secondedition, 2002.

REFERENCES

1. Greg Bemstein, Bala Rajagopalan and Debanjan Saha, "Optical Network Control – Architecture, Protocols and Standards", Pearson Education, 2004.
2. Behrouz A Forouzan, "Data Communications and Networking", Tata McGraw-Hill, Fourthedition, 2006.
3. Behrouz A. Forouzan and Sophia Chung Fegan, "Local Area Networks", Tata McGraw-Hill, 2001.
4. Rajiv Ramaswami and Kumar N. Sivarajan, "Optical Networks: A Practical Perspective", Morgan Kaufmann, Second edition (Elsevier Indian Edition), 2004. (UNITs IV & V)
5. Uless Black, "Optical Networks - Third Generation Transport Systems", Pearson Education, 2002.
6. C. Siva Ram Murthy and Mohan Gurusamy, "WDM Optical Networks: Concepts, Designand Algorithms", Prentice-Hall of India, 2002.
7. Fred Halsall, "Multimedia Communications – Applications, Networks, Protocols", PearsonEdition, 2001.

BCO 045A	OPERATIONS RESEARACH	4:0:0 [4]

Objective:

- Students will get an introduction to use quantities methods and techniques for effective decisions making;
- Students will get introduction about model formulation and applications that are used in solving business decision problems.

UNIT 1	Introduction: Definition and scope of operations research (OR), OR model, solving the OR model, art of modeling, phases of OR study. Linear Programming: Two variable Linear Programming model and Graphical method of solution, Simplex method, Dual Simplex method, special cases of Linear Programming, duality, sensitivity analysis.
UNIT 2	Transportation Problems: Types of transportation problems, mathemataical models , transportation algorithms, Assignment: Allocation and assignment problems and models, processing of job through machines
UNIT 3	Network Teachniques: Shortest path model, minimum spanning Tree Problem, Max-Flow problem and Min-cost problem. Project Management: Phases of project management, guidelines for network construction, CPM and PERT.
UNIT 4	Theory of Games: Rectangular games, Minimax theorem, graphical solution of 2 x n or m x 2 games, game with mixed strategies, reduction to linear programming model. Quality Systems: Elements of Queuing model, generalized poisson queing model, single server models.
UNIT 5	Inventory Control: Models of inventory, operation of inventory system, quantity discount. Replacement: Replacement models: Equipments that deteriorate with time, equipments that fail with time

Outcomes:

- After completion of this syllabus student will be able to understand the characteristics of different types of decision-making environments and the appropriate decision making approaches and tools to be used in each type.
- After completion of this syllabus student will be able to build and solve Transportation Models and Assignment Models.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															

CO3															
CO4															
CO5															

- H = Highly Related; M = Medium L = Low

Text / Reference Books:

1. Wayne L. Winston,"Operations Research" Thomson Learning, 2003.

2. Hamdy H. Taha, "Operations Research-An Introduction" Pearson Education,2003.

3. R. Panneer Seevam, "Operations Research" PHI Learning, 2008.

4. V.K.Khanna, "Total Quality Management" New Age International, 2008.

BCO 046A	Simulation and Modelling	4:0:0 [4]

Objective:

1. To familiar basic concepts in modeling and simulation (M&S)
2. To understand various simulation models and give practical examples for each category
3. To construct a model for a given set of data and motivate its validity
4. To generate and test random number variates and apply them to develop simulation models.

UNIT1	Introduction**-** When simulation is appropriate and when not, advantages and disadvantages of simulation, application areas in communication, computer and software design, systems and systems environment, components of a system, discrete and continuous systems, model of a system, types of models, discrete-event simulation, steps in a simulation study. Simulation Examples- Simulation of queueing systems, on-demand and inventory systems, simulation for reliability analysis etc
UNIT 2	General Principles**-** Concepts in discrete event simulation: event scheduling/time advance algorithms, world views. List Processing: properties and operations, data structures and dynamic allocation, techniques;
UNIT 3	Simulation Software- Integrated environments. Examples and review of some existing software popular and useful in the industry, e.g., Arena, Auto Mod, Extend, Flex sim, Micro Saint, ProModel, Quest, SIMUL8, WITNESS etc. Simulation using languages and environments like C++/Java/GPSS/SSF etc. Experimentation and Statistical-Analysis Tools: common features and relevant current products.
UNIT 4	Statistical Models in Simulation- Terms and concepts. Statistical Models. Review of discrete and continuous distributions. Review of Poisson (stationary and non-stationary) processes. Empirical Distributions; Elementary Queueing Theory- Basic Structure of Queueing Models.Input Source (Calling Population).Queue, Queue Discipline, Service Mechanisms. Notations and relationships between *L, W, Lq,* and *Wq*. Little's Formula. Role of Exponential Distribution and Properties. Birth and Death Processes. M/M/s queues. Finite queue variation in M/M/s/K models with different s values. Finite Calling Population cases. Queueing Models involving Non-Exponential Distributions: M/G/1, M/D/s, M/Ek/s (involving Erlang distribution), Models without a Poisson Input, Models involving hyperexponential distributions, Priority Discipline Queueing Models: Preemptive and Non-Preemptive with results, properties and server number variations, Queueing Networks: Equivalence Property. Infinite Queues in Series and Product Form Solutions. Jackson Networks,
UNIT 5:	Application of Queueing Models- Review of Characteristics (calling population

	system capacity, arrival processes, behavior and disciplines, service times and mechanisms etc) and notations, Application of Long-Run Measures of Performance: Time average in system, average time spent per customer, Little's Formula and server utilization, costs. Steady State behaviour of Infinite (M/G/1, M/M/c/infinity, M/M/c/N/infinity) and finite (M/M/c/K/K) Calling Population Models, Use of Network of Queues.

Course Outcome (CO) of **Simulation and Modeling**

At the end of this course students will have:

CO1: To provide an in-depth understanding of the terminology of modeling and simulation
CO2: To equip our students with technical concept of simulation, simulation models and gives practical examples for each category
CO3: Develop a modeling strategy for a real world engineering system, which considers prediction and evaluation against design criteria, and integrates any required sub-system models.
CO4: Interpret the simulation results of an engineering system model, within the context of its capabilities and limitations, to address critical issues in an engineering project

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H												H		
CO2	L			H											M
CO3			H		M				M	M				L	
CO4				M		L									

H = Highly Related; M = Medium L = Low

Text Books:

1.Jerry Banks, John S. Carson II, Barry L. Nelson and David M. Nicol, *Discrete-Event System and Simulation,*Prentice Hall of India, New Delhi, 2005
2.Averill M. Law, *Simulation modeling and analysi (SIE),* Tata McGraw Hill India, 20073.David Cloud, Larry Rainey, A*pplied Modeling and Simulation,* Tata McGraw Hill, India.

Reference Books:

1.Gabriel A. Wainer, Discrete-event modeling and simulation: a practitioner's approach, CRC Press, 2009.

2. Bernard P. Zeigler, Herbert Praehofer, Tag Gon Kim, Theory of modeling and simulation: integrating discrete event and continuous complex dynamic systems, Academic Press, 2000.
3. Averill M. Law, W. David Kelton, Simulation modeling and analysis, McGraw Hill, 2000.
4. Walter J. Karplus, George A. Bekey, Boris YakobKogan, Modeling and simulation: theory and practice, Springer, 2003.
5.StanislawRaczynski, Modeling and simulation: the computer science of illusion, Wiley, 2006.
6.MohammadSalamehObaidat, Georgios I. Papadimitriou, Applied system simulation: methodologies an application, Springer, 2003.
7.vanDijk, Nico M.; Boucherie, Richard J. (Eds.) 2011. Queueing Networks: A Fundemental Approach. 798 p. 148 illus. Springer.
8.Bhat,U.Narayan, An Introduction to Queueing Theory: Modeling and Analysis in Applications, Springer 2008 (Birkhäuser Boston).
9.James J. Nutaro, *Building software for simulation: theory and algorithms, with applications in C++*. Wiley, 2010.

BCO 092A	Robotic Process Automation	4:0:0 [4]

Objective:

Unit 1	**Programming Basics & Recap:**Programming Concepts Basics – 1:Understanding the application,Basic Web Concepts,Protocols, EmailClients, DataStructures, DataTables,Algorithms,Software **Processes,SoftwareDesign,SDLC.:**Programming Concepts Basics – 2 : Scripting,.Net Framework,.Net ,Fundamentals,XML,Control structures and functions,XML,HTML,CSS,Variables& Arguments **RPA Basics** :History of Automation,What is RPA,RPA vs Automation,Processes & Flowcharts,Programming Constructs in RPA,What Processes can be Automated,Types of Bots,Workloads which can be automated
Unit 2	**RPA Advanced Concepts** :Standardization of processes,RPA Developemt methodologies,Difference from SDLC,Robotic control flow architecture,RPA business case, RPA Team ,Proccess Design Document/Solution Design Document, Industries best suited for RPA,Risks& Challenges with RPA,RPA and emerging ecosystem **Installation** :Installing Studio community edition,The User Interface,Keyboard Shortcuts,AboutUpdating,About Automation Projects,Introduction to Automation Debugging,Managing Activation Packages,Reusing Automations Library, Installing the Chrome Extension,Installing the Firefox Extension,Connecting your project to a source control system,Activities Guide **Variables**:Managing Variables,Naming Best Practices,The Variables Panel, GenericValue Variables,TextVariables,True or False Variables,NumberVariables, ArrayVariables,Date and Time Variables,Data Table Variables,Managing Arguments,Naming Best Practices,The Arguments Panel,UsingArguments,About Imported Namespaces,Importing New Namespaces
Unit 3	**Control Flow** :Control Flow Introduction,If Else Statements,Loops,Advanced Control Flow,Sequences,Flowcharts,About Control Flow,Control Flow Activities,The Assign Activity,The Delay Activity,The Do While Activity,The If Activity,The Switch Activity,The While Activity,The For Each Activity,The Break Activity **Data Manipulation** :Data Manipulation Introduction,Scalar variables, collections and Tables,TextManipulation,DataManipulation,Gathering and Assembling Data. **Recording and Advanced UI Interaction** :Recording Introduction,Basic andDesktop Recording,WebRecording,Input/Output Methods,Screen Scraping,DataScraping, Scraping advanced techniques. **Selectors** :Selectors,Defining and Assessing Selectors,Customization, Debugging, Dynamic Selectors,Partial Selectors,RPA Challenge.

Unit 4	**Advanced Automation concepts and techniques**: Image, Text & Advanced Citrix Automation :Introduction to Image & Text ,Automation,Image based automation,Keyboard based automation,InformationRetrieval,Advanced Citrix Automation challenges,BestPractices,Using tab for Images,Starting Apps. **Excel Data Tables & PDF** :Data Tables in RPA,Excel and Data Table basics,Data Manipulation in excel,Extracting Data from PDF,Extracting a single piece of data,Anchors,Using anchors in PDF **Email Automation** :Email Automation,Incoming Email automation,Sending Email automation **Exceptional Handling & Best Practice:**Debugging and Exception Handling :Debugging Tools,Strategies for solving issues,Catching errors **Project Organization**:What is project organization ,Bestpractices ,Avoidingpit falls,Invoke Activity
Unit 5	**Introduction to Orchestrator** Orchestrator:Tenants,Authentication,Users,Roles, Robots,Environments,Queues &Transactions,Schedules **Emerging and Future Trends in IT:**Emerging and Future Trends in IT:Artificial Intelligence,MachineLearning,Agentawareness,Natural Language Processing, Computer Vision **Capstone Project** Real life case studies which can be used to apply the concepts learnt during the course. The projects shall test student's skills right from process transformation and documentation to the design and development of the actual robot.

BCO 093A	Robotic Process Automation Lab	0:0:2 [2]

Robotics Process Automation Lab

List of Experiments

1. Installation of RPA packages.
2. Variables and data types
3. Control flow
    a. Conditional Statements
    b. Iteration
4. Data Manipulation- scalar variables, collections, tables, text manipulation
5. Recording-basic, desktop and web
6. Scrapping
    a. Screen scrapping
    b. Data scrapping
7. Selectors
8. Image and text automation
9. Excel and Data tables
10. Email Automation

Open Elective (V Semester)

BCO 010B	Database Management Systems	4:0:0 [4]

OJECTIVE:

- To provide knowledge of relational model
- To learn about ER diagrams.
- To learn about Query Processing and Transaction Processing

UNIT 1	Introduction - Database Systems versus File Systems, View of Data, Data Models, database languages, Database Users and Administrators. Transaction Management, Decision Support Systems, Components of a Database management System. Distributed Processing and Client- Server Architecture. Entity-Relationship Model – Basic Concepts, Constraints,Keys, Design Issues, E-R Diagrams.
UNIT 2	Relational Model- Structures of relational databases, Integrity Constraints, Logical database Design, Tables, Views, Data Dictionary. Relational Algebra, Relational Calculus. SQL – Basic Structures, Query Handling, Embedded SQL, Open Database Connectivity (ODBC), Java Database Connectivity (JDBC), Triggers, Security and Authorization. Query by Example (QBE), User Interfaces and Tools, Forms and Graphical User Interfaces. Report Generators. Overview of Relational Query Optimization.
UNIT 3	Relational Database Design- Functional Dependencies, Multi-valued Dependencies, Normal Forms, Decomposition into Normalized Relations, Physical Database Design – File Structures. Object-Relational Databases – Nested Relations, Complex Data types, Object-Relational Features in SQL: 1999.
UNIT 4	Internet Databases- World Wide Web, Client Side Scripting and Applets, Web Servers and Sessions, Services, Server Side Scripting. XML – Structure of XML Data, XML Document Schema, XQuery, Storage of XML Data, XML Applications.
UNIT 5	Advanced Topics- Fundamental Concepts of Transaction Management, Concurrency Control, Recovery Systems, Data Analysis and OLAP. Introduction to Data Mining, Data Farming, Data Warehousing, Spatial and Geographic Databases, Temporal databases.

Course Outcome (CO):

At the ends of this course studentswill have:

CO1: Awareness of database management basics and different models that we use for database.
CO2: Design and architecture of relational model, relational algebra and SQL queries.
CO3: Implement different form of normalization.
CO4: Logical representation of internet database.
CO5: Analysis and concepts of transaction, concurrency and recovery systems.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H												H	M	
CO2			H		M				M						L
CO3				H		M							M		
CO4				M								M		L	L
CO5	M	L		H					L				M		

H = Highly Related; M = Medium L = Low

Text Books:

1. Database Systems Concepts – Korthe, TMH
2. An Introduction to Database Design – Date

Reference Books:

4. Fundamentals of Database Systems – Elmasri and Navathe
5. Database Management and Design – Hansen and Hansen .

Object-Oriented Database Design – Harrington

BCO 059A	Data Mining	4:0:0 [4]

Objectives

- To introduce students to the basic concepts and techniques of Data Mining.
- To develop skills of using recent data mining software for solving practical problems.
- To gain experience of doing independent study and research.

UNIT 1	Introduction: Basic concepts of data mining, including motivation and definition; different types of data repositories; data mining functionalities; concept of interesting patterns; data mining tasks; current trends, major issues and ethics in data mining
UNIT 2	Data Types of data and data quality; Data Preprocessing: data cleaning, data integration and transformation, data reduction, discretization and concept hierarchy generation; Exploring Data: summary statistics, visualization, multidimensional data analysis
UNIT 3	Association and Correlation Analysis: Basic concepts: frequent patterns, association rules - support and confidence; Frequentitemset generation - Apriori algorithm, FP-Growth algorithm; Rule generation, Applications of Association rules; Correlation analysis.
UNIT 4	Clustering Algorithms and Cluster Analysis: Concept of clustering, measures of similarity, Clustering algorithms: Partitioning methods - k-means and k-medoids, CLARANS, Hierarchical methods - agglomerative and divisive clustering, BIRCH, Density based methods - Subspace clustering, DBSCAN; Graph-based clustering - MST clustering; Cluster evaluation; Outlier detection and analysis.
UNIT 5	Classification: Binary Classification - Basic concepts, Bayes theorem and Naïve Bayes classifier, Association based classification, Rule based classifiers, Nearest neighbor classifiers, Decision Trees, Random Forest; Perceptrons; Multi-category classification; Model over fitting, Evaluation of classifier performance - cross validation, ROC curves.

