Walchand College of Engineering, Sangli.

(An Autonomous Institute)

Curriculum (Structure and Syllabud)

for

Third Year B.Tech Computer Science and Engineering

Academic Year 2016-17

Walchand College of Engineering, Sangli.

(An Autonomous Institute)

Teaching and Evaluation Scheme for Year 2016-17

Third Year UG Program in Computer Science & Engineering Semester I

Course Code	Course	Teaching Scheme				Evaluation Scheme			
						Component	Marks		
		L	T	P	Credits		Max.	Min. for passing	
1OE371	Open Elective I	3	0	0	3	ISE-1	10		40
						MSE	30		
						ISE-2	10		
						ESE	50	20	
2CS3**	Professional Electives I	3	0	0	3	ISE-1	10		40
						MSE	30		
						ISE-2	10		
						ESE	50	20	
3CS 301	Computer Architecture	3	0	0	3	ISE-1	10		40
						MSE	30		
						ISE-2	10		
						ESE	50	20	
3CS 302	Principles of Compiler Design	3	1	0	4	ISE-1	10		40
						MSE	30		
						ISE-2	10		
						ESE	50	20	
3CS 303	Design and Analysis of Algorithm	3	0	0	3	ISE-1	10		40
						MSE	30		
						ISE-2	10		
						ESE	50	20	
3CS 351	Design and Analysis of Algorithm Lab	0	0	2	1	ISE	50	20	
						ESE (POE)	50	20	
3CS 352	Object Oriented Modeling and Design	2	0	2	3	ISE	50	20	
						ESE (POE)	50	20	
3CS 341	Mini-Project-I	0	0	2	1	ISE	50	20	
						ESE (POE)	50	20	
2CS3**	Professional Electives II	2	0	2	3	ISE	100	40	
Total		19	1	8	24	Total Contact Hours/Week: 28 Hrs			

Professional Elective I		Professional Elective II	
3CS 311	Advanced Operating System	3CS 361	Computer Graphics and Visualization
3CS 312	Advanced Data Structure	3CS 362	Advanced Programming

Course Code	**Course Name**	**Offered by Department**
1OE 315 (3OE 302)	Remote Sensing & GIS, GPS	Civil
1OE 329 (3OE 304)	Manufacturing Engineering	Mechanical
1OE 330 (3OE 305)	Energy Engineering	Mechanical
1OE 331 (3OE 306)	Mechanisms and Machines	Mechanical
1OE 343 (3OE 307)	Electrical Machines	Electrical
1OE 357 (3OE 308)	Electronic Systems	Electronics
1OE 358 (3OE 311)	Fundamental of Analog and Digital Communication	Electronics
1OE 371 (3OE 309)	Software Engg. & Database Essentials	CSE
1OE 385 (3OE 310)	Internet of Things	IT

Note: Course code in the bracket indicates pre-revised code

Walchand College of Engineering, Sangli.

(An Autonomous Institute)

Teaching and Evaluation Scheme for Year 2016-17

Third Year UG Program in Computer Science & Engineering Semester II

Course Code	Course	Teaching Scheme				Evaluation Scheme			
						Component	Marks		
		L	T	P	Credits		Max.	Min. for passing	
1OE 378	Open Elective II	3	0	0	3	ISE-1	10		40
						MSE	30		
						ISE-2	10		
						ESE	50	20	
3CS 321	Distributed & Cloud Computing	3	0	0	3	ISE-1	10		40
						MSE	30		
						ISE-2	10		
						ESE	50	20	
3CS 322	Advanced Database System	3	0	0	3	ISE-1	10		40
						MSE	30		
						ISE-2	10		
						ESE	50	20	
3CS 323	Pervasive Computing	3	1	0	4	ISE-1	10		40
						MSE	30		
						ISE-2	10		
						ESE	50	20	
3CS 3**	Professional Elective III	3	0	0	3	ISE-1	10		40
						MSE	30		
						ISE-2	10		
						ESE	50	20	
3CS 3**	Professional Elective IV	3		0	3	ISE-1	10		40
						MSE	30		
						ISE-2	10		
						ESE	50	20	
3CS 371	Advanced Database System Lab	0	0	2	1	ISE	50	20	
						ESE POE	50	20	
3CS 372	Internet and Web Programming	2	0	2	3	ISE	50	20	
						ESE	50	20	
3CS 342	Mini-Project-II	0	0	2	1	ISE POE	50	20	
						ESE POE	50	20	
Total		20	1	6	24	Total Contact Hours/Week: 27 Hrs			