Outcomes:

Upon end of this course, Students will be able to

- Implementation and use of the basic concepts and techniques of Data Mining.
- Developskills of using recent data mining software for solving practical problems.
- Expandexperience of doing independent study and research.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

Related; M = Medium L = Low

Text Books:

1. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, Introduction to Data Mining. Pearson (2005), India. ISBN 978-8131714720

2. Jiawei Han and MichelineKamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, 3rd edition (July 2011). 744 pages. ISBN 978-0123814791

Reference Books:

1. T. Hastie, R. Tibshirani and J. H. Friedman, The Elements of Statistical Learning, Data Mining, Inference, and Prediction.Springer, 2nd Edition, 2009.768 pages. ISBN 978-0387848570
2. C. M. Bishop, Pattern Recognition and Machine Learning. Springer, 1st edition, 2006. 738 pages. ISBN 978- 0387310732
3. Ian H. Witten and Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann, 3rd edition (January 2011). 664 pages. ISBN 978-0123748560.

BCO 060A	Object Oriented Databases	4:0:0 [4]

Objective:

The course will enable the students to:

- Understand the fundamental concepts of object oriented programming and database management. These concepts include aspects of object oriented concept in database design.
- Understand the role of the DBMS & RDBMS in the organization.
- Understand object oriented in query language.

UNIT 1	**Overview of Object Oriented Concepts:** Need for Object Oriented Programming: Procedural Languages, The Object Oriented Approach, Advantages of Object Oriented Programming. Characteristics of Object Oriented Languages : Objects, Classes, Inheritance, Reusability, New Data Types, Polymorphism and OverLoading
UNIT 2	**Object oriented Data Model:** OO Relationships, Relationship integrity, ER Diagramming models for OO Relationships - different notations (Coad/Yurdan notation, Shlaer/Meelor notation, OMT notation, UML notation and Booch Notation), Integrating Objects into a Relational Database.
UNIT 3	**Object orientation in Query Languages:** Introduction to Object Definition Language (ODL) – Class declarations, attributes in ODL, Relationships in ODL, Inverse relationships, Multiplicity of relationships, methods and types in ODL. Additional ODL concepts: Multi-way relationships in ODL, sub- classes in ODL, multiple inheritances in ODL, extents, declaring keys in ODL. From ODL to Relational Designs , Object relational model –from relations to object relations, Nested relations, references, OO v/s object relational, from ODL Introduction to OQL – features of OQL, additional forms of OQL expressions, object Assignment and creation in OQL, user defined types in SQL, operations on object-relational data, Ordering relationships on UDTs
UNIT 4	**Object Oriented Database Systems** : (including Object Relational Database Systems):Relational v/s Object Oriented Database Systems : Semantic Database Systems, Object Hierarchies - reneralization, Specialization, Aggregation, E-R model, RM/T, SDM, SAM, Duplex, IFO. The architecture of Object Oriented Databases, Query Languages for OO Databases, Gemstone/O2/Orion/Objectstone, Object Relational Database Management System (ORDBMS) - Oracle 8i, 9i, DB2. Overview of object database systems: ORDBMS implementation and challenges, database design for an ORDBMS, OODBMS – ODMG data model and ODL, comparison of RDBMS,OODBMS and ORDBMS.
UNIT 5	Object Database standards: Basics of OODBMS terminology, understanding of types, inheritance, representing logical Relationships, basic interface and class structure, declaring attributes, specifying relationships, Adding operator signatures and the complete schema.

Outcome:

After completion this syllabus student will get

- Detail knowledge about database with object oriented concept like class inheritance and reusability.
- Detail knowledge about relational and object oriented database.
- Detail knowledge about Object orientation in Query language.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

H = Highly Related; M = Medium L = Low

Text Books:

1. Jan L Harrington : Object oriented Database Design clearly explained , Morgan Kaufman publishers, academic press , 2000. (chapters – 3, 4, 5 , 6,7,8,9,10)

2. Ramakrishnan and Gehrke: Database Management Systems – Third edition – International edition, Mc-graw Hill, 2003. (chapter 23 only)

Reference books:

1. R. Cattel, "Object Data management", (1993), Addison-Wesely.

2. W. Kim, "Modern Database Systems", (1995), ACM Press, Addison-Wesely.

3. CSR Prabhu, "Object Oriented Databases Systems : Approaches and Architectures" (1999),Prentice Hall of India.

4. H Garcia –Molina , J D Ullman and J Widom: Database Systems The complete book , Pearson Education, 2004.(chapter 4 and 9 only)

BCO 061A	Web Designing Techniques	4:0:0 [4]

Objective:

At the end of the course, the student should be able to:

- To gain the skills and project-based experience needed for entry into web design and development careers.
- Touse a variety of strategies and tools to create websites.
- To develop awareness and appreciation of the myriad ways that people access the web and will be able to create standards-based websites that are accessible and usable by a full spectrum of users.

UNIT 1	Computer network, uses of computer networks, network hardware, network protocol, Reference models: The OSI reference model, the TCP/IP Reference model, a comparison of the OSI and TCP/IP reference models. Introduction of Ethernet, Hub, Client Server Architecture, Switch, modem.
UNIT 2	**The World Wide Web (WWW): HTML History**, Hypertext and Hypertext Markup Language.**HTML Documents**: Tags, Elements of an HTML Document: Text Elements, Tag Elements, Structural elements of HTML documents, Header tags, Body tags, Paragraphs, Title. **List:** Numbered list, Non-Numbered lists, Definition lists
UNIT 3	**Formatting HTML Documents:** Logical styles (source code, text enhancements, variables), Physical Styles (Bold, Italic, underlined, crossed), **Managing images in html**: Image format (quality, size, type), Importing images (scanners), Tags used to insert images.**Frames** Tables in HTML documents: Tags used in table definition, Tags used for border thickness, Tags used for cell spacing, Tags used for table size, Dividing table with lines, Dividing lines with cells, Cell types: Titles cells, Data cells
UNIT 4	**Hypertext and Link in HTML Documents** URL/FTP/HTTP,Types of links: Internal Links, External Link, Link Tags, Links with images and buttons, Links that send email messages Special effects in HTML documents.
UNIT 5	**Web Designing with PHP (Introduction):Orientation and First Steps:** PHP's Place in the Web World, Basic Rules of PHP Programs, Application of PHP on the internet. Advantages of PHP.

Course Outcome (CO):

At the end of this course studentswill have:

CO1: Able to understand the basics of computer network, various protocols
CO2: Ability to understand WWW and HTML language
CO3: Ability to develop projects by formatting HTML documents &managing images in HTML
CO4: Able to understand Hypertext and Link in HTML
CO5: Ability to understand PHP programming language

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1		H		H		L		M			H		H		
CO2	H			M			L			H				M	
CO3			M		H				M			L	L		H
CO4	H	M		L								L	M	L	
CO5	M			H		L				M				H	

TEXT BOOKS

1. Deitel H.M. and Deitel P.J., “Internet and World Wide Web How to program”, Pearson International, 2012, 4th Edition.
2. Gopalan N.P. and Akilandeswari J., “Web Technology”, Prentice Hall of India, 2011Paul Dietel and Harvey Deitel,”Java How to Program”, Prentice Hall of India, 8th Edition C

B.Tech CSE Semester VI

BCO 090A	INTERNET OF THINGS	2-0-0 [2]

Course Objectives:

The objective of the course is to:

1. Introduction to IoT concepts.
2. Understand IoT Market perspective.
3. Data and Knowledge Management and use of Devices in IoT Technology.
4. Understand State of the Art – IoT Architecture.
5. Real World IoT Design Constraints, Industrial Automation and Commercial Building Automation in IoT.

UNIT 1	**M2M to IoT**-The Vision-Introduction, From M2M to IoT, M2M towards IoT-the global context, A use case example, Differing Characteristics.
UNIT 2	**M2M to IoT – A Market Perspective**– Introduction, Some Definitions, M2M Value Chains, IoT Value Chains, An emerging industrial structure for IoT, The international driven global value chain and global information monopolies. **M2M to IoT-An Architectural Overview**– Building an architecture, Main design principles and needed capabilities, An IoT architecture outline, standards considerations.Sensor modules, nodes and systems.
UNIT 3	**M2M and IoT Technology Fundamentals-** Devices and gateways, Local and wide area networking, Data management, Business processes in IoT, Everything as a Service(XaaS), M2M and IoT Analytics, Knowledge Management
UNIT 4	**IoT Architecture-State of the Art –** Introduction, State of the art, **Architecture Reference Model-** Introduction, Reference Model and architecture, IoT reference Model
UNIT 5	**IoT Reference Architecture-** Introduction, Functional View, Information View, Deployment and Operational View, Other Relevant architectural views. **Real-World Design Constraints-** Introduction, Technical Design constraints-hardware is popular again, Data representation and visualization, Interaction and remote control.**Industrial Automation-** Service-oriented architecture-based device integration, SOCRADES: realizing the enterprise integrated Web of Things, IMC-AESOP: from the Web of Things to the Cloud of Things, **Commercial Building Automation-** Introduction, Case study: phase one-commercial building automation today, Case study: phase two-commercial building automation in the future..

Course Outcome (CO) of **Internet of Things**

At the end of this course students will have:

CO1: To provide the basic understanding of IoT concepts
CO2: To equip our students with the market perspective of IoT and have the knowledge of architectural overview of IoT.

CO3:To be familiar with contemporary issues in IoT and Data and Knowledge Management and use of Devices in IoT Technology.

CO4: To be familiar with IoT tools and to enhance analytical skills to develop innovative solutions, automation.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	H											L		
CO2				H				L		M					M
CO3			M			M								L	M
CO4				H	M				M				M		

Textbook:

- Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, **"From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence",** 1st Edition, Academic Press, 2014.

Reference Books:

- Vijay Madisetti and Arshdeep Bahga, **"Internet of Things (A Hands-on-Approach)",** 1stEdition, VPT, 2014.
- Francis daCosta, **"Rethinking the Internet of Things: A Scalable Approach to Connecting Everything",** 1st Edition, Apress Publications, 2013

BCO 023A	DESIGN AND ANALYSIS OF ALGORITHMS	3-0-0 [3]

OBJECTIVES:

At the end of the course, the student should be able to:

- Design effective, efficient, elegant, and readable algorithms for various classes of computing problems
- Determine space and time complexity of algorithms by the use various algorithm design techniques like (divide and conquer, backtracking, greedy, etc.)

UNIT 1	Introduction, algorithms specification, time and space complexity, performance analysis, recurrence relations. Divide and Conquer – finding max min.
UNIT 2	Dynamic Programming and Greedy Methods – Huffman tree construction, Knapsack problem, 0/1 Knapsack problem,least common subsequence**,** matrix chain multiplication. Backtrack: 4-queen problem, Branch and Bound: assignment problem
UNIT 3	Graph algorithms–flow problems, String Matching Algorithms: Naive algorithm, automata and KMP matcher algorithms, Boyer-Moore algorithm
UNIT 4	Number Theory Problems – CRT, GCD algorithms, modular arithmetic, Lower Bound Theory; Approximate Algorithms – Set cover, vertex cover, .Randomized Algorithms – Las Vegas and Monte Carlo methods
UNIT 5	NP Completeness: Definitions of P, NP-Hard and NP-Complete Problems. Decision Problems**..**

OUTCOMES: After study of this subject student will be able to know

CO1: Various methods of calculating complexity

CO 2: Finding out thebest method for different algorithms

CO3: About computational geometry, like Lower bound theory, modular arithmetic and CRT

CO4: Various Decision Problems like NP Complete, NP hard

CO5: Knowledge of Graph and its algorithm

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	H	M		M			M				L	H	M	
CO2	H	H			H							L	H	M	M
CO3	M	H	L		M			M				M			M
CO4	H	L	M		M							L	M	H	

CO5	H	M	M		M			L				L	M	M	L

Textbooks:

1. Cormen, Leizerson&Rivest, Introduction to algorithms, Prentice-Hall. 2002
2. Horowitz &Sahni, Fundamentals of Computer Algorithms, Galgotia Publication. 1999

Reference Books:

1. Aho, HopCroft, Ullman, The Design and Analysis of Computer Algorithms, Addison-Wesley. 2001.
2. Introduction to Design and Analysis of Algorithms, Anny Levitin, Person Education Press. 2007.
3. Gilles Brassard & Paul Bratley, Fundamental Algorithms, Prentice-Hall. 1998

BCO 024A	ADVANCED COMPUTER ARCHITECTURE	3-0-0 [3]

OBJECTIVES:-

- Tocover the underlying concepts and techniques used in Advance Computer Architecture.
- To discusses principles of parallel algorithmsdesign and different parallel programming models
- To have general idea of Computer Organization. In addition, a familiaritywith Memory organization, Computational models required.

UNIT 1	Introduction - What is computer architecture? Software-hardware interface. Performance andPower.Performancemetrics.Performancemeasurement.Benchmarkprograms.
UNIT 2	Instructions- Instruction Set. Operations. Operands and addressing modes.Roleof compilers and system software.Understanding implementation of function calls andreturns, array references, pointers.
UNIT 3	Computer Arithmetic**-** Signed integers. Floating point.Rounding and accuracy.Addition and Subtraction.Multiplication.Division; Processor - Data path elements.Data pathcontrol.
UNIT 4:	Pipelining Speedup. Pipeline hazards.Stalling.Forwarding.Branchprediction.Exceptions.Speculation.Multiple issue.Dynamic scheduling; Cache memory- Locality ofreference.Cache organization and access. Multilevelcaches.Performance.Cache coherence.
UNIT 5	Virtual memory- Hardware support for address translation, page fault handling.Translation look aside buffer. Hardware-software interface.

OUTCOMES:-Upon completion of the syllabus the students will be able to know :

- Understand the concepts and techniques used in Advance Computer Architecture.
- Use of the principles of parallel algorithmsdesign and different parallel programming models.
- Familiar with general idea of Computer Organization. In addition, a familiarity with Memory organization, Computational models required

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

- H = Highly Related; M = Medium L = Low

Text Books:

1.David A. Patterson and John L. Hennessy, Computer Organization and Design: The Hardware and SoftwareInterface, Morgan Kaufmann Publishers, Fourth Edition.(2009)

Reference Books:

1.John L. Hennessy and David A. Patterson, Computer Architecture: A Quantitative Approach, MorganKaufmann Publishers (2007)

BEE 047B	EMBEDDED COMPUTING SYSTEMS	3-0-0 [3]

OBJECTIVES:-

- To understand and design embedded systems and real-time systems
- To identify the unique characteristics of real-time systems
- To explain the general structure of a real-time system
- To define the unique design problems and challenges of real-time systems
- To apply real-time systems design techniques to various software programs.

UNIT 1	Hardware Concepts -Application and characteristics of embedded systems, Overview of Processors and hardware units in an embedded system, General purpose processors, Microcontrollers:8051, Application- Specific Circuits (ASICs), ASIP, FPGA, ARM-based System on a Chip (SoC), Network on Chip (NoC), Levels of hardware modelling, Verilog, Sensors, A/D-D/A converters, Actuators
UNIT 2	Interfacing using RS-232,UART, USB, I2C, CAN bus, Flexray, SRAM and DRAM, Flash memory.
UNIT 3	Real-Time Operating Systems- Real-Time Task Scheduling: Some important concepts, Types of real-time tasks and their characteristics, Task scheduling, Clock-Driven scheduling, Hybrid schedulers, Event-Driven scheduling, Earliest Deadline First (EDF) scheduling, Rate monotonic algorithm (RMA).
UNIT 4:	Commercial Real-time operating systems: Time services, Features of a Real-time operating system, Unix-based Real-time operating systems, POSIX-RT, A survey of contemporary Real- time operating systems, Microkernelbased systems, Benchmarking real-time systems.
UNIT 5	Embedded Application Development - UML 2.0, State charts, General language characteristics, MISRA C, Hardware/Software Co- design, Hardware/software partitioning, Testing embedded systems, Design for testability and Self-test.

OUTCOMES:-

Upon completion of this course, the student will be able to:

- Understand and design embedded systems and real-time systems
- Identify the unique characteristics of real-time systems
- Explain the general structure of a real-time system
- Define the unique design problems and challenges of real-time systems
- Apply real-time systems design techniques to various software programs.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															

CO3															
CO4															
CO5															

- H = Highly Related; M = Medium L = Low

Text Books:

1. Embedded Systems Design – A Unified Hardware /Software Introduction, by Frank Vahid and Tony Givargis,John Wiley.(2001)
2. An Embedded Software Primer, by David E.Simon, Pearson Education Asia. (1999)

Reference Books:

1. Wayne Wolf, Computers as Components; Principles of Embedded Computing System Design – Harcourt India, Morgan Kaufman Publishers.(2000)

BCO 025A	DESIGN& ANALYSIS OF ALGORITHMS LAB	0-0-2

List of Experiments

1. Write a Program to Explore a Binary Heap
2. Write a Program for Merging of two search trees
3. Write a program to implement Huffman tree construction
4. Write a Program for Computing a spanning tree having smallest value of largest edge
5. Write a Program for Finding the decimal dominant in linear time
6. Write a Program for Problems on Graphs. Etc.
7. Write a program to find Greatest Common Divisor
8. Write a program for fractional Knapsack problem
9. Write a program for 0/1 Knapsack problem
10. Write a program to implement Naive algorithm,
11. Write a program to implement KMP matcher algorithms,
12. Write a program to implement Boyer-Moore algorithm
13. Write a program to implement modular arithmetic
14. Write a program to implement Set cover,
15. Write a program to implement vertex cover

OUTCOMES: After study of this subject student will be able to know
CO1: Various methods of calculating complexity
CO 2: Finding out the best method for different algorithms
CO3: About computational geometry, like Lower bound theory, modular arithmetic and CRT
CO4: Various Decision Problems like NP Complete, NP hard
CO5: Knowledge of Graph and its algorithm

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	H	M		M			M				L	H	M	
CO2	H	H			H							L	H	M	M
CO3	M	H	L		M			M				M			M
CO4	H	L	M		M							L	M	H	
CO5	H	M	M		M			L				L	M	M	L

Program Elective I (VI Sem)

BCO 037A	ADVANCE PROGRAMMING IN JAVA	4-0-0 [4]

OBJECTIVES:-Students will be able to know the following

- To learn the Java programming language: its syntax, idioms, patterns, and styles.
- To become comfortable with object oriented programming: Learn to think in objects
- To learn the essentials of the Java class library, and learn how to learn about other parts of the library when you need them.
- To introduce event driven Graphical User Interface (GUI) programming

UNIT 1	Revisited of GUI, Database Programming using JDBC Introduction to JDBC ,JDBC Drivers & Architecture CURD operation Using JDBC Connecting to non-conventional Databases. Connectivity with SQL server, Oracle and MS access.
UNIT 2	Networking , Networking Basics ,The Networking Classes and Interfaces InetAddress ,Factory Methods ,Instance Methods ,Inet4Address and Inet6Address, TCP/IP Client Sockets ,URL,URLConnection,Http URL Connection, The URI Class,Cookies, TCP/IP Server Sockets,Datagram, DatagramSocket ,DatagramPacket,
UNIT 3	RMI (Remote Method Invocation) RMI overview RMI architecture, Designing RMI application, Executing RMI application. Example demonstrating RMI
UNIT 4	Servlet: Web Application Basics. Architecture and challenges of Web Application.Introduction to servlet life cycle Developing and Deploying Servlets Exploring Deployment Descriptor (web.xml). Handling Request and Response Initializing a Servlet Accessing Database Servlet Chaining Session Tracking & Management Dealing with cookies Transferring Request Accessing Web Context Passing INIT and CONTEXT Parameter Sharing information using scope object Controlling concurrent access User Authentication Filtering Request and Response Programming Filter Filter Mapping Servlet Listeners .
UNIT 5	Basic JSP Architecture Life Cycle of JSP (Translation, compilation) JSP Tags and Expressions Role of JSP in MVC-2 JSP with Database JSP Implicit Objects Tag Libraries JSP Expression Language (EL) Using Custom Tag JSP Capabilities: Exception Handling Session Management Directives JSP with Java Bean.

OUTCOMES:-

Upon end of this course, students will be able to:

- About the Java programming language: its syntax, idioms, patterns, and styles.

- Becomecomfortable with object oriented programming: Learn to think in objects
- Learnthe essentials of the Java class library, and learn how to learn about other parts of the library when you need them.
- Introduce event driven Graphical User Interface (GUI) programming

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program Outcomes												Program specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1			H	L	H		L			M	H			H	
CO2		L	H		H	L		L					M	H	
CO3			H	M					L	L		M		H	
CO4				M	H			M			L			H	M
CO5		L			H	M			M					H	

Text Books:

1. J2EE: The complete Reference by James Keogh
2. Java 6 And J2Ee 1.5, Black Book by kogent
3. Java Server Programming Java EE6 (J2EE 1.6), Black Book by kogent

Reference books:-

1. Programming with Java A Primer, E.Balaguruswamy Tata McGraw Hill Companies
2. Java Programming John P. Flynt Thomson 2nd
3. Java Programming Language Ken Arnold Pearson

BCO 069A	Advance Programming in Java Lab	0-0-2

Exp.No.	Name of the Experiment
1	Design a registration page using HTML.
2	**Implementing JDBC** Program 2(A)Write a program by using JDBC to execute insert, select and update query by using PreparedStatement and display the results. Program 2(B) Write a program by using JDBC to execute an update query by using PreparedStatement and display the results. Program 2(C) Write a program and execute ResultSetMetaData Interface by using JDBC.
3	**Implementing Servlet** Program 3(A) Write a program and execute a simple servlet demonstrating servlet lifecycle. Program 3(B) Write a program and execute a servlet program that receives input from html page. Program 3(C) Write a program and execute ServletRequest and ServletResponse Interfaces with methods. Program 3(D) Write a program and execute HttpServlet Class doGet() and doPost() Methods. Program 3(E) Write a program to store the user information into Cookies. Write another program to display the above stored information by retrieving from Cookies.
4	**Implementing JSP,JSP Custom Tags and Directives** Program4(A) Write a program to connect HTML page,JSP page and mysql database.Program 4(B) Write a program and implement custom tags in JSP Program 4(C) Write a program and implement JSP directives.
5	**Implementing JavaBean** Program 5 Write a program and implement Javabeans using JSP page.
6	**Implementing JSP Standard ActionElements** Program 6 Write a program and implement JSP StandardActionElements.
7	**Implementing JSP Scripting Elements** Program 7:Write a program and execute JSP Scriptlets,Declarations and Expressions.
8	**Learning session management** Program 8(A):Write program and execute session management using URL rewriting Program 8(B) :Write program and execute session management using Hidden Fields.

Program 8(C): Write program and execute session management using Cookie

Program 8(D): Write a program and execute session management using Session Objects.

9 **Remote Method Invocation (RMI)**

Program 9(A):Write a program and execute Remote Method Invocation

10 **Configure web.xml**

Program 10:Write a code to deploy web.xml file

11 **Performing Client-Server Communication and Networking**

Program 11(A): WAP to implement Client-Server Program
Program 11(B): WAP to implement InetAddress.
Program 11(C): WAP for Sending Email in java

12. **Implementing Multithreading**

Program 12: WAP to implement multithreading(three threads using single run method).

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program Outcomes												Program specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1			H	L	H		L			M	H			H	
CO2		L	H		H	L		L					M	H	
CO3			H	M					L	L		M		H	
CO4				M	H			M			L			H	M
CO5		L			H	M			M					H	

12

BCO 038A	ADVANCE PROGRAMMING IN .NET	4:0:0 [4]

OBJECTIVES:-After completion of this course, the students would be able to

- Understand, analyze and explain .NET Framework and C#.
- Understand, analyze and use basic C# constructs, delegates and events.
- Understand, analyze and use language interfaces, and inheritance.
- Familiar with using .NET collections (sets, lists, dictionaries).
- Understand, analyze and exposed to the Common Language Runtime (CLR), garbage collection, and assemblies

UNIT 1	Developing Web form pages Configure web forms pages, Page directives such as View State, request validation, event validation, Master PageFile; ClientIDMode; using web.config; setting the html doctype, Implement master pages and themes, Creating and applying themes; adding multiple content placeholders; nested master pages; control skins; passing messages between master pages; switching between themes at runtime; loading themes at run time; applying a validation schema, Implement globalization, Resource files, browser files, CurrentCulture, currentUICulture, ASP:Localize, Handle page lifecycle events, IsPostback, IsValid, dynamically creating controls, control availability within the page lifecycle, accessing control values on postback, overriding page events Implement caching, Data caching; page output caching; control output caching; cache dependencies; setting cache lifetimes; substitution control Manage state Server-side technologies ,client-side technologies,configuring session state.
UNIT 2	Developing and using web form controlsValidate user input, Client side, server side and via AJAX; custom validation controls; regex validation; validation groups; datatype check; jQuery validation Create page layout, AssociatedControlID; Web parts; navigation controls; File Upload controls Implement user controls Registering a control; adding a user control; referencing a user control; dynamically loading a user control; custom event; custom properties; setting toolbox visibility Implement server controlsComposite controls, INamingContainer, adding a server control to the toolbox, global assembly cache, creating a custom control event, globally registering from web.config; TypeConverters ,Manipulate user interface controls from code-behind, HTML encoding to avoid cross-site scripting, navigating through and manipulating the , control hierarchy; FindControl; controlRenderingCompatibilityVersion; URL encoding; RenderOuterTable.
UNIT 3	Implementing client side scripting and Ajax Add dynamic features to a page by using JavaScript, Referencing client ID; script manager; scriptcombining;Page.clientscript.registerclientscriptblock;Page.clientscript.registerc lientscripting include; sys.require Alter a page dynamically by manipulating the DOM Using jQuery, adding, modifying or removing page elements, adding effects, jQuery selectors Handle JavaScript events DOM events, custom events, handling events by using jQuery Implement ASP.NET AJAX ,Client-side templating, creating a script service, extenders (ASP.NET AJAX control toolkit), interacting with the server, Microsoft AJAX client library, custom extenders; multiple update panels;

	triggers; UpdatePanel.UpdateMode; timer Implement AJAX by using jQuery $.get, $.post, $.getJSON, $.ajax, xml, html, JavaScript Object Notation (JSON), handling return types
UNIT4	Configurationg and extending a web application Configure authentication and authorization Using membership, using login controls, roles, location element, protecting an area of a site or a page Configure providers Role, membership, personalisation, aspnet_regsql.exe Create and configure HttpHandlers and HttpUNITs Generic handlers, asynchronous handlers, setting MIME types and other content headers, wiring UNITs to application events Configure initialisation and error handling HandlingApplication_Start, Session_Start and Application_BeginRequest in global.asax, capturing unhandled exceptions, custom error section of web.config, redirecting to an error, page; try and catch; creating custom exceptions Reference and configure ASMX and WCF servicesAdding service reference, adding web reference, changing endpoints, wsdl.exe, svcutil.exe; updating service URL; shared WCF contracts assembly This objective does not include: creating WCF and ASMX services Configure projects and solutions, and reference assemblies Local assemblies, shared assemblies (global assembly cache), web application projects, solutions, settings file, configuring a web application by using web.config or multiple .config files; assemblyinfoDebug a web application Remote, local, JavaScript debugging, attaching to process, logging and tracing, using local IIS, aspnet_regiis.exe Deploy a web application Pre-compilation, publishing methods (e.g., MSDeploy, xcopy, and FTP), deploying an MVC application
UNIT5	Displaying and manipulating dataImplement data-bound controls Advanced customisation of DataList, Repeater, ListView, FormsView, DetailsView, TreeView, DataPager, Chart, GridView Implement DataSource controls ObjectDataSource, LinqDataSource, XmlDataSource, SqlDataSource, QueryExtender, EntityDataSource Query and manipulate data by using LINQ Transforming data by using LINQ to create XML or JSON, LINQ to SQL, LINQ to entities, LINQ to objects, managing DataContext lifetime Create and consume a data service WCF, web service; server to server calls; JSON serialisation, XML serialization Create and configure a dynamic data project , Dynamic data controls, custom field templates; connecting to DataContext and ObjectContext

OUTCOMES:-After completion of this course, the students would be able to

- Understand programming language concepts, particularly C# and object-oriented concepts.
- Write, debug, and document well-structured .NET applications
- Implement .NET classes from specifications
- Effectively create and use objects from class libraries
- Understand the behavior of primitive data types, object references, and arrays
- Apply decision and iteration control structures to implement algorithms

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

- H = Highly Related; M = Medium L = Low

Program Elective II (VI Sem)

BCO 047A	GRAPH THEORY	4-0-0 [4]

OBJECTIVES:

- This course is aimed to cover a variety of different problems in GraphTheory.
- In this course Students will come across a number of theorems and proofs. Theorems will be stated and proved formally using various techniques.
- Various graphs algorithms will also be taught along with its Analysis. They will also know about the practical implementation of graph theory

UNIT 1	Basics– Graphs, degree sequences, distance in graphs, complete, regular and bipartite graphs, basic properties.
UNIT 2	Structure and Symmetry – Cut vertices, bridges and blocks, automorphism groups, construction problem.
UNIT 3	Trees and connectivity – Properties of trees, Arboricity, vertex and edge connectivity, Mengers theorem.
UNIT 4:	Eulerian and Hamiltonian graphs – Characterization of Eulerian graphs -Sufficient conditions for Hamiltonian graphs.
UNIT 5	Colouring and planar graphs – vertex and edge colouring, perfect graphs, planar graphs, Euler's theorem, Kuratowski's theorem, Colouring of planar graphs, Crossing number andthickness.Matching, factors, decomposition and domination.Extremal Graph theory – Turan's theorem, Ramsay's theorem, Szemeredi'sregularity lemma, applications

OUTCOMES:-

Upon completion of this course, the student will be able to:

CO1: Basic concepts of graph theory.
CO2: Apply the basic concepts of mathematical logic
CO3: Real time problems using concepts of graph theory
CO4: Knowledge of tree, properties of tree and various theorems
CO5: Understanding various theorems related graph like coloring of graph

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	L	M		H								L	H	
CO2	H	H	L		M								L	M	M
CO3	H	H	H		H							M	M	H	H
CO4	H	M	H	M	M								M	L	

CO5	H	M			M								M	L	

Text Books:

1.Graph Theory, by J. A. Bondy and U. S. R. Murthy, Springer Verlag (2008.)
2.Introduction to Graph Theory,by D. B. West, PHI, 2004.

Reference Books:

1. Graph Theory, by R. Diestel : Springer Verlag (Free Download available).(2003)

BCO 048A	REAL TIME SYSTEMS	4-0-0 [4]

OBJECTIVE:

At the end of the course, the student should be able to:

- To explain the timing requirements of real-time systems.
- To distinguish between hard and soft real-time systems.
- To discuss the defining characteristics of real-time systems.
- To describe scheduling algorithms for hard real-time system.