Professional Elective III		**Professional Elective IV**	
3CS 331	Digital Image Processing	3CS 334	Soft Computing
3CS 332	Intelligent Systems	3CS 335	Advanced Computer Network Technology
3CS 333	Computer Modeling and Simulation	3CS 336	Software Testing and Quality Assurance

Course Code	Course Name	Offered by Department
1OE 308 (3OE 321)	Machine Foundations	Civil-APM
1OE 309 (3OE 322)	Theory of Structures	Civil-APM
1OE 336 (3OE 324)	Power Plant Engineering	Mechanical
1OE 337 (3OE 325)	Fabrication Technology	Mechanical
1OE 338 (3OE 326)	Mechanical Power Transmission	Mechanical
1OE 350 (3OE 327)	Renewable Energy	Electrical
1OE 364 (3OE 328)	Embedded Programming	Electronics
1OE 365 (3OE 331)	Signal Processing and Foundation	Electronics
1OE 378 (3OE 329)	Data Analytics	CSE
1OE 392 (3OE 330)	Web Design & Applications	IT

Note: Course code in the bracket indicates pre-revised code

Title of the Course: Open Elective I - Software Engineering and Database Essential(1OE371)	L	T	P	Cr
	3	-	-	3

Pre-Requisite Courses:-NIL

Textbooks:

1. PankajJalote, *“An integrated approach to S/W engineering”*,Narosa Publishers, 2nd Edition.
2. Abraham Silberschatz, Henry F. Korth and S. Sudarshan, *Database System Concepts*, Mc-Graw Hill, 4th Edition 2002 / 6th Edition 2011
3. PankajJalote, “Software Project Management in practice”, Pearson education

References:

1. Roger S. Pressman, *“Software Engineering: Practitioner’s Approach”*. McGraw Hill
2. Raghu Ramakrishnan and Johannes Gehrke, *Database Management Systems*, 3rd Edition. 2002

Course Objectives :

1. Understand importance of engineering approach to software development and comprehend the knowledge of software processes & models practiced at IT industries.
2. Be acquainted with the SDLC phases in detail and appreciate the importance of software quality by virtue of software testing methods.
3. To use conceptual designs to prepare database schemas.
4. To understand the relational model and the theoretical issues associated with relational database Design.
5. To learn SQL and Database Architectures.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**	**Bloom’s Cognitive**	
		level	**Descriptor**
CO1	explain proficiency to undertake software projects based on software engineering practices.	2	Understanding
CO2	summarizing the spirit of team-working in SDLC phases & project planning benefits.	2	Understanding
CO3	describe the conceptual designs of Database, identifies the need, analyze the problem and Design ER diagram as well as prepare the relational database schema.	1,4	Remembering, Analyzing
CO4	apply SQL to extract required information from the database. Compare, analyses various ways of writing the queries for a given problem and Differentiating database Architecture.	4	Analyzing.

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1			2			2	3		3		
CO2		1	3			3					1
CO3			1						2	3	3
CO4			1							3	