UNIT 1	Introduction: Definition, Typical Real Time Applications: Digital Control, High Level Controls, Signal Processing etc. Release Times, Deadlines, and Timing Constraints, Hard Real Time Systems and Soft Real Time Systems, Reference Models for Real Time Systems: Processors and Resources, Temporal Parameters of Real Time Workload, Periodic Task Model, Precedence Constraints and Data Dependency.
UNIT 2	Real Time Scheduling: Common Approaches to Real Time Scheduling: Clock Driven Approach, Weighted Round Robin Approach, Priority Driven Approach, Dynamic Versus Static Systems, Optimality of Effective-Deadline-First (EDF) and Least-Slack-Time-First (LST) Algorithms, Offline Versus Online Scheduling, Scheduling Aperiodic and Sporadic jobs in Priority Driven and Clock Driven Systems.
UNIT 3	Resources Access Control: Effect of Resource Contention and Resource Access Control (RAC), Non-preemptive Critical Sections, Basic Priority-Inheritance and Priority-Ceiling Protocols, Stack Based Priority-Ceiling Protocol, Use of Priority-Ceiling Protocol in Dynamic Priority Systems, Preemption Ceiling Protocol, Access Control in Multiple-Unit Resources, Controlling Concurrent Accesses to Data Objects.
UNIT 4	Multiprocessor System Environment: Multiprocessor and Distributed System Model, Multiprocessor Priority-Ceiling Protocol, Schedulability of Fixed-Priority End-to-End Periodic Tasks, Scheduling Algorithms for End-to-End Periodic asks, End-to-End Tasks in Heterogeneous Systems, Predictability and Validation of Dynamic Multiprocessor Systems, Scheduling of Tasks with Temporal Distance Constraints.
UNIT 5	Real Time Communication: Model of Real Time Communication, Priority-Based Service and Weighted Round- Robin Service Disciplines for Switched Networks, Medium Access Control Protocols for Broadcast Networks, Internet and Resource Reservation Protocols, Real Time Protocols, Communication in Multicomputer System, An Overview of Real Time Operating Systems.

OUTCOMES:-

Upon completion, the students will be able to know the following.

- The basics and importance of real-time systems.
- Generate a high-level analysis document based on requirements specifications.
- Generate a high-level design document based on analysis documentation.
- Generate a test plan based on requirements specification.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1	H	M											H		
CO2	H		M										M		
CO3	H		H											H	
CO4	H			L										M	
CO5	H														

- H = Highly Related; M = Medium L = Low

.

Text books.

1. C. Siva Ram Murthy and G. Manimaran, "Resource Management in Real-Time Systems and Networks", Prentice-Hall of India, 2005. (UNITs I, II, IV & V)
2. Jane W.S. Liu, "Real-Time Systems", Prentice Hall, USA, 2000. (UNIT III)

References Books:

1. Rajib Mall, "Real-Time Systems Theory and Practice", Pearson Education, India, 2007.
2. C.M. Krishna and Kang G. Shin, "Real-Time Systems", McGraw-Hill International Editions, 1997.

BCO 049A	DISTRIBUTED COMPUTING	4-0-0 [4]

OBJECTIVES: The student will be able to know the following

- To understand the major tools and techniques that allow programmers to effectively program the parts of the code that require substantial communication and synchronization.
- To study the core ideas behind modern coordination paradigms and concurrent data structures.
- To introduce a variety of methodologies and approaches for reasoning about concurrent programs.
- To realize not only the basic principles but also the best practice engineering techniques of concurrent computing;

UNIT 1	Characterization of Distributed Systems: Introduction, Examples of distributed Systems, Issues in Distributes Operating Systems, Resource sharing and the Web Challenges. System Models: Architectural models, Fundamental Models Theoretical Foundation for Distributed System: Limitation of Distributed system, absence of global clock, shared memory, Logical clocks, Lamport‟s& vectors logical clocks, Causal ordering of messages, global state, termination detection. Distributed Mutual Exclusion: Classification of distributed mutual exclusion, requirement of mutual exclusion theorem, Token based and non token based algorithms, performance metric for distributed mutual exclusion algorithms.
UNIT 2	Distributed Deadlock Detection: system model, resource Vs communicationdeadlocks, deadlock prevention, avoidance, detection & resolution, centralized dead lock detection, distributed dead lock detection, path pushing algorithms, edge chasing algorithms.Agreement Protocols: Introduction, System models, classification of Agreement Problem-Interactive consistency Problem, Applications of Agreement algorithms.
UNIT 3	Distributed Objects and Remote Invocation: Communication between distributed objects, Remote procedure call, Events and notifications, Java RMI case study. Transactions and Concurrency Control: Transactions, Nested transactions, Locks, Optimistic Concurrency control, Timestamp ordering, Comparison of methods for concurrency control
UNIT 4	Distributed Transactions: Introduction, Flat and nested distributed transactions,Atomic commit protocols, concurrency control in distributed transactions, Distributed deadlocks, Transaction recovery. Distributed shared memory – Design and Implementation issues, consistency models, CORBA Case Study: CORBA RMI, CORBA services.
UNIT 5	File service components, design issues, interfaces, implementation techniques, Sun Network File System – architecture and implementation, other distributed file systems – AFS, CODA. Name services – SNS name service model.

OUTCOMES:-The students will be able to know the following.

- Identifying techniques to formally prove correctness of multiprocessor programs;

- Presenting techniques to formally study the progress properties of concurrent algorithms;
- Analyzing the performance of multiprocessor algorithms;
- Identifying limitations and impossibility results which express where the effort should not be put in solving a task

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

- H = Highly Related; M = Medium L = Low

Text Books:

1. "Advanced Concepts in Operating Systems", by MukeshSinghal&Niranjan G Shivaratri,Tata McGraw Hill(2001).
2. “Distributed System: Concepts and Design”, by Coulouris, Dollimore, Kindberg , Pearson Education (2006)

Reference Books:

1.Tanenbaum S , “Distributed Operating Systems”, Pearson Education (2005).
2. P K Sinha, “"Distributed System: Concepts and Design”, PHI (2004).

BCO 050A	HUMAN COMPUTER INTERFACE	4-0-0 [4]

OBJECTIVE:-

- To know what the user-cantered design cycle is and how to practice this approach to design your own website or other interactive software systems
- T critique existing website and other interactive software using guidelines from human factor theories
- To analyze one after another the main features of a GUI: the use of colors, organization and layout of content, filling the interface with useful and relevant information, and communication techniques; and to critique designs in order to provide better solutions

UNIT1	Introduction0 The human, The computer, The interaction, Paradigms, Usability of Interactive Systems, Guidelines, Principles, and Theories
UNIT 2	Design Process- Interaction design basics, HCI in the software process, Design rules, Implementation support, Evaluation techniques, Universal design, User support
UNIT 3	Models and Theories0 Cognitive models, Socio-organizational issues and stakeholder requirements, Communication and collaboration models, Task analysis, Dialogue notations and design, Models of the system, Modeling rich interaction
UNIT 4	Interaction Styles- Direct Manipulation and Virtual Environments, Menu Selection, Form Filling and Dialog Boxes, Command and Natural Languages, Interaction Devices, Collaboration and Social Media Participation
UNIT5	Design Issues- Quality of Service, Balancing Function and Fashion, User Documentation and Online Help, Information Search, Information Visualization UNIT6: Outside the Box- Group ware, Ubiquitous computing and augmented realities, Hypertext, multimedia, and the world wide web

Learning Outcomes: -

After completing this course students must be able to demonstrate the knowledge and ability to:

- Explain the human components functions regarding interaction with computer
- Explain Computer components functions regarding interaction with human
- Demonstrate Understanding of Interaction between the human and computer components.
- Use Paradigms, implement Interaction design basics, Use HCI in the software process
- Apply Design rules, Produce Implementation supports, Use Evaluation techniques
- **MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

- H = Highly Related; M = Medium L = Low

Text Books:

1,“Human Computer Interaction” by Alan Dix, Janet Finlay , ISBN :9788131717035, Pearson Education (2004)

2.“Designing the User Interface - Strategies for Effective Human Computer Interaction”, by Ben ShneidermanISBN : 9788131732557, Pearson Education (2010).

Reference Books:

1. Usability Engineering: Scenario-Based Development of Human-Computer Interaction , by Rosson, M. and Carroll, J. (2002)

2. The Essentials of Interaction Design, by Cooper, et al. , Wiley Publishing(2007)

3.Usability Engineering, by Nielsen, J. Morgan Kaufmann, San Francisco, 1993. ISBN 0-12-518406-9

4.The Resonant Interface: HCI Foundations for Interaction Design , by Heim, S. , Addison-Wesley. (2007)

5.Usability engineering: scenario-based development of human-computer interaction, By Rosson, M.B &Carroll, J.M. , Morgan Kaufman.(2002)

BCO 051A	APPROXIMATION OF ALGORITHMS	4-0-0 [4]

OBJECTIVES:-Upon completion of this course, students will be able to do the following:

- Analyze the asymptotic performance of algorithms.
- Write rigorous correctness proofs for algorithms.
- Demonstrate a familiarity with major algorithms and data structures.
- Apply important algorithmic design paradigms and methods of analysis.
- Synthesize efficient algorithms in common engineering design situations.

UNIT1	Background; 2-approx for unweighted vertex cover via max‟l matching,2-approx for steiner tree via MST; Christofides 1.5-approx for metric TSP;
UNIT 2	PTAS for knapsack by coarsening; ln(n)-approx for set cover by greedy;
UNIT 3	Linear programming; Duality, 2-approx for weighted vertex cover via duality;
UNIT 4	Basic probability, randomized rounding; max-cut, max-sat by randomized rounding;
UNIT5	Chernoffbound, load-balancing (from chapter); Semi-definite programming, maxcut by semi-definite programming multi-commodity flow by lagrangianrelaxation?Hardness of approximation

OUTCOMES:
Upon completion of this course, the student will be able to:

CO1: Explain the different ways to analyze randomized algorithms (expected running time, probability of error). Recite algorithms that employ randomization. Explain the difference between a randomized algorithm and an algorithm with probabilistic inputs.
CO2: Analyze randomized algorithms. Employ indicator random variables and linearity of expectation to perform the analyses. Recite analyses of algorithms that employ this method of analysis.
CO3: Explain what amortized running time is and what it is good for. Analysis of different Methods of amortized analysis (aggregate analysis, accounting, potential method). Perform amortized analysis.
CO4: Methods of amortized analysis (aggregate analysis, accounting, potential method). Perform amortized analysis.
CO5: Explain what competitive analysis is and to which situations it applies. Perform competitive analysis.

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	H	M		M								H	M	
CO2	H		M		M								M	M	
CO3	M	H	M		H								M	H	
CO4	M	H	M	M	H								H	H	

CO5	H	M	L		H			L				L	H	M	M

Text Books:

1.Approximation Algorithms, by Vijay Vazirani, Springer-Verlag, ISBN: 3-540-65367-8, published 2001.

Reference Books:

1.DoritHochbaum (Editor), Approximation Algorithms for NP-Hard Problems, Brooks/Cole Pub Co; ISBN:0534949681; 1st edition (July 26, 1996)

2.AlexanderSchrijver, Theory of Linear and Integer Programming, ISBN: 0471982326, Wiley, John &Sons,Incorporated, June 1998.

BCO 052A	WIRELESS ADHOC NETWORKS	4-0-0 [4]

OBJECTIVE:-To demonstrate the knowledge and ability to:

- Covers fundamental principles of ADHOC Networks.
- To develop a comprehensive understanding of AdHoc network protocols .
- To understand current and emerging trends in Wireless Networks.

UNIT 1	**Introduction**– Issues in Ad-Hoc Wireless Networks. MAC Protocols – Issues, Classifications of MAC protocols, Multi channel MAC & Power control MAC protocol.
UNIT 2	**AD-HOC NETWORK ROUTING & TCP:** Classifications of routing protocols – Hierarchical and Power aware. Multicast routing – Classifications, Tree based, Mesh based. Ad Hoc Transport Layer Issues. TCP Over Ad Hoc – Feedback based, TCP with explicit link, TCP-Bus, Ad Hoc TCP, and Split TCP
UNIT 3	**WSN ROUTING, LOCALIZATION & QOS:** Issues in WSN routing – OLSR, AODV. Localization – Indoor and Sensor NetworkLocalization.QoS in WSN
UNIT 4	Basic probability, randomized rounding; max-cut, max-sat by randomized rounding;
UNIT 5	**MESH NETWORKS :**Necessity for Mesh Networks – MAC enhancements – IEEE 802.11s Architecture – Opportunistic routing – Self configuration and Auto configuration – Capacity Models – Fairness – Heterogeneous Mesh Networks – Vehicular Mesh Networks

Course Outcome(CO):

CO1: Understand and analyze the fundamental of sensor networks and energy efficient sensor node and network architectures.

CO2: Understand and analyze the design issues in physical Layer.

CO3: Understand and analyze different communication protocols and their performance.

CO4: The broad education necessary to understand and analyze different routing strategies.

CO5: Understand the modern tool used for wireless sensors networks.

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	H											H		
CO2	M	H	L										H		
CO3	M	M											H		
CO4	M	M											H		
CO5					H									M	H

Text books:-

1.C.SivaRamMurthyandB.Smanoj,"AdHocWirelessNetworksarchitecturesandProtocols",PearsonEducation, 2004.

2.FengZhaoandLeonidasGuibas,"WirelessSensor Networks",MorganKaufmanPublishers, 2004.

Reference Books:-

1.C.K.Toh,"AdHocMobileWirelessNetworks",PearsonEducation,2002.

2.ThomasKragandSebastinBuettrich,"WirelessMeshNetworking",O'ReillyPublisher, 2007.

BCO 070B	Big Data Analytics	4-0-0 [4]

OBJECTIVE:-To demonstrate the knowledge and ability to:

- Covers fundamental principles of Big Data.
- To develop a comprehensive understanding of Big Data Analytics.
- To understand current and emerging trends in Big Data.

UNIT 1	Big Data, Complexity of Big Data, Big Data Processing Architectures, Exploring the Big Data Technologies, Big Data Business Value, Big Data Analytics, Visualization and Data Scientist
UNIT 2	Data Warehouse, Re-Engineering the Data Warehouse, Workload Management in the Data Warehouse, Integration of Big Data and Data Warehouse, Data Driven Architecture, Information Management and Lifecycle
UNIT 3	Understanding Hadoop features, Learning the HDFS and MapReduce architecture, Understanding Big Data Analysis with Machine Learning, Supervised Machine Learning Algorithms, Unsupervised machine learning algorithm, Recommendation algorithms
UNIT 4	Understanding Hadoop features, Learning the HDFS and MapReduce architecture, Understanding Big Data Analysis with Machine Learning, Supervised Machine Learning Algorithms, Unsupervised machine learning algorithm, Recommendation algorithms
UNIT 5	to Big Data Applications (Graph Processing) Text Analytics and The New Information Management Paradigm, Big Data's Implication for Businesses, Big Data Implications for Information Management

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

H = Highly Related; M = Medium L = Low

Textbooks:

1. Data Warehousing in the Age of Big Data by Krish Krishnan, Morgan Kaufmann.
2. A.Ohri, "R for Business Analytics", Springer, 2012.

References:

1. Big Data Analytics with R and Hadoop by Vignesh Prajapati
2. Principles of Big Data Preparing, Sharing, and Analyzing Complex Information, 1st Edition, by J Berman, published by Morgan Kaufmann
3. “Big Data Analytics - From Strategic Planning to Enterprise Integration with Tools, Techniques, NoSQL, and Graph” By David Loshin, Morgan Kaufmann
4. Big Data Imperatives: Enterprise 'big Data' Warehouse, 'BI' Implementations and Analytics by Soumendra Mohanty, Apress
5. Big Data Analytics Using Splunk By Peter Zadrozny , Raghu Kodali, Apress 2013
6. Franks, Bill, “Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics”, Wiley, 1st Edition, 2012.
7. Big Data Application Architecture Q&A: a Problem - Solution Approach Nitin Sawant, Himanshu Shah 8. Big Data Now: Current Perspectives from O'Reilly Radar By O'Reilly Radar Team

BCO 091A	DevOps	4-0-0 [4]

OBJECTIVE:

- To explain the drivers responsible for the emergence of DevOps and introduce the key concepts and principles of DevOps.
- To describe the business benefits of DevOps and continuous delivery and describe the Service Delivery process.
- To describe how DevOps relates to Lean and Agile methodologies and learn the most common and popular DevOps tools.
- To summarize case studies of IT organizations that are making the transformation to Adaptive IT and DevOps models.
- To implement DevOps within the enterprise via the Docker platform.

UNIT 1	**DevOps Essential:** What is DevOps?,Why DevOps?,Evolution of Software Methodologies,Dev Challenges v/s DevOps Solution, Ops Challenges v/s DevOps Solution, Stages Of DevOps Lifecycle: Continuous Deveopment, Continuous Testing, Continuous Integration, Continuous Deployment, Continuous Monitoring; Dark Launching Technique, The DevOps Ecosystem, The Various DevOps Tools
UNIT 2	**Git & GitHub:** What is Version Control System(VCS)? ,Why VCS?, VCS tools, Distributed VCS, What is Git & Why Git?, Features Of Git, Git Workflow, Git Configurations, Creating Git Repository, Syncing Repositories: Adding Origin, Pushing changes, Pulling changes; Clone operation; Perform, Review & Commit Changes; Stacking Unfinished Changes; Move, Rename & Delete Operations; Tagging Versions In Repository
UNIT 3	**DevOps Tools: JENKINS** Challenges before Continuous Integration,What is Continuous Integration?,Benefits of Continuous Integration,Tools of Continuous Integration, Introduction to Jenkins, Configuring Jenkins, Build Setup in Jenkins, Jenkins Dashboard, Creating jobs in Jenkins. **PUPPET** Challenges before Configuration Management, Advantages of Inheritance, What is Configuration Management?, Configuration management Components, Configuration management Tools, What is Puppet?, Puppet Environment, Master-slave Architecture of Puppet, How Puppet Works?, Components of Puppet, Generating Master & Agent Certificates, Basic Puppet Terminologies
UNIT 4	**DevOps Tools (Contd.): CHEF** Chef Fundamentals, Chef environment, Chef Cookbooks, Knife Commands, Node Object & Search, Chef Data-bags, Roles in Chef, Deploying Nodes in Production, Vagrant file. **NAGIOS** What is Continuous Monitoring?, Introduction to Nagios, Nagios Setup, Nagios Plugins Introduction to Events, Objects in Nagios, Nagios Commands, Nagios Notification.
UNIT 5	**Docker:** Challenges before Containerization, Understanding microservices, VMs for microservices, What is a Container?, VM v/s Containers, Benefits of Containerization Introduction to Docker: Docker Fundamentals, Architecture of Docker, Creating & Executing Docker, Images, Image Distribution, Docker Registry, What is Docker Hub?,

	Docker Containers, Creating Docker Files.

Course Outcome (CO):

At the end of this course students will be able to:

CO1: Explain the drivers responsible for the emergence of DevOps and discuss the key concepts and principles of DevOps.
CO2: Explain the business benefits of DevOps and continuous delivery and describe the Service Delivery process.
CO3: Describe how DevOps relates to Lean and Agile methodologies and learn the most common and popular DevOps tools.
CO4: Summarize case studies of IT organizations that are making the transformation to Adaptive IT and DevOps models.
CO5: Enable DevOps within the enterprise via the Docker platform.

Open Elective- VI Semester

BCO 001A	SOFTWARE ENGINEERING	3-0-0 [3]

Objective

- To learn about generic models of software development process.
- To understand fundamental concepts of requirements engineering and Analysis Modeling.
- To understand the different design techniques and their implementation.
- To learn various testing and maintenance measures

UNIT 1	Introduction- Introduction to Software Engineering, Software Components, Software Characteristics, Software Crisis, Engineering aspects of Software production – necessity of automation .Job responsibilities of Programmers and Software Engineers as Software developers.Software Development Life Cycle (SDLC)
UNIT 2	Process Models and Program Design Techniques- Software Development Process Models – Code & Fix model, Waterfall model, Incremental model, Rapid Prototyping model, Spiral (Evolutionary) model.Software Requirement Specifications (SRS), Management of User Needs, Data Flow Diagrams, Entity Relationship Diagrams, Decision Tables, SRS Document, Design Techniques – Structured Programming, Coupling and Cohesion, Abstraction and Information Hiding, SoftwareModeling Tools –Data flow Diagrams, UML and XML.
UNIT 3	Software Testing: Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Verification and Validation: Testing of Software Products – Black-Box Testing and White-Box Testing, Static Analysis, Symbolic Execution and Control Flow Graphs –Cyclomatic Complexity.
UNIT 4	Software Project Management: Management Functions and Processes, Project Planning and Control, Organization and Intra-team Communication, Risk Management. Software Cost Estimation – underlying factors of critical concern. Metrics for estimating costs of software products – Function Points. Techniques for software cost estimation –Expert judgment, Work break-down structure and Process breakdown structure, COCOMO and COCOMO-II.
UNIT 5	Software Maintenance,Need for Maintenance, Categories of Maintenance, An Overview of CASE Tools. Advanced Topics: Support environment for Development of Software Products. Representative Tools for Editors, Linkers, Interpreters, Code Generators, Debuggers. Tools for Decision Support and Synthesis, Configuration control and Engineering Databases.

Course Outcome (CO):

At the end of this course studentswill have:

CO1: An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.
CO2: An ability to identify, formulates, and solve engineering problems.
CO3: An understanding of professional and ethical responsibility.
CO4: The broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	L		H		M						L				
CO2		H		H								M	M		
CO3								H	M	L				H	
CO4	L		L			L	M								L

H = Highly Related; M = Medium L = Low

Text Books:

3. Fundamentals of Software Engineering – Carlo Ghezziet. Et.al.
4. Software Engineering – Design, Reliability Management – Pressman.

Reference Books:

5. Software Engineering – Ian Sommerville.
6. Software Engineering - Shoeman.
7. Software Engineering with Abstraction – Berzins and Luqi
8. Pankaj Jalote, Software Engineering, Wiley

BCO 062A	SOFTWARE TESTING	4-0-0 [4]

OBJECTIVES

1. To apply the testing strategies and methodologies in their projects
2. To understand test management strategies and tools for testing
3. A keen awareness on the open problems in software testing and maintenance

UNIT 1	**TESTING BASICS** Testing as an engineering activity – Role of process in software quality – Testing as a process – Basic definitions – Software testing principles – The tester's role in a software development organization – Origins of defects – Defect classes – The defect repository and test design – Defect examples – Developer / Tester support for developing a defect repository.
UNIT 2	**TEST CASE DESIGN** Introduction to testing design strategies – The smarter tester – Test case design strategies – Using black box approach to test case design – Random testing – Equivalence class partitioning – Boundary value analysis – Other black box test design approaches – Black box testing and COTS – Using white box approach to test design – Test adequacy criteria – Coverage and control flow graphs – Covering code logic – Paths – Their role in white box based test design – Additional white box test design approaches – Evaluating test adequacy criteria.
UNIT 3	**LEVELS OF TESTING** The need for levels of testing – Unit test – Unit test planning – Designing the unit tests – The class as a testable unit – The test harness – Running the unit tests and recording results – Integration tests – Designing integration tests – Integration test planning – System test – The different types – Regression testing – Alpha,beta and acceptance tests.
UNIT 4	**TEST MANAGEMENT** Basic concepts – Testing, debugging goals, policies – Test planning – Test plan components – Test plan attachments – Locating test items – Reporting test results – The role of three groups in test planning and policy development – Process and the engineering disciplines – Introducing the test specialist – Skills needed by a test specialist – Building a testing group.
UNIT 5	**CONTROLLING AND MONITORING** Defining terms – Measurements and milestones for controlling and monitoring – Status meetings – Reports and control issues – Criteria for test completion – SCM – Types of reviews – Developing a review program – Components of review plans – Reporting review results.

Course Outcome (CO):

At the end of this course studentswill have:

CO1: An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.
CO2: An ability to identify, formulates, and solve engineering problems.
CO3: An understanding of professional and ethical responsibility.

CO4: The broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	L		H		M						L				
CO2		H		H								M	M		
CO3								H	M	L				H	
CO4	L		L			L	M								L

H = Highly Related; M = Medium L = Low

.

Text Books:

1. SrinivasanDesikan, Gopalaswamy Ramesh, "*Software Testing: Principles and Practices*", Pearson 2012
2. Aditya P. Mathur, "*Foundations of Software Testing*", Pearson, 2008

References:

1. Paul Ammann, Jeff Offutt, "*Introduction to Software Testing*", Cambridge University Press, 2008
2. Paul C. Jorgensen, "*Software Testing: A Craftsman's Approach*", Auerbach Publications, 2008

BCO 063A	ADVANCED TOPICS IN DATABASE MANAGEMENT SYSTEM	4-0-0 [4]

Objective:

1. To learn about the fundamentals of distributed databases.
2. To learn about the fundamentals of Object Oriented databases
3. To learn about the fundamentals of xml databases
4. To learn about the fundamentals of mobile databases.

UNIT 1	Parallel and Distributed Databases Database System Architectures: Centralized and Client-Server Architectures – Server System Architectures – Parallel Systems- Distributed Systems – Parallel Databases: I/O Parallelism –Inter and Intra Query Parallelism – Inter and Intra operation Parallelism – Distributed Database Concepts - Distributed Data Storage – Distributed Transactions – Commit Protocols – Concurrency Control – Distributed Query Processing – Three Tier Client Server Architecture, Case Studies.
UNIT 2	Object and Object Relational Databases Concepts for Object Databases: Object Identity – Object structure – Type Constructors –Encapsulation of Operations – Methods – Persistence – Type and Class Hierarchies – Inheritance– Complex Objects – Object Database Standards, Languages and Design: ODMG Model –ODL – OQL – Object Relational and Extended – Relational Systems : Object Relational featuresinSQL/Oracle – Case Studies.
UNIT 3	XML Databases XML Databases: XML Data Model – DTD - XML Schema - XML Querying – Web Databases –JDBC – Information Retrieval – Data Warehousing – Data Mining
UNIT 4	Mobile Databases Mobile Databases: Location and Handoff Management - Effect of Mobility on Data Management - Location Dependent Data Distribution - Mobile Transaction Models - Concurrency Control -Transaction Commit Protocols- Mobile Database Recovery Schemes
UNIT 5	Multimedia Databases Multidimensional Data Structures – Image Databases – Text/Document Databases- Video Databases – Audio Databases – Multimedia Database Design.

Outcomes:

Upon completion of the subject, students will be able to:

- Understand and apply the basic concepts of information retrieval.
- Appreciate the limitations of different information retrieval techniques.
- Write programs to implement search engines.
- Evaluate search engines.

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1		L	L								L		M	L	
CO2		L	H		L			L		L	M		M	M	
CO3		M	H								M		H	M	
CO4		M	M								M			M	
CO5		L	M	L							L	L	M	H	

Text Books:

1.C. D. Manning, P. Raghavan, and H. Schutze, *An Introduction to Information Retrieval,* Cambridge University Press, 2009.

Reference Books:

2. 1.R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval, Pearson Education*, 1999.

BCO 035A	Programming in Java	4:0:0 [4]

Objective

- Cover issues related to the definition, creation and usage of classes, objects and methods.
- Discuss the principles of inheritance and polymorphism and demonstrate though problem analysis assignments how they relate to the design of methods, abstract classes and interfaces.
- Provide the foundation of good programming skills by discussing keys issues to the design of object-oriented software, including programming design patterns, automatic documentation techniques and programming testing.
- Cover the basics of creating APIs as well as allow students to explore the Java Abstract Programming Interface (API) and Java Collection Framework through programming assignments.
- Discuss basic principles and tools of collaborating programming (versioning systems, code review) and study their usage through group programming projects.

UNIT 1	**Java Fundamentals:** Features of Java ,OOPs concepts , Java virtual machine , Reflection byte codes ,Byte code interpretation , Data types, variable, arrays, expressions, operators, and control structures , Objects and classes
UNIT 2	**Java Classes:** Abstract classes ,Static classes ,Inner classes ,Packages,Wrapper classes Interfaces ,This ,Super ,Access control
UNIT 3	**Exception handling:** Exception as objects ,Exception hierarchy ,Try catch finally ,Throw, throws
UNIT 4	**IO package:** Input streams ,Output streams ,Object serialization ,De serialization ,Sample programs on IO files ,Filter and pipe streams
UNIT 5	**Multi threading:** Thread Life cycle ,Multi threading advantages and issues ,Simple thread program ,Thread synchronization .GUI: Introduction to AWT programming, Layout and component managers ,Event handling ,Applet class ,Applet life-cycle ,Passing parameters embedding in HTML ,Swing components – JApplet, JButton, JFrame, etc. Sample swing programs

Outcome:

At the end of this course student will:

Course Outcome: At the end of this course student will:

CO1:Understand how object-oriented concepts are incorporated into the Java programming language

CO2: Develop problem-solving and programming skills using OOP concept

CO3:Understand the benefits of a well structured program

CO4:Develop the ability to solve real-world problems through software development in high-level programming language like Java

CO5:Develop efficient Java applets,threading and applications using OOP concept

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1		M		M	H				M		H		M	H	
CO2	L		M		H		L	L		M		M		H	M
CO3		M		H	M	L		L		M	H		M	H	
CO4			H	M			L		M		H		M	H	
CO5			H	M		L						M	H	H	

- H = Highly Related; M = Medium L = Low

References:

1. Programming with Java A Primer, E.Balaguruswamy Tata McGraw Hill Companies

2. Java Programming John P. Flynt Thomson 2nd

3. Java Programming Language Ken Arnold Pearson

4. The complete reference JAVA2, Herbert schildt. TMH

B.Tech CSE Semester VII

BCO 027A	Mobile Computing	3-0-0 [3]

OBJECTIVES:

To understand the concept of mobile computing and architecture of mobile communication.

- To apply the concepts of mobile communications to the transactions and transaction management.
- Apply the concepts of mobile computing and conventional wired network and simulate it on the simulator.
- To understand the working of heterogeneousnetworks.

UNIT 1	Technical Background – Transmission Fundamentals , Communication Networks , Protocols and the TCP/IP Suite
UNIT 2	Wireless Communication Technology : Cellular Wireless Networks , Antennas and Wave Propagation , Modulation Techniques , Multiple Access in Wireless System ,Mobile Adaptive Computing , Mobility Management , Data Dissemination and Management,Context-Aware Computing
UNIT 3	Introduction to Mobile Middleware , Middleware for Application Development: Adaptation and Agents , Service Discovery Middleware: Finding Needed Services
UNIT 4	Introduction to Ad Hoc and Sensor Networks , Challenges , Protocols
UNIT 5	Wireless Security , Approaches to Security , Security in Wireless Personal Area Networks , Security in Wireless Local Area Networks ,Security in Wireless Metropolitan Area Networks (802.16) , Security in Wide Area Networks

Course Outcome (CO):

At the end of this course studentswill have:

CO1: An ability to understand mobile computing technical background, communication networks and different protocols.
CO2: An ability to understand wireless communication technology and solve challenges in wireless.
CO3: An understanding of middleware responsibility.
CO4: To understand the impact of engineering solutions in a global, economic, environmental, and security.

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H		H		M						L			L	
CO2		H		L								M	M		
CO3								L	M	L				M	
CO4	L		L			L									L

Text Books:

1.Wireless Communications and Networking, Willam Stallings, Pearson Education. (2002)
2."Fundamentals of Mobile & Pervasive Computing" by Frank Adelstein, Sandeep Ks Gupta ,ISBN : 9780070603646,TMH (2005)

Reference Books:

1.Jochen Schiller, *"Mobile Communications,"* Addison-Wesley (2009)
2.R. Dayem, *"Mobile Data & Wireless Lan Technologies,"* Prentice-Hall (2005)

BCO 028A	COMPILER CONSTRUCTION	3-0-0 [3]

OBJECTIVE: At the end of the course, the student should be able to:

- Apply the principles in the theory of computation to the various stages in the design of compilers;
- Explain the stages involved in the translation process;
- Analyse problems related to the stages in the translation process;
- Design a compiler for a simple programming language; and
- Implement a compiler based on its design.