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weightage respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment, oral, seminar, test (surprise/declared/quiz), and group discussion.[One assessment tool per ISE. The assessment tool used for ISE 1 shall not be used for ISE 2]
MSE: Assessment is based on 50% of course content (Normally first three modules)
ESE: Assessment is based on 100% course content with70-80% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Module 1: Introduction Software Engineering Basics: Software Crisis, Need of software engineering approach. **Software Processes:** Software Processes: project management process, software development process models, Configuration management process, process management process.	**7 Hrs.**
Module 2: Software Quality & Project Planning **Notion of Software Quality:** Quality objectives, Need for improvement, Software quality factors, Quality standards, **Project Planning Basics:** Project management plan, Cost estimation, Project scheduling, Staffing and personnel Planning, Risk management.	**6 Hrs.**
Module 3: Software Development Phases Software Requirement Process, Design principles, Structured design methodology, Coding Standards, levels of Testing.	**6 Hrs.**
Module 4:Introduction and Database Modeling using ER Model ***Introduction*:** General introduction to database systems, its advantages and applications, Database System Architecture, Database users and Administrator, Data models, Database management system, Database languages, View of Database, Data Models. **ER *Model*:** Entity set, Entity types, attributes, Notations, Relationship sets, Relationship types, Keys- super key, candidate key, primary key, Extended Features of ER Model-Generalization, Specialization and aggregation	**6 Hrs.**
Module 5: Relational Model and SQL **Relational *Model*:** Structure of Relational Database, Reduction of ER model into Relational schemas, Schema-instance distinction, Key , Relational algebra, Tuple relation calculus, Domain relational calculus, Example queries, ***SQL*:** Introduction to SQL, Data definition statements with constraints, Insert, Update and Delete, Set Operations, Aggregate functions group by and having clauses, Nested Queries, Views, Joins.	**8 Hrs.**
Module 6:Database Architectures Centralized &Client-Server architectures, server system architecture, Architectures for parallel databases, Distributed database concepts, Homogeneous &	**6 Hrs.**

Heterogeneous databases, distributed data storage, data fragmentation, and replication and allocation techniques for distributed database.	

Module wise Measurable Students Learning Outcomes :
After the completion of the course the student should be able to:

Module 1: Introduction Software Engineering

- Awareness of Software processes & Models used at IT.

Module 2: Software Quality & Project Planning

- Understand quality parameters and standards.
- Know project planning phases and responsibilities.

Module 3: Software Development Phases

- As per SDLC phase understand requirement process and need of SRS artifact.
- Understand functional & non-functional requirements as well. Realize the importance of design aspects, concepts & methodology. Practices to learn how to draw DFD on requirements.
- Know testing concepts, levels of testing

Module 4: Introduction and Database Modeling using ER Model

- Understanding the concept of database system and its applications.
- Studying database system architecture and various database models.
- Understanding the problem statement and preparing the conceptual model using ER diagram.

Module 5: Relational Model and SQL

- Studying relational data model using any RDBMS.
- Extracting information from the database using SQL

Module 6: Parallel and Distributed Databases and C/S architectures

- Awareness of Database Architectures and its operation.

Title of the Course: Professional Elective I Advanced Operating System 3CS311	L	T	P	Cr
	3	0	0	3

Pre-Requisite Courses: Operating System

Textbook:

1. Operating System Concepts – A.Silberschatz , P. Galvin, G. Gagne Wiley Publication 8th Edition

References:

1. Distributed O.S Concepts and Design, P.K.Sinha, PHI
2. Advanced concepts in Operating Systems, Mukesh Singhal & N.G.Shivaratri, TMH
3. Distributed Computing, Sunita Mahajan, Seema Shah, OXFORD University Press
4. Distributed System Principles and Paradigms, Andrew S. Tanenbaum, 2nd edition , PHI
5. Distributed Systems, Colouris , 3rd Edition

Course Objectives :

1. To provide a high level overview of distributed operating systems and the networks that interconnects to them.
2. To describe the various methods for creating and managing distributed file systems.
3. To understand the basic need of synchronization in distributed OS and to resolve the different issues.
4. To explore the operating system requirements of multimedia data including CPU, disk scheduling and network management.
5. To examine the Linux process model, memory management and to illustrate how Linux schedules process and provide interprocess communication.
6. To explore the principles of Windows design and specific components of the system.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**	Bloom's Cognitive	
		level	Descriptor
CO1	understand the concept of distributed operating system (DOS), its different components and distributed file system.	2	Understanding
CO2	demonstrate the kernel level / API level working of Linux and Windows operating systems.	3	Applying
CO3	categories different synchronization algorithm in DOS and potential issues in multimedia operating system	4	Analyzing