UNIT 1	Overview of compilation- The structure of a compiler and applications of compiler technology; Lexical analysis - The role of a lexical analyzer, specification of tokens, recognition of tokens, hand-written lexical analyzers, LEX, examples of LEX programs. Introduction to syntax analysis -Role of a parser, use of context-free grammars (CFG) in the specification of the syntax of programming languages, techniques for writing grammars for programming languages (removal left recursion, etc.), non- context-free constructs in programming languages, parse trees and ambiguity, examples of programming language grammars.
UNIT 2	Top-down parsing- FIRST & FOLLOW sets, LL(1) conditions, predictive parsing, recursive descent parsing, error recovery. LR-parsing - Handle pruning, shift-reduce parsing, viable prefixes, valid items, LR(0) automaton, LR-parsing algorithm, SLR(1), LR(1), and LALR(1) parsing. YACC, error recovery with YACC and examples of YACC specifications.
UNIT 3	Syntax-directed definitions (attribute grammars)-Synthesized and inherited attributes, examples of SDDs, evaluation orders for attributes of an SDD, Dependency graphs-attributed and L-attributed SDDs and their implementation using LR-parsers and Recursive Descent parsers respectively.
UNIT 4	Semantic analysis- Symbol tables and their data structures. Representation of "scope". Semantic analysis of expressions, assignment, and control-flow statements, declarations of variables and functions, function calls, etc., using S- and L-attributed SDDs (treatment of arrays and structures included). Semantic error recovery.
UNIT 5	Intermediate code generation - Different intermediate representations – quadruples, triples, trees, flow graphs, SSA forms, and their uses. Translation of expressions (including array references with subscripts) and assignment statements. Translation of control-flow statements – it- then-else, while-do, and switch. Short-circuit code and control-flow translation of Boolean expressions. Back patching. Examples to illustrate intermediate code generation for all constructs. Run-time environments:- Stack allocation of space and activation records. Access to non-local data on the stack in the case of procedures with and without nesting of procedures.

Course Outcome

At the end of this course students will have:

CO1 To apply the knowledge of lex tool & yacc tool to devleop a scanner & parser.
CO2 To design parser and Intermediate Code Generation in compiler.
CO3 To deal with different translators.
CO4 To learn the new code optimization techniques to improve the performance of a program in terms of speed & space.
CO5 To use the knowledge of patterns, tokens & regular expressions for solving a problem.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1	H	H	L		H			L					H	M	
CO2		H				L							H		
CO3		L		H	L										
CO4		H					H							H	H
CO5		H		L		H									

H = Highly Related; M = Medium L = Low

Text Books:

1. Compilers: Principles, Techniques, and Tools, by A.V. Aho, Monica Lam, Ravi Sethi, and J.D. Ullman, (2nd ed.), Addison-Wesley, 2007 (main text book, referred to as ALSU in lab assignments).
2. K.D. Cooper, and Linda Torczon, Engineering a Compiler, Morgan Kaufmann, 2004.

Reference Books:

1. K.C. Louden, Compiler Construction: Principles and Practice, Cengage Learning, 1997.
2. D. Brown, J. Levine, and T. Mason, LEX and YACC, O‟Reilly Media, 1992.

BCO 029A	DATA MINING &WAREHOUSING	3:0:0 [3]

OBJECTIVE:

- To compare and contrast different conceptions of data mining.
- To explain the role of finding associations in commercial market basket data.
- To characterize the kinds of patterns that can be discovered by association rule mining.
- To describe how to extend a relational system to find patterns using association rules.
- To evaluate methodological issues underlying the effective application of data mining.

UNIT 1	Introduction: Basic concepts of data mining, including motivation and definition; different types of data repositories; data mining functionalities; concept of interesting patterns; data mining tasks; current trends, major issues and ethics in data mining
UNIT 2	Data: Types of data and data quality; Data Preprocessing: data cleaning, data integration and transformation, data reduction, discretization and concept hierarchy generation; Exploring Data: summary statistics, visualization, multidimensional data analysis
UNIT 3	Association and Correlation Analysis: Basic concepts: frequent patterns, association rules - support and confidence; Frequentitemset generation - Apriori algorithm, FP-Growth algorithm; Rule generation, Applications of Association rules; Correlation analysis.
UNIT 4	Clustering Algorithms and Cluster Analysis: Concept of clustering, measures of similarity, Clustering algorithms: Partitioning methods - k-means and k-medoids, CLARANS, Hierarchical methods - agglomerative and divisive clustering, BIRCH, Densitybased methods - Subspace clustering, DBSCAN; Graph-based clustering - MST clustering; Cluster evaluation; Outlier detection and analysis.
UNIT 5	Classification: Binary Classification - Basic concepts, Bayes theorem and Naïve Bayes classifier, Association based classification, Rule based classifiers, Nearest neighbor classifiers, Decision Trees, Random Forest; Perceptrons; Multi-category classification; Model overfitting, Evaluation of classifier performance - cross validation, ROC curves. Applications: Text mining, Web data analysis, Recommender systems.Prerequisites: Familiarity with basic Linear Algebra and Probability will be assumed.

OUTCOMES: : At the end of the course, the student should be able to:

- Compare and contrast different conceptions of data mining.
- Explain the role of finding associations in commercial market basket data.
- Characterize the kinds of patterns that can be discovered by association rule mining.
- Describe how to extend a relational system to find patterns using association rules.
- Evaluate methodological issues underlying the effective application of data mining.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

H = Highly Related; M = Medium L = Low

Text Books:

1. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, Introduction to Data Mining. Pearson (2005), India.ISBN 978-8131714720

2. Jiawei Han and MichelineKamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, 3rd edition (July 2011) 744 pages. ISBN 978-0123814791

Reference Books:

1. T. Hastie, R. Tibshirani and J. H. Friedman, The Elements of Statistical Learning, Data Mining, Inference, andPrediction.Springer, 2nd Edition, 2009.768 pages. ISBN 978-0387848570

2. C. M. Bishop, Pattern Recognition and Machine Learning.Springer, 1st edition, 2006.738 pages. ISBN 978-0387310732

3. Ian H. Witten and Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques, MorganKaufmann, 3rd edition (January 2011).664 pages. ISBN 978-0123748560.

BCO 030A	PRINCIPLES OF INFORMATION SYSTEM SECURITY	3-0-0 [3]

Objective:

- To explain the objectives of information security
- To analyse the trade-offs inherent in security
- To describe the enhancements made to IPv4 by IPSec
- To understand the basic categories of threats to computers and networks
- To discuss issues for creating security policy for a large organization

UNIT 1	Information Security: Introduction, History of Information security, What is Security, CNSS Security Model, Components of Information System, Balancing Information Security and Access, Approaches to Information Security Implementation, The Security Systems Development Life Cycle.
UNIT 2	Cryptography: Concepts and Techniques, symmetric and asymmetric key cryptography, steganography, Symmetric key Ciphers: DES structure, DES Analysis, Security of DES, variants of DES, Block cipher modes of operation , AES structure, Analysis of AES , Key distribution Asymmetric key Ciphers: Principles of public key cryptosystems, RSA algorithm, Analysis of RSA, Diffie-Hellman Key exchange
UNIT 3	Message Authentication and Hash Functions: Authentication requirements and functions, MAC and Hash Funtions, MAC Algorithms: Secure Hash Algorithm, Whirlpool, HMAC, Digital signatures, X.509, Kerberos
UNIT 4	Security at layers(Network, Transport, Application): IPSec, Secure Socket Layer(SSL), Transport Layer Security(TLS), Secure Electronic Transaction(SET), Pretty Good Privacy(PGP), S/MIME
UNIT 5	Inruders, Virus and Firewalls: Intruders, Intrusion detection, password management, Virus and related threats, Countermeasures, Firewall design principles, Types of firewalls

OUTCOMES: At the end of the course, the student should be able to:

CO1: Explain the objectives of information security and analyze the importance of information Security in real world.
CO2: Analyse the trade-offs inherent in security and designing and analysis of different encryption Algorithms.
CO3: Implementation of MAC and Hash functions, security at different layers of a network
CO4: Understand the basic categories of threats to computers and networks and explore different types of intruders and viruses.
CO5: Discuss issues for creating security policy for a large organization

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	H					M					M			M
CO2	H	H	H	M								M		M	
CO3	H			L									H		
CO4	L				H						M			L	
CO5	M				H	H	H	M		L	M				M

H = Highly Related; M = Medium L=Low

Text Books –

1. Stalling Williams: Cryptography and Network Security: Principles and Practices, 4th Edition, Pearson Education, 2006.

2. Kaufman Charlie et.al; Network Security: Private Communication in a Public World, 2nd Ed., PHI/Pearson.

Reference Books:

1. Pieprzyk Josef and et.al; Fundamentals of Computer Security, Springer-Verlag, 2008.

2. Trappe & Washington, Introduction to Cryptography, 2nd Ed. Pearson.

BCO 031A	Compiler Design Lab	0:0:2 [2]

List Of Experiments

1 Familiarization with LEX by writing simple specifications for tokens such as identifiers, numbers, comments in C/C++, etc. All LEX specifications must be compiled andexecuted with appropriate inputs. At least ten such exercises must be completed in two labclasses.

2 LEX specification for tokens of the small language in ALSU‟s book

3 Complete the specifications in (2) above to make a complete lexical analyzer. (1 lab class)

4 Familiarization with YACC by writing simple specifications for desk calculator, variable declarations in C (only numbers and array). All YACC specifications must be compiled and executed with appropriate inputs. Note that this exercise also requires LEX specifications o the tokens involved. (2 lab classes)

5 YACC specifications for the syntax of the small language in ALSU‟s book (appendix A)(1 lab class)

6 Adding error recovery to (5) above to make a complete parser. (1 lab class)

7 S-attributed specification of the semantics of the small language in ALSU‟s book

8 Adding semantic error recovery to the semantic analyzer in (7) above to make a complete semantic analyzer. (1 lab class)

9 Intermediate code generation for the constructs of the small language in ALSU‟s book (appendix A) to be incorporated into the semantic analyzer of (8) above. Students doing this last assignment may be awarded bonus marks. (3 lab classes)

10 Write a programme to parse using Brute force technique of Top-down parsing.

11 Write a program for generating for various intermediate code forms
i) Three address code ii) Polish notation

12 Develop an operator precedence parser (Construct parse table also)

13 Develop a recursive descent parser

14 Develop a lexical analyser to recognize a few patterns.

(VII Semester)

Program Elective –V

BCO 084A	CYBER SECURITY	4-0-0 [4]

OBJECTIVE:

- To study various cyber security concepts like vulnerability, injection tools
- To overview the applications of cyber security.
- To be familiar with utilization of cyber security concepts in cyber crime.
- To have a comprehensive knowledge of cyber security..

UNIT 1	**Systems Vulnerability Scanning** Overview of vulnerability scanning, Open Port / Service Identification, Banner / Version Check, Traffic Probe, Vulnerability Probe, Vulnerability Examples, OpenVAS, Metasploit. Networks Vulnerability Scanning - Netcat, Socat, understanding Port and Services tools - Datapipe, Fpipe, WinRelay, Network Reconnaissance – Nmap, THC-Amap and System tools. Network Sniffers and Injection tools – Tcpdump and Windump, Wireshark, Ettercap, Hping Kismet
UNIT 2	**Network Defense tools** Firewalls and Packet Filters: Firewall Basics, Packet Filter Vs Firewall, How a Firewall Protects a Network, Packet Characteristic to Filter, Stateless Vs Stateful Firewalls, Network Address Translation (NAT) and Port Forwarding, the basic of Virtual Private Networks, Linux Firewall, Windows Firewall, Snort: Introduction Detection System
UNIT 3	**Web Application Tools** Scanning for web vulnerabilities tools: Nikto, W3af, HTTP utilities - Curl, OpenSSL and Stunnel, Application Inspection tools – Zed Attack Proxy, Sqlmap. DVWA, Webgoat, Password Cracking and Brute-Force Tools – John the Ripper, L0htcrack, Pwdump, HTC-Hydra
UNIT 4	**Introduction to Cyber Crime and law** Cyber Crimes, Types of Cybercrime, Hacking, Attack vectors, Cyberspace and Criminal Behavior, Clarification of Terms, Traditional Problems Associated with Computer Crime, Introduction to Incident Response, Digital Forensics, Computer Language, Network Language, Realms of the Cyber world, A Brief History of the Internet, Recognizing and Defining Computer Crime, Contemporary Crimes, Computers as Targets, Contaminants and Destruction of Data, Indian IT ACT 2000.
UNIT 5	**Introduction to Cyber Crime Investigation** Firewalls and Packet Filters, password Cracking, Keyloggers and Spyware, Virus and Warms, Trojan and backdoors, Steganography, DOS and DDOS attack, SQL

	injection, Buffer Overflow, Attack on wireless Networks

OUTCOME:

At the end of this course, students will demonstrate ability to:

- Show various cyber security concepts like vulnerability, injection tools
- Understands the applications of cyber security.
- Understands with utilization of cyber security concepts in cyber crime.
- Show the comprehensive knowledge of cyber security.
- **MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

- H = Highly Related; M = Medium L = Low

Reference Books:

1. Anti-Hacker Tool Kit (Indian Edition) by Mike Shema, Publication Mc Graw Hill.

2. Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives by Nina Godbole and Sunit Belpure, Publication Wiley

BCO 085A	DIGITAL IMAGE PROCESSING	4-0-0 [4]

OBJECTIVE:

- To cover the basic theory and algorithms that are widely used in digital image processing
- To expose students to current technologies and issues that are specific to image processing system
- To develop hands-on experience in using computers to process images
- To familiarize with MATLAB Image Processing Toolbox

UNIT 1	**Introduction and Fundamentals** Motivation and Perspective, Applications, Components of Image Processing System, Element ofVisual Perception, A Simple Image Model, Sampling and Quantization, Steps in image Processing, Image sensing and Acquisition, Relationships between pixels. **Image Enhancement in Spatial Domain** Introduction; Basic Gray Level Functions – Piecewise-Linear Transformation Functions: Contrast Stretching; Histogram Specification; Histogram Equalization; Local Enhancement; Enhancement using Arithmetic/Logic Operations – Image Subtraction, Image Averaging; Basics of Spatial Filtering; Smoothing - Mean filter, Ordered Statistic Filter; Sharpening – The Laplacian.
UNIT 2	**Image Enhancement in Frequency Domain** Fourier Transform and the Frequency Domain, Basis of Filtering in Frequency Domain, Filters –Low-pass, High-pass; Correspondence Between Filtering in Spatial and Frequency Domain;Smoothing Frequency Domain Filters – Gaussian Lowpass Filters; Sharpening Frequency DomainFilters – Gaussian Highpass Filters; Homomorphic Filtering. **Color Image processing** Color Fundamentals, color models-RGB, CMY, CMYK, HSI, pseudocolor image processing-Intensity slicing, color Transformations- Formation, Color complements, color silicing, color image smoothing and sharping
UNIT 3	**Image Restoration** A Model of Restoration Process, Noise Models, Restoration in the presence of Noise only-SpatialFiltering – Mean Filters: Arithmetic Mean filter, Geometric Mean Filter, Order Statistic Filters –Median Filter, Max and Min filters; Periodic Noise Reduction by Frequency Domain Filtering –Bandpass Filters; Minimum Mean-square Error Restoration.
UNIT 4	**Morphological Image Processing** Introduction, Logic Operations involving Binary Images, Dilation and Erosion, Opening and Closing,Morphological Algorithms – Boundary Extraction, Region Filling, Extraction of ConnectedComponents, Convex Hull, Thinning, ThickeningAlgorithms to Establish Correspondence, Algorithms to Recover Depth
UNIT 5	**Segmentation** Introduction, Region Extraction, Pixel-Based Approach, Multi-level Thresholding, LocalThresholding, Region-based Approach, Edge and Line Detection: Edge Detection, Edge Operators,Pattern Fitting Approach, Edge Linking and Edge Following, Edge Elements Extraction byThresholding, Edge Detector Performance, Line Detection, Corner Detection.