CO-PO Mapping :

	a	b	c	d	e	f	g	h	i	j	k
CO1			1		1						
CO2				3	3						2
CO3				2							

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weightage respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment, oral, seminar, test (surprise/declared/quiz), and group discussion.[One assessment tool per ISE. The assessment tool used for ISE 1 shall not be used for ISE 2]
MSE: Assessment is based on 50% of course content (Normally first three modules)
ESE: Assessment is based on 100% course content with70-80% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Module 1: Distributed Operating System Motivation, Types of Network-based Operating Systems, Network Structure, Network Topology, Communication Structure, Communication Protocols, Robustness, Design Issues, An Example-Networking.	**Hrs.**
Module 2: Distributed File System Background, Naming and Transparency, Remote File Access, Stateful versus Stateless Service, File Replication, An Example : AFS.	**Hrs.**
Module 3: Distributed Synchronization Event Ordering, Mutual Exclusion, Atomicity, Concurrency Control, Deadlock Handling, Election Algorithm, Reaching Agreement.	**Hrs.**
Module 4: Multimedia Operating System Multimedia-introduction, Compression, Requirements of Multimedia Kernels, CPU Scheduling, Disk Scheduling, Network Management, An Example : CineBlitz	**Hrs.**
Module 5: Case Study – The Linux System History, Design Principles, Kernel Modules, Process Management, Scheduling, Memory Management, File Systems, Input and output, Inter-process Communication, Network Structure, Security.	**Hrs.**
Module 6: Case Study – Windows History, Design Principles, System Components, Environmental Subsystems, File System, Networking, Programmer Interface.	**Hrs.**

Module wise Measurable Students Learning Outcomes :
After the completion of the course the student should be able to:

Module 1

1. Understand the general structure of distributed operating system.

2. Summarize the design issues in DOS.

Module 2

1. Apply naming convention in distributed file system using various methods.

2. Demonstrate the real time example of distributed file system.

Module 3

1. Use different synchronization algorithms in DOS.

2. Implement various schemes for handling deadlocks, concurrency-control.

Module 4

1. Understand overview of multimedia OS

2. Apply techniques in kernel design, CPU/disk scheduling.

Module 5

1. Understand complete working of Linux OS

2. Evaluate the performance of OS and fine tuning by changing core algorithms.

Module 6

1. Understand complete working of Windows OS

2. Design and implement device drivers and kernel level API

Title of the Course: Professional Elective I -Advanced Data Structures (3CS312)	L	T	P	Cr
	3	-	-	3

Pre-Requisite Courses: Data Structures, Algorithms

Textbooks:

1. Peter Brass, “ Advanced Data Structures”, Cambridge University Press
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, “Introduction to Algorithms,” 3rd Edition, PHI, 2009

References:

1. Reinhard Diestel, “Graph Theory”, Spinger-Verlag, 2000
2. MIT Courseware by Erik Demaine

Course Objectives :

a. To be able to understand balanced search trees and tree structures for set intervals, transformations of data structures and data structures for strings.
b. To be able to learn and understand advanced concepts and types of heaps, graphs and hashing.
c. To be able to apply the knowledge, identify appropriate data structures to be used in real applications and analyse the algorithms.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**	Bloom’s Cognitive	
		Level	Descriptor
CO1	explain and describe the different advanced data structures.	2	Understanding
CO2	apply and demonstrate knowledge of advanced data structures such as trees, graphs etc. for solving real world problems	3, 4	Applying Analyzing
CO3	compare and evaluate the impact of data structure transformations done and performance of different searching techniques.	4, 5	Analyzing Evaluating

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1	3										
CO2				2	3						
CO3		1		3							

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weightage respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment, oral, seminar, test (surprise/declared/quiz), and group discussion.[One assessment tool per ISE. The assessment tool used for ISE 1 shall not be used for ISE 2]
MSE: Assessment is based on 50% of course content (Normally first three modules)
ESE: Assessment is based on 100% course content with70-80% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Module 1: Balanced Search Trees	**6 Hrs.**
AVL trees, B Trees, Lexical Search Trees, Red-black trees, Splay Trees	
Module 2: Tree Structures for Sets of Intervals	**7 Hrs.**

Interval Trees, Segment Trees, Trees for the Union of Intervals, Trees for Sums of Weighted Intervals, Trees for Interval-Restricted Maximum Sum Queries, Orthogonal Range Trees, Range-Counting and the Semigroup Model, kd-Trees and Related Structures	
Module 3: Heaps	**7 Hrs.**
Balanced Search Trees as Heaps, Array-Based Heaps, Heap-Ordered Trees and Half-Ordered Trees, Leftist Heaps, Skew Heaps, Binomial Heaps, Changing Keys in Heaps, Fibonacci Heaps, Double-Ended Heap Structures and Multidimensional Heaps	
Module 4: Data structure transformations and strings	**6 Hrs.**
Data Structure Transformations Making Structures Dynamic, Making Structures Persistent **Data Structures for Strings** Tries and Compressed Tries, Dictionaries Allowing Errors in Queries, Suffix Trees, Suffix Arrays	
Module 5: Hashing	**6 Hrs.**
Basic Hash Tables and Collision Resolution, Universal Families of Hash Functions, Perfect Hash Functions, Hash Trees, Extendible Hashing, Membership Testers and Bloom Filters	
Module 6: Selected Graph Problems	**7 Hrs.**
Network flows: Max flow – mincut theorem, Probabilistic methods – Markov’s inequality, Dynamic Graph Problems	

Module wise Measurable Students Learning Outcomes :
After the completion of the course the student should be able to:

Module 1:

- Identify the advantages of balanced search trees
- Apply the different operations on balanced trees

Module 2:

- Demonstrate and use different tree structures for sets of intervals

Module 3:

- Explain and manipulate heaps, binomial heaps, multidimensional heaps etc.

Module 4:

- Transform static data structure to dynamic and allow queries in old states of a dynamic data structure.
- Demonstrate and use various text processing techniques required in real world applications.

Module 5:

- Useand compare various static as well as dynamic hashing techniques.

Module 6:

- Demonstrate selected graph problems widely used for real world problem solving.

Title of the Course: Computer Architecture (3CS301)	L	T	P	Cr
	3	-	-	3

Pre-Requisite Courses: Processor Architecture

Textbooks:

1. Kai Hwang, Faye A. Briggs, "Computer Architecture and Parallel Processing" McGrawhill international Edition
2. Kai Hwang, "Advanced Computer Architecture", Tata McGrawhill Edition

References:

1. William Stallings, "Computer Organization and Architecture, Designing for performance" Prentice Hall, Sixth edition
2. Kai Hwang, Scalable Parallel Computing MGH
3. Harrold Stone, High performance computer Architecture

Course Objectives :

1. To introduce the different types of computer architectures.
2. To study representative systems from the view point of theory, technology, architecture, and software aspects.
3. To study system scalability in a broader sense to reflect the interplay among the Parallel architectures and Parallel Programming

Course Learning Outcomes:

CO	After the completion of the course the student should be able to	Bloom's Cognitive	
		level	Descriptor
CO1	explain different types of computer architectures	2	Understanding
CO2	illustrate different techniques to enhance the performance of parallel application on different architectures.	3	Applying
CO3	analyze and Compare the different parallel / multithreaded programming techniques.	4	Analyzing

CO-PO Mapping :

	a	b	c	d	e	f	g	h	i	j	k
CO1		2					3				
CO2	1	3	2	1							
CO3		2	2		1		1			2	

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weightage respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment, oral, seminar, test (surprise/declared/quiz), and group discussion.[One assessment tool per ISE. The assessment tool used for ISE 1 shall not be used for ISE 2]