OUTCOME:

At the end of the course, the student should be able to:

- Cover the basic theory and algorithms that are widely used in digital image processing
- Expose students to current technologies and issues that are specific to image processing system
- Develop hands-on experience in using computers to process images
- Familiarize with MATLAB Image Processing Toolbox

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

- H = Highly Related; M = Medium L = Low

References:

1. Digital Image Processing 2nd Edition, Rafael C. Gonzalvez and Richard E. Woods. Published by: Pearson Education.
2. Digital Image Processing and Computer Vision, R.J. Schalkoff. Published by: John Wiley and Sons, NY.
3. Fundamentals of Digital Image Processing, A.K. Jain. Published by Prentice Hall, Upper Saddle River, NJ.

BCO 086B	MACHINE LEARNING	4-0-0 [4]

Course Objectives

- To understand the basic concepts of learning and decision trees.
- To understand the neural networks and genetic algorithms
- To understand the Bayesian techniques
- To understand the instant based learning
- To understand the analytical learning and reinforced learning

UNIT 1	INTRODUCTION, CONCEPT LEARNING AND SUPERVISED LEARNING ALGORITHMS: Introduction, Types of learning, Learning Problems – Designing Learning systems, Perspectives and Issues – Concept Learning – Version Spaces and Candidate Elimination Algorithm , Linear Regression Model, Naïve Bayes Classifier, Decision Tree, K Nearest Neighbor, Logistic Regression, Support Vector Machine, Random Forest Algorithm.
UNIT 2	UNSUPERVISED LEARNING ALGORITHM: Clustering- K-means Clustering, Hierarchical Clustering, Probabilistic Clustering, Apriori Algorithm, Association Rule Mining, Gaussian Mixture Model, Expectation Maximization. ENSEMBLE LEARNING-Bagging, Boosting and Stacking
UNIT 3	REGULARIZATION- Overfitting, Underfitting, Bias-Variance trade off, Cost Function, Regularized Linear Regression and Regularized Logistic Regression, Model Selection and train/Validation/Test Sets, VC Dimension. STATISTICAL LEARNING- Feature Extraction, Principal Component Analysis, Singular Value Decomposition, Feature Selection and subset selection.
UNIT 4	NEURAL NETWORKS AND GENETIC ALGORITHMS Neural Network Representation – Problems – Perceptron – Multilayer Networks and Back Propagation Algorithms – Gradient Descent.
UNIT 5	ANALYTICAL LEARNING AND REINFORCED LEARNING Perfect Domain Theories – Explanation Based Learning – Inductive-Analytical Approaches - FOCL Algorithm – Reinforcement Learning – Task – Q-Learning – Temporal Difference Learning- Markov Decision Processes(MDP), Introduction to Natural Language Processing and Recommended System- Collaborative and Content based Filtering.

OUTCOME:On Completion of the course, the students will be able to

- Choose the learning techniques with this basic knowledge.
- Apply effectively neural networks and genetic algorithms for appropriate applications.
- Apply bayesian techniques and derive effectively learning rules.
- Choose and differentiate reinforcement and analytical learning techniques

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

H = Highly Related; M = Medium L = Low

Required Texts:

- Machine Learning, Tom Mitchell, McGraw Hill, 1997, ISBN 0-07-042807-7.

TEXT BOOK:

1. Tom M. Mitchell, "Machine Learning", McGraw-Hill Education (INDIAN EDITION), 2013.

REFERENCES:

2. Ethem Alpaydin, "Introduction to Machine Learning", 2nd Ed., PHI Learning Pvt. Ltd., 2013.
3. T. Hastie, R. Tibshirani, J. H. Friedman, "The Elements of Statistical Learning", Springer; 1st edition, 2001.

Program Elective –V Lab

BCO 087A	CYBER SECURITY LAB	0-0-2 [2]

List of Experiments

1. TCP scanning using NMAP
2. Port scanning using NMAP
3. TCP / UDP connectivity using Netcat
4. Network vulnerability using OpenVAS
5. Web application testing using DVWA
6. Manual SQL injection using DVWA
7. XSS using DVWA
8. Automated SQL injection with SqlMap

BCO 088A	DIGITAL IMAGE PROCESSING LAB	0-0-2 [2]

List of Experiments

1. Write a code in MATLAB for read, Show, access image pixel and write an image in Matlab.
2. Write a MATLAB code that will do the following
    a. Read any gray scale image.
    b. Display that image
    c. Again display the image such that the pixels having intensity values below than 50 willdisplay as black and pixels having intensity values above than 150 will display as white. Andthe pixels between these will display as it is.
3. Write a MATLAB code for display Histogram of an image and Histogram Equalization
4. Write a MATLAB code for perform Arithmetic and logic operations on image
5. Write a MATLAB code for perform Spatial Filters
6. Write a MATLAB code for perform frequency domain Filters
7. Write a MATLAB code for color image smoothing and sharping.
8. Write a MATLAB code for perform Dilation and Erosion
9. Write a MATLAB code for performEdge and Line Detection, Corner Detection

BCO 089 B	MACHINE LEARNING LAB	0-0-2 [2]

List of Experiments

(The following tasks can be implemented in a language of your choice or any tools available)

1) Implement the CANDIDATE – ELIMINATION algorithm. Show how it is used to learn from training examples.

2)Write a program to implement Linear Regression and Logistic Regression

3) Implement the ID3 algorithm for learning Boolean–valued functions for classifying the training examples by searching through the space of a Decision Tree.

4) Design and implement Naïve Bayes Algorithm for learning and classifying TEXT DOCUMENTS.

5) Implement K-Nearest Neighbor algorithm to classify the iris data set. Calculate the score also.

6)Write a program to implement Support Vector Machine. Also discuss the confusion matrix and score of model.

7) Apply EM algorithm to cluster a set of data and also apply K-Means algorithm on the same data set to compare two algorithms.

8)Build an Artificial Neural Network by implementing Back-Propagation algorithm and test the same using appropriate data set.

9)Implement the Non-Parametric Locally Weighted Regression Algorithm in order to fit data points. Select appropriate data set for your experiment and draw graph.

10)Build a Face detection system to recognize faces in a frame or image. You can use OpenCV for this task.

Program Elective –VI (VII Semester)

BCO 053A	PATTERN RECOGNITION	4-0-0 [4]

Objective:

- To understand concepts of Pattern Identification in Data Mining.
- To understand algorithms for Classification.
- To Use of Classifiers.

UNIT 1	Introduction- Paradigms for pattern recognition, Statistical and Syntactic pattern recognition, Soft and Hard computing schemes for pattern recognition. Statistical Pattern Recognition- Patterns and classes, Supervised, Semi-supervised, and Unsupervised classification
UNIT 2	Representation**-** Vector space representation of patterns and classes, patterns and classes as strings, Tree-based representations, Frequentitemsets for representing classes and clusters, Patterns and classes as logical formulas.
UNIT 3	Proximity Measures**-** Dissimilarity measures, metrics, similarity measures, Edit distance, Hausdorff metric between point sets, Kernel functions, Contextual and conceptual similarity between points.
UNIT 4	Dimensionality Reduction**-** Feature selection: Branch and bound, Sequential feature selection, Feature extraction: Fisher's linear discriminant, Principal components as features; Nearest Neighbor Classifiers- Nearest neighbor classifier, Soft nearest neighbor classifiers, Efficient algorithms for nearest neighbor classification, K-nearest neighbor classifier, minimal distance classifier, condensed nearest neighbor classifier and its modifications.
UNIT 5	Bayes Classifier- Bayes classifier, naïve Bayes classifier, Belief net; Decision Trees- Axis-parallel and oblique decision trees, Learning decision trees, Information gain and Impurity measures.

OUTCOMES: : At the end of the course, the student should be able to:

1. Concepts of Pattern Identification in Data Mining.
2. Understanding algorithms for Classification.
3. Use of Classifiers.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

H = Highly Related; M = Medium L = Low

Text Books:

1.V. Susheela Devi and M. NarasimhaMurty, *Pattern Recognition: An Introduction,* Universities Press,Hyderabad, 2011.

Reference Books:

1.R. O. Duda, P. E. Hart and D. G. Stork, *Pattern Classification,* John Wiley and Sons, 2000.
2.M. NarasimhaMurty and V. Susheela Devi, *Pattern Recognition,* NPTEL Web Course, 2011 (http://nptel.iitm.ac.in/courses.php?disciplineId=106).

BCO 054A	SOFT COMPUTING	4-0-0 [4]

Objective:

1. To understanding the concepts of Soft Computing.
2. To develop usage of Genetic Algorithms.
3. To understand supervised learning algorithms.

UNIT 1	Introduction - What is soft computing ?Differences between soft computing and hard computing, Soft Computing constituents, Methods in soft computing, Applications of Soft Computing.
UNIT 2	Introduction to Genetic Algorithms- Introduction to Genetic Algorithms (GA), Representation, Operators in GA, Fitness function, population, building block hypothesis and schema theorem.; Genetic algorithms operators- methods of selection, crossover and mutation, simple GA(SGA), other types of GA, generation gap, steady state GA, Applications of GA
UNIT 3	Neural Networks- Concept, biological neural system, Evolution of neural network, McCulloch-Pitts neuron model, activation functions, feedforward networks, feedback networks, learning rules – Hebbian, Delta, Percepron learning and Windrow-Hoff, winner-take-all.
UNIT 4	Supervised learning- Perceptron learning, single l layer/multilayer perceptron, linear separability, hidden layers, back propagation algorithm, Radial Basis Function network; Unsupervised learning - Kohonen, SOM, Counter-propagation, ART, Reinforcement learning, adaptive resonance architecture, applications of neural networks to pattern recognition systems such as character recognition, face recognition, application of neural networks in image processing.
UNIT 5	Fuzzy systems **-** Basic definition and terminology, set-theoretic operations, Fuzzy Sets, Operations on Fuzzy Sets, Fuzzy Relations, Membership Functions, Fuzzy Rules & Fuzzy Reasoning, Fuzzy Inference Systems, Fuzzy Expert Systems, Fuzzy Decision Making; Neuro-fuzzy modeling- Adaptive Neuro-Fuzzy Inference Systems, Coactive Neuro-Fuzzy Modeling, Classification and Regression Trees, Data Clustering Algorithms, Rulebase Structure Identification and Neuro-Fuzzy Control , Applications of neuro-fuzzy modeling.

OUTCOMES: At the end of the course, the student should be able to:

1. Understanding the concepts of Soft Computing.
2. Developing usage of Genetic Algorithms.
3. To understand supervised learning algorithms.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

H = Highly Related; M = Medium L = Low

Text Books:

1. S.N. Shivanandam, *Principle of soft computing*, Wiley. ISBN13: 9788126527410 (2011)
2. Jyh-Shing Roger Jang, Chuen-Tsai Sun, EijiMizutani, "Neuro-Fuzzy and Soft Computing", Prentice-Hall ofIndia, 2003.
3. George J. Klir and Bo Yuan, "Fuzzy Sets and Fuzzy Logic-Theory and Applications", Prentice Hall, 1995.
4. James A. Freeman and David M. Skapura, "Neural Networks Algorithms, Applications, and ProgrammingTechniques", Pearson Edn., 2003.

Reference Books:

1. Mitchell Melanie, "An Introduction to Genetic Algorithm", Prentice Hall, 1998.
2. David E. Goldberg, Genetic Algorithms in Search, Optimization & Machine Learning, Addison Wesley, 1997.

BCO 055A	GAME THEORY	4-0-0 [4]

Objective:

- To understand the concept of Game Theory.
- To develop strategies to solve different problems.
- To analyse the problem and strategies of their solution.

UNIT 1	Introduction- What is Game Theory? Definition of Games.Actions, Strategies, Preferences, Payoffs. Examples**;** Strategic Form Games - Strategic form games and examples: Prisoner's Dilemma, Bach or Stravinsky, Matching Pennies, Tragedy of Commons, Braess Paradox
UNIT 2	Dominant Strategy Equilibrium- Strongly dominant strategies, weakly dominant strategies, dominant strategy equilibrium; Examples of Prisoner's Dilemma and Vickrey Auction.
UNIT 3	Pure Strategy Nash Equilibrium- Best response strategies; Notion of pure strategy Nash equilibrium. Examples of Nash Equilibrium.Examples of Nash Equilibrium in popular games.Symmetric Games and Symmetric Equilibria; Mixed Strategy Nash Equilibrium- Randomization of Actions, Mixed strategy Nash equilibrium, Necessary and sufficient conditions for a Nash equilibrium. Examples of mixed strategy Nash equilibrium.Computing mixed strategy Nash equilibria. Related algorithmic issues.
UNIT 4	Two Player Zerosum Games (Matrix Games)- Max-minimization and Minmaximization. Saddle points.Nash equilibrium in matrix games.Mini-max theorem.Solution via linear programming. Examples; Extensive games with Perfect Information-Extensive games, Strategies and outcomes, Nash equilibrium, Subgame perfect equilibrium,findingsubgame perfect equilibria using backward induction. Allowing for simultaneous moves.Examples.
UNIT 5	Bayesian Games- Motivational Examples. Definition of a Bayesian Game and Bayesian Nash Equilibrium and examples.

OUTCOMES: At the end of the course, the student should be able to:

- Understanding the concept of Game Theory.
- Develop strategies to solve different problems.
- Analyse the problem and strategies of their solution

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

H = Highly Related; M = Medium L = Low

Text Books:

1. Martin Osborne. *An Introduction to Game Theory*.Oxford University Press, 2003.
2.Y. Narahari. *Essentials of Game Theory and Mechanism Design*.IISc Press, 2011

Reference Books:

1.Phiip D. Straffin, Jr. *Game Theory and Strategy*. The Mathematical Association of America, January 1993.
2.KenBinmore, *Fun and Games : A Text On Game Theory*, D. C. Heath & Company, 1992.

BCO 056A	WIRELESS SENSOR NETWORKS	4-0-0 [4]

OBJECTIVE: At the end of the course, the student should be able to:

- To understand the architecture of WSN.
- To identify the functionalities of layers in architecture.
- To analyse the working of main protocols of all layers.

UNIT 1	**INTRODUCTION** Challenges for wireless sensor networks, Comparison of sensor network with ad hoc network, Single node architecture – Hardware components, energy consumption of sensor nodes, Network architecture – Sensor network scenarios, types of sources and sinks, single hop versus multi-hop networks, multiple sinks and sources, design principles, Development of wireless sensor networks
UNIT 2	**PHYSICAL LAYER** Introduction, wireless channel and communication fundamentals – frequency allocation, modulation and demodulation, wave propagation effects and noise, channels models, spread spectrum communication, packet transmission and synchronization, quality of wireless channels and measures for improvement, physical layer and transceiver design consideration in wireless sensor networks, Energy usage profile, choice of modulation, Power Management.
UNIT 3	**DATA LINK LAYER** MAC protocols – fundamentals of wireless MAC protocols, low duty cycle protocols and wakeup concepts, contention-based protocols, Schedule-based protocols - SMAC, BMAC, Traffic-adaptive medium access protocol (TRAMA), Link Layer protocols – fundamentals task and requirements, error control, framing, linkmanagement.
UNIT 4	N**ETWORK LAYER** Gossiping and agent-based uni-cast forwarding, Energy-efficient unicast, Broadcast and multicast, geographic routing, mobile nodes, Data-centric routing – SPIN, Directed Diffusion, Energy aware routing, Gradient-based routing – COUGAR, ACQUIRE, Hierarchical Routing – LEACH, PEGASIS, Location Based Routing – GAF, GEAR, Data aggregation – Various aggregation techniques.
UNIT 5	**CASE STUDY:** Target detection tracking, Habitat monitoring, Environmental disaster monitoring, Practical implementation issues, IEEE 802.15.4 low rate WPAN, Operating System Design Issues, Introduction to TinyOS – NesC, Interfaces, UNITs, configuration, Programming in TinyOS using NesC, Emulator TOSSIM.

Course Outcome(CO):

CO1: Understand and analyze the fundamental of sensor networks and energy efficient sensor node and network architectures.

CO2: Understand and analyze the design issues in physical Layer.