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with70-80% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Module 1: Design and implement parallel application on different architectures. Necessity of high performance, Constraints of conventional architecture, Parallelism in uniprocessor system, Evolution of parallel processors, future trends, Architectural Classification, Applications of parallel processing, Instruction level Parallelism and Thread Level Parallelism, Vector processing concepts, Principles of scalable performance: Performance Metrics and Measures, Speedup Performance Laws	**6 Hrs.**
Module 2:Pipeline Architecture Principles and implementation of Pipelining, Classification of pipelining processors, General pipelining reservation table, Design aspect of Arithmetic and Instruction pipelining, Pipelining hazards and resolving techniques, Data buffering techniques, Job sequencing and Collision, Advanced pipelining techniques, loop unrolling techniques, out of order execution, software scheduling, trace scheduling, Predicated execution, Speculative loading, Register Stack Engine, Software pipelining, VLIW (Very Long Instruction Word) processor, Case study: Superscalar Architecture- Pentium/ Ultra SPARC	**10 Hrs.**
Module 3:Distributed Memory Architecture Loosely coupled and tightly coupled architectures, Cluster computing as an application of loosely coupled architecture. Examples –CM*	**4 Hrs.**
Module 4:Program and Network Properties Program Partitioning and Scheduling, Grain Sizes and Latency, Static Multiprocessor Scheduling, System Interconnect Architectures-Network Properties and Routing, Static Connection Networks, Dynamic Connection Networks	**6 Hrs.**
Module 5:Multithreaded Architecture Multithreaded architectures–principles of multithreading, Concepts of Dataflow computing, static and dynamic dataflow architectures. Dataflow operators, Dataflow language properties, advantages & potential problems	**4 Hrs.**
Module 6:Parallel Programming Models, Languages and Compilers Parallel Programming Models, Types and levels of parallelism, Data dependency analysis- Bernstein's condition, Hardware and Software Parallelism, Parallel Languages and role of Compilers-Language Features for Parallelism, Parallel Language Constructs, Optimizing Compilers for Parallelism. Code Optimization and Scheduling- Scalar Optimization with Basic Blocks, Local and Global Optimizations	**9 Hrs.**

Module wise Measurable Students Learning Outcomes :
After the completion of the course the student should be able to:

Module 1

1. Understand the parallel and pipeline processing
2. Evaluate the performance issues in parallel processing

Module 2

1. Understand pipeline architecture in detail
2. Interpret the design aspects of different components of pipeline architecture

Module 3

1. Understand the distributed memory architecture
2. Demonstrate the cluster computing deployment

Module 4

1. Understand the core concept of program partitioning and scheduling.
2. Design and implement different routing algorithms.

Module 5

1. Understand the basic principles of multithreading, data flow architectures.
2. Analyze potential problems in multithreading architecture.

Module 6

1. Understand parallel programming in detail.
2. Design and implement parallel algorithms on different parallel architectures.

Title of the Course: Principle of Compiler Design (3CS302)	L	T	P	Cr
	3	1	-	4

Pre-Requisite Courses: Processor Architecture, Data Structures, Theory of Computation.

Textbooks:

1. D.M. Dhamdhere, “Systems Programming and Operating Systems” Second revised Edition, 2005, Tata McGraw- Hill Publishing Company limited, New Delhi.
2. “Compilers - Principles, Techniques and Tools”, A.V. Aho, R. Shethi and J.D. Ullman, Pearson Education.

References:

1. John J Donavan, “System Programming”, Tata McGraw- Hill Publishing Company limited, New Delhi.
2. Sumitabha Das, “Unix Concepts and Administration”, TMGH, 3rd Edition.
3. “Compilers - Principles, Techniques and Tools”, A.V. Aho, R. Shethiand J.D. Ullman, Addison Wesley Publishing Company.

Course Objectives :

1. Build proficiency and insight to understand System Software’s role in program execution.
2. Mature logical design perspective for primitive System Programs such as Assemblers, Macro processors, Linkers and Loaders.
3. Study various phases like lexical analysis, syntax analysis, semantic analysis, Intermediate Code Generation and code generation, involved in compiler design.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**	Bloom’s Cognitive	
		level	Descriptor
CO1	discuss and construct the basics of system programs like editors, compiler, assemblers, linker, loader, interpreter, macro etc.	2,3	Understanding, Applying
CO2	compare and analyze various phases of compiler construction, verify constructed design for language	3, 4, 5, 6	Applying, Analyzing, Evaluating, Creating
CO3	study and show storage allocation and code generation strategy involved in compiler construction	3, 4	Applying, Analyzing

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1	1	2		3	1			1			1
CO2		2	3	1							1
CO3		2		3	1					1	

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weightage respectively.

Assessment	Marks

ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment, oral, seminar, test (surprise/declared/quiz), and group discussion.[One assessment tool per ISE. The assessment tool used for ISE 1 shall not be used for ISE 2]
MSE: Assessment is based on 50% of course content (Normally first three modules)
ESE: Assessment is based on 100% course content with70-80% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Module 1:Introduction and Assemblers Language Processing Activities, Fundamentals of Language Processing & Specifications, LPDTs, Elements of assembly language programming, A simple assembly scheme, Pass structure of assemblers, design of a two pass assembler	**6 Hrs.**
Module 2: Macro Processors Macros Facility, Macro Definition and call, Macro Expansion & Nested Macro Calls, Advanced Macro Facilities, Design of Macro Preprocessor	**6 Hrs.**
Module 3: Introduction to Compiler, Lexical and Syntax Analysis Introduction to Compiler and phases, Role of a Lexical analyzer, input buffering, specification and recognition of tokens, Role of Parser, Writing grammars for context free environments, Top-down parsing, Recursive descent and predictive parsers (LL), Bottom-Up parsing, Operator precedence parsing, LR, SLR and LALR parsers.	**8 Hrs.**
Module 4: Syntax Directed Translation Syntax directed definitions, construction of syntax tree, Bottom-up evaluation of S-attributed definitions, L-attributed definitions, Top-down translation, Bottom-up evaluation of inherited attributes.	**6 Hrs.**
Module 5: Run time environments & IC generation Source language issues, storage organization and allocation strategies, parameter passing, symbol table organizations and generations, dynamic storage allocations. Intermediate languages, declarations, assignment statements and Boolean expressions, case statements, back patching, procedure calls.	**7 Hrs.**
Module 6: Code Optimization & Generation Sources of optimization, Basic blocks and Flow graphs, Optimization of basic blocks, Loops in flow graphs, Code improving transformations, Issues in the design of a code generator, Run time storage management, Next-use information, Peephole optimization, Code generation algorithm, Register allocation and assignment, DAG	**6 Hrs.**

Module wise Measurable Students Learning Outcomes :
After the completion of the course the student should be able to:

Module 1: Introduction and Assemblers

1. Learner understands the Semantic Gap and notion of System Software's.
2. Also builds proficiency for bare programming aspects required for any language processor.
3. Learner understands Assemblers and Macro Processor functionalities and devices algorithm with proper

4. Learners builds ability in understanding design of 2 Pass Assembler & Macro Processor.

Module 2: Macro Processors

1. Learner understands Assemblers and Macro Processor functionalities and devices algorithm with proper data structure and other requirement and hardware constraints.
2. Learner builds Logical thinking and consequence thinking on design of algorithm.

Module 3: Introduction to Compiler, Lexical and Syntax Analysis

1. Understand concept of compiler.
2. Understand the token separation process in lexical analysis.
3. Understand the role of Syntax analyzer in the compilation process.
4. Understand and implement various parsing algorithms.

Module 4: Syntax Directed Translation

1. Understand the various three address code representation for intermediate code representation
2. Understand the importance of syntax directed translation in compiler design
3. Understand Top-down and Bottom-up translation.

Module 5: Run time environments & IC generation

1. Understand and follow the source language issues involved in compiler design.
2. Understand Storage allocation strategies used in compiler design process.

Module 6: Code Optimization & Generation

1. Understand various issues involved in code generation
1. Understand the algorithms used for code generation

Title of the Course: Design and Analysis of Algorithm (3CS303)	L	T	P	Cr
	3	-	-	3

Pre-Requisite Courses:-

Textbooks:

1. “Fundamentals of Computer Algorithms”, Horowitz, Sahni and Rajasekaran, Galgotia Publications.
2. “Design and Analysis of Algorithms”, Aho, Hopfcraft and Ullman, Addison Wesley.

References:

1. “Introduction to Algorithms”, Thomas Cormen, PHI Publication.
2. “Introduction to Design and Analysis of Algorithm”, Goodman, McGraw Hill.
3. “Introduction to the Design and Analysis of Algorithm”, R.C.T. Lee, S.S. Tseng, R.C. Chang, Tata McGraw Hill.