CO3: Understand and analyze different communication protocols and their performance.

CO4: The broad education necessary to understand and analyze different routing strategies.

CO5: Understand the modern tool used for wireless sensors networks.

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	H											H		
CO2	M	H	L										H		
CO3	M	M											H		
CO4	M	M											H		
CO5					H									M	H

Text Books:

1. KazemSohraby, Daniel Minoli and TaiebZnati, "Wireless Sensor Networks Technology-Protocols and Applications", John Wiley & Sons, 2007.
2. Feng Zhao, Leonidas Guibas, "Wireless Sensor Networks: an information processing approach", Else vier publication, 2004.

Refererece Books:

1. C.S.Raghavendra Krishna, M.Sivalingam and Taribznati, "Wireless Sensor Networks", Springer publication, 2004.
2. HolgerKarl , Andreas willig, "Protocol and Architecture for Wireless Sensor Networks", John wiley publication, Jan 2006.
3. K.Akkaya and M.Younis, " A Survey of routing protocols in wireless sensor networks", Elsevier Adhoc Network Journal, Vol.3, no.3,pp. 325-349, 2005.
4. Philip Levis, " TinyOS Programming", 2006 – www.tinyos.net.
5. I.F. Akyildiz, W. Su, Sankarasubramaniam, E. Cayirci, "Wireless sensor networks: a survey", computer networks, Elsevier, 2002, 394 - 422.
6. Jamal N. Al-karaki, Ahmed E. Kamal, "Routing Techniques in Wireless sensor networks: A survey", IEEE wireless communication, December 2004, 6 – 28.

BCO 058A	COMPUTER VISION	4-0-0 [4]

OBJECTIVE:

- To understanding Computer vision.
- To analyse the basics of Machine and their processing.
- To learn algorithms in visualization of computer.

UNIT 1	Introduction- Machine vision systems, optics and lenses, image sensors, human vision and Neuro-visual model; Marr's paradigm; Imaging geometry - world co-ordinate system and camera co-ordinate system, co-ordinate transformations, projection geometry, camera calibration, radiometry.
UNIT 2	Early processing and image filtering: Noise removal, region segmentation, concept of primal sketch, scale space, edge detection and localization, edge linking, Hough transform, corner and junction detection. Reflectance map and photometric stereo: Image brightness and radiometry, image formation and surface reflectance under different conditions, reflectance map and bidirectional reflectance distribution function, photometric stereo recovering albedo and surface orientation, shape from shading.
UNIT 3	Range measurement and recovering scene geometry: Binocular technique stereo pair, epipolar line and plane, Stereo matching, photogrammetry, monocular technique - texture processing and shape from texture, depth from focusing and symmetry, different range finder (active) - laser range finder, light-stripe method.
UNIT 4	Motion estimation: Motion field, optical flow - smoothness, boundary conditions, discontinuities of optical flow, block based method, pre-recursive method, Bayesian method, Motion segmentation method, motion from points and lines, token tracking, stereo and motion tracking, use of Kalman filter, focus of expansion, structure from motion, motion compensated filtering and restoration, video compression, active and passive surveillance.
UNIT 5	Representation and analysis of polyhedral scene: understanding line drawings, gradient and dual space, generalized cylinder, volumetric representation, edge and junction labeling; Labeling and recognition of scene objects; Construction of model-base and visual learning, model based recognition system - Acronym, model based recognition from sparse range data, 3D model based vision system, scene understanding. Special systems for computer vision: Visual information processing architecture, language and control, Applications

OUTCOMES: At the end of the course, the student should be able to:

- Understanding Computer vision.
- Analyse the basics of Machine and their processing.
- Learn algorithms in visualization of computer.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

H = Highly Related; M = Medium L = Low

Text Books:

1. D. H. Ballard and C. M. Brown: Computer Vision, Prentice Hall, New York,1986.
2. R. M. Haralick, L. G. Shapiro: Computer and Robot Vision, Addison-Wesley Pub Co, reading, Mass., 1992.

Reference Books:

1. Y. Shirai: Three-Dimensional Computer Vision, Springer-Verlag Berlin, 1988.
2. B. K. P. Horn: Robot Vision, MIT Press, Cambridge, 1986.

Open Elective (VII Semester)

BCO 001A	SOFTWARE ENGINEERING	4-0-0 [4]

Objective

- To learn about generic models of software development process.
- To understand fundamental concepts of requirements engineering and Analysis Modeling.
- To understand the different design techniques and their implementation.
- To learn various testing and maintenance measures

UNIT 1	Introduction- Introduction to Software Engineering, Software Components, Software Characteristics, Software Crisis, Engineering aspects of Software production – necessity of automation .Job responsibilities of Programmers and Software Engineers as Software developers.Software Development Life Cycle (SDLC)
UNIT 2	Process Models and Program Design Techniques- Software Development Process Models – Code & Fix model, Waterfall model, Incremental model, Rapid Prototyping model, Spiral (Evolutionary) model.Software Requirement Specifications (SRS), Management of User Needs, Data Flow Diagrams, Entity Relationship Diagrams, Decision Tables, SRS Document, Design Techniques – Structured Programming, Coupling and Cohesion, Abstraction and Information Hiding, SoftwareModeling Tools –Data flow Diagrams, UML and XML.
UNIT 3	Software Testing: Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Verification and Validation: Testing of Software Products – Black-Box Testing and White-Box Testing, Static Analysis, Symbolic Execution and Control Flow Graphs –Cyclomatic Complexity.
UNIT 4	Software Project Management: Management Functions and Processes, Project Planning and Control, Organization and Intra-team Communication, Risk Management. Software Cost Estimation – underlying factors of critical concern. Metrics for estimating costs of software products – Function Points. Techniques for software cost estimation –Expert judgment, Work break-down structure and Process breakdown structure, COCOMO and COCOMO-II.
UNIT 5	Software Maintenance,Need for Maintenance, Categories of Maintenance, An Overview of CASE Tools. Advanced Topics: Support environment for Development of Software Products. Representative Tools for Editors, Linkers, Interpreters, Code Generators, Debuggers. Tools for Decision Support and Synthesis, Configuration control and Engineering Databases.

Course Outcome (CO):

At the end of this course studentswill have:

CO1: An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.
CO2: An ability to identify, formulates, and solve engineering problems.
CO3: An understanding of professional and ethical responsibility.
CO4: The broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	L		H		M						L				
CO2		H		H								M	M		
CO3								H	M	L				H	
CO4	L		L			L	M								L

H = Highly Related; M = Medium L = Low

Text Books:

1. Fundamentals of Software Engineering – Carlo Ghezziet. Et.al.
2. Software Engineering – Design, Reliability Management – Pressman.

Reference Books:

9. Software Engineering – Ian Sommerville.
10. Software Engineering - Shoeman.
11. Software Engineering with Abstraction – Berzins and Luqi
12. Pankaj Jalote, Software Engineering, Wiley

BCO 041A	**Multimedia Computing**	**3:0:0 [3]**

Course Objective:

- To present a step-by-step approach to multimedia systems design.
- To introduce multimedia standards and compression and decompression technologies.
- To provide a detailed analysis of the various storage technologies.

UNIT 1	Introduction to Multimedia, Media and Data Streams: Medium, Main Properties of a Multimedia System, Traditional Data Streams Characteristics, Data Stream Characteristics for Continuous Media, Information Units; Sound/Audio: Basic Sound Concepts, MIDI, Elements of Speech, Speech Generation, Speech Analysis, Speech Transmission;
UNIT 2	Images andGraphics: Digital Image Representation, Image Format, Graphic Format, Image Synthesis, Image Analysis, Image Transmission; Video and Animation: Basic Concepts, Television: Conventional Systems, Enhanced Definition System, High Definition System, Transmission , Computer Based Animation: Animation Languages, Method of Controlling Animation Display of Animation, Transmission of Animation;
UNIT 3	Data Compression: Storage Space, Coding Requirement, Source, Entropy and Hybrid Encoding, Basic Compression Techniques, Lossy Sequential DCT-based Mode, Expanded Lossy DCT-based Mode, Lossless Mode, Hierarchical Mode, H.261, MPEG, DVI; Computer Technology: Communication Architecture, Hybrid Systems, Digital Systems;
UNIT 4	Multimedia Operating Systems: Real Time and Multimedia, Resource Management, Real Time Process Management in Conventional Operating Systems, Real Time Processing Requirement, Traditional Real Time Scheduling, Earliest Deadline First Algorithm, Rate Monotonic Algorithm, EDF and Rate Monotonic: Context Switches, EDF and Rate Monotonic: Processor Utilization, Preemptive versus Non-preemptive Task Scheduling, Scheduling of Continuous Media Tasks, Traditional File System: Disk Scheduling: Shortest-Seek-Time First, SCAN, C-SCAN, Multimedia File System: Disk Scheduling Algorithm;
UNIT 5	Synchronization: Basic Synchronization Issues, Intra and Inter-object Synchronization, Live and Synthetic Synchronization, Lip Synchronization Requirement, Pointer Synchronization Requirement, Elementary Media Synchronization, The Synchronization Reference Model, Multimedia Synchronization Specification Methods, Interval-based Specifications, Axes-based Specification, Control Flow-based Specification, Event-based Synchronization;

Course Outcome:

- Students will be capable of understanding different realizations of multimediatools and their usage.
- Students will be capable of implementing various multimedia standards and compression technologies
- Students will be capable of analyzing various storage technologies

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1			L			H									
CO2	H						H					M	M		
CO3		H	H	M	H				H	L	H	H	H	H	H
CO4	M							L							

Text Books:

- "Multimedia Computing Communications & Applications " by Ralf Steinmetz, Klara Nahrstedt, Pearson Education (2004)
- Principles of Multimedia by Parekh Ranjan, Tata McGraw-Hill(2007)

BCO 042A	INFORMATION RETRIEVAL	4-0-0 [4]

OBJECTIVE:

- To understand the difference between Data mining and Information retrieval.
- To describe methods for Term and Document frequency calculations.
- To develop an understanding of Retrieval methods.

UNIT 1	Introduction- Basics of Information Retrieval and Introduction to SearchEngines; Boolean Retrieval-: Boolean queries, Building simple indexes, Processing Booleanqueries.
UNIT 2	Term Vocabulary and Posting Lists**-** Choosing document units, Selection ofterms, Stop word elimination, Stemming and lemmatization, Skip lists, Positional postingsand Phrase queries; Dictionaries and Tolerant Retrieval: Data structures for dictionaries,Wildcard queries, Permuterm and K-gram indexes, Spelling correction, Phonetic correction.
UNIT 3	Index Construction- Single pass scheme, Distributed indexing, Map Reduce,Dynamic indexing; Index Compression - Statistical properties of terms, Zipf's law, Heap'slaw, Dictionary compression, Postings file compression, Variable byte codes, Gamma codes.
UNIT 4	Vector Space Model- Parametric and zone indexes, Learning weights, Termfrequency and weighting, Tf-Idf weighting, Vector space model for scoring, variant tf-idffunctions.
UNIT 5	Computing Scores in a Complete Search System- Efficient score and ranking,Inexact retrieval, Champion lists, Impact ordering, Cluster pruning, Tiered indexes, Queryterm proximity, Vector space scoring and query operations.

Outcomes:

Upon completion of the subject, students will be able to:

CO1: Understand and apply the basic concepts of information retrieval.
CO2: Appreciate the limitations of different information retrieval techniques.
CO3: Write programs to implement search engines.
CO4: Evaluate search engines.
CO5: appreciate the different applications of information retrieval techniques in the Internet or Web environment.

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1		L	L								L		M	L	
CO2		L	H		L			L		L	M		M	M	
CO3		M	H								M		H	M	
CO4		M	M								M			M	
CO5		L	M	L							L	L	M	H	

Text Books:

1.C. D. Manning, P. Raghavan, and H. Schutze, *An Introduction to Information Retrieval,* CambridgeUniversity Press, 2009.

Reference Books:

1.R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval, Pearson Education*, 1999.

BCO 064A	CLOUD COMPUTING	4-0-0 [4]

OBJECTIVE: At the end of the course, the student should be able to:

1. To understand the architecture of Cloud.
2. To develop an understanding of various aspects of cloud computing.
3. To familiarize the students with fault Tolerance and security measures in cloud.

UNIT 1	**Understanding cloud computing:** Introduction to Cloud Computing - Benefits and Drawbacks - Types of Cloud Service Development - Deployment models
UNIT 2	**Cloud Architecture Technology and Architectural Requirements:** The Business Case for Clouds - Hardware and Infrastructure – Accessing the cloud – Cloud Storage – Standards- Software as a Service – Discovering Cloud Services Development tools. Three Layered Architectural Requirement - Provider Requirements - Service Centric Issues - Interoperability – QoS.
UNIT 3	Fault Tolerance - Data Management Storage and Processing - Virtualization Management - Scalability - Load Balancing - Cloud Deployment for Enterprises - User Requirement - Comparative Analysis of Requirement.
UNIT 4	**Security Management in Cloud:** Security Management Standards - Security Management in the Cloud Availability Management - SaaS Availability Management - PaaS Availability Management - IaaS Availability Management - Access Control - Security Vulnerability, Patch, and Configuration Management – Privacy in Cloud- The Key Privacy Concerns in the Cloud - Security in Cloud Computing.
UNIT 5	**Virtualization:** Objectives - Benefits - Virtualization Technologies - Data Storage Virtualization – Storage Virtualization – Improving Availability using Virtualization - Improving Performance using Virtualization- Improving Capacity using Virtualization.

OUTCOMES: At the end of the course, the student should be able to:

- Understand the architecture of Cloud.
- To develop an understanding of various aspects of cloud computing.
- Fault Tolerance and security measures in cloud.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

H = Highly Related; M = Medium L = Low

Text books:

1. *David S Linthicum, "Cloud Computing and SOA Convergence in your Enterprise A Step by Step Guide", Addison Wesley Information Technology Series.*
2. Anthony T Velte, Toby J.Velte, Robert Elsenpeter, "Cloud computing A Practical Approach ", Tata McGraw Hill Publication
3. Tim Mather, Subra Kumaraswamy, Shahed Latif, "Cloud Security and Privacy –

Reference Books:

1. An Enterprise Perspective on Risks and Compliance" , O'Reilly Publications, First Edition
2. Michael Miller, "Cloud Computing – Web-Based Applications that Change the Way You Work and Collaborate Online", Pearson Education, New Delhi, 2009.
3. Cloud Computing Specialist Certification Kit – Virtualization Study Guide

BCO 065A	ENGINEERING SYSTEM ANALYSIS AND DESIGN	4-0-0 [4]

OBJECTIVE:

- To understand the Engineering system and its elements.
- To learn methods of system analysis and approaches.
- To design and analyse through object oriented approach.

UNIT 1	INTRODUCTION- Systems, Elements of a system, Types of systems, Subsystems, Super systems, Need for system analysis and design, CASE tools for analysis and its limitations.
UNIT 2	System Analysis-Methods of system analysis, system development life cycle,Structured approach, development tools, data base and networking techniques.
UNIT 3	System design- Design technologies, Design principles, Design tools and methodologies, feasibility survey, conversion and testing tools, design management and maintenance tools.
UNIT 4	Object oriented analysis and design- Introduction, Object modeling, Dynamic modeling, functional modeling, UML diagrams and tools.
UNIT 5	Case studies- Developing prototypes for systems like, online exam management, Computer gaming and online website management.

OUTCOMES: At the end of the course, the student should be able to:

- To understand the Engineering system and its elements.
- Methods of system analysis and approaches.
- How to design and analyse through object oriented approach.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

H = Highly Related; M = Medium L = Low

Text Books:

1.Perry Edwards, “System analysis and design”, McGraw Hill international edition, 1993.
2. Len Fertuck, “System analysis and design with CASE tools”, Wm C. Brown Publishers, 1992.

Reference Books:

1. Er. V.K. Jain, “System analysis and design “, Dreamtech Press.

2. Kenneth E.Kendall and Julie E.Kendall, “System analysis and design”, Prentice Hall, India, 2007.