Course Objectives :

1. Learn fundamental concepts and key techniques for designing and analyzing algorithms.
2. Study and apply different algorithm design methods namely, greedy method, divide and conquer, dynamic programming and backtracking.
3. Study the Parallel architectures for designing parallel algorithms.
4. Design and analyze the complexities of various algorithms following above methods.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**	Bloom’s Cognitive	
		level	Descriptor
CO1	Perceive & Apply fundamental concepts in Designing and analyzing computer science algorithm (recursive and non-recursive)	2,3	Understanding Applying
CO2	Solve real world problems using algorithmic approaches like Divide and Conquer, Greedy, Dynamic Programming and Back tracking.	3	Applying
CO3	Inspect Efficient Algorithm for Tree and Graph traversal and differentiate based on their Complexity.	6,4	Creating Analyzing
CO4	Recognize the general principles and good algorithm design techniques for developing efficient computer algorithms. Decide among different types of data structures the best one for different types of problems.	4,5	Analyzing, Evaluating

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1	1	2	3	1							
CO2				3	3					1	
CO3		2		3							
CO4				3	2	2				2	2

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End

Semester Examination (ESE) having 20%, 30% and 50% weightage respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment, oral, seminar, test (surprise/declared/quiz), and group discussion.[One assessment tool per ISE. The assessment tool used for ISE 1 shall not be used for ISE 2]
MSE: Assessment is based on 50% of course content (Normally first three modules)
ESE: Assessment is based on 100% course content with70-80% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Module 1:Introduction and Divide and Conquer Method What is algorithm, Algorithm Specification, Performance Analysis, Divide and Conquer Method - The general method, Binary search, Finding the maximum and minimum, Merge sort, Quick sort.	**7 Hrs.**
Module 2:The Greedy Method The general method, Knapsack problem, Job sequencing with deadlines, minimum-cost spanning trees – Prim's and Kruskal's Algorithms, Optimal storage on tapes , Single source shortest paths.	**6 Hrs.**
Module 3:Dynamic Programming The general method, Multistage graphs, All pair shortest paths, Optimal binary search trees, 0/1 knapsack, Reliability design, Traveling Sales person problem.	**7 Hrs.**
Module 4:Backtracking The general method, 8-queen problem, sum of subsets, Knapsack Problem, Hamiltonian Cycle, Graph Coloring.	**6 Hrs.**
Module 5:Basic Traversal and Search Techniques and Polynomial Problems Techniques for Binary Trees, Techniques for Graphs – Breadth First Search & Traversal, Depth First Search & Traversal, AND/OR graphs, Connected components and Spanning Trees, Biconnected components and depth first search. NP Hard and NP Complete.	**7 Hrs.**
Module 6:Advanced topics in Algorithm Approximation algorithms, Randomized algorithms, Branch and Bound, Introduction to parallel Architectures-PRAM, MESH and Hypercube.	**6 Hrs.**

M Module wise Measurable Students Learning Outcomes :
After the completion of the course the student should be able to:

Module 1

1. To understand the fundamental concepts in Algorithm design.
2. To understand the various notations used in analysis of an Algorithm
3. To study Greedy method and analyze the complexity of various algorithms.

Module 2

1. To study Divide and Conquer method and analyze the complexity of various algorithms.

Module 3

1. To study Dynamic Programming method and identify the various problems which can be solved by using Dynamic programming.
2. Design efficient algorithms for various problems applying Dynamic programming method.

Module 4

1. To study Backtracking method and identify the various problems which can be solved by using Backtracking method.
2. Design efficient algorithms for above problems.

Module 5

1. To design Efficient Algorithm for Tree and Graph traversal techniques.
2. To understand the NP-Problems.

Module 6

1. To Study Advanced concepts in algorithm design like Randomized and Approximation Algorithms
2. To understand and apply Parallel architectures for designing effective algorithms.

Title of the Course: Design and Analysis of Algorithm Lab (3CS351)	L	T	P	Cr
	-	-	2	1

Pre-Requisite Courses: Data Structure, Programming Language

Textbooks:

1. *"Fundamentals of Computer Algorithms"*, Horowitz, Sahni and Rajasekaran, Galgotia Publications.
2. *"Design and Analysis of Algorithms"*,Aho, Hopfcraft and Ullman, Addison Wesley.

References:

1. *"Introduction to Algorithms"*, Thomas Cormen, PHI Publication.
2. *"Introduction to Design and Analysis of Algorithm"*, Goodman, McGraw Hill.
3. *"Introduction to the Design and Analysis of Algorithm"*, R.C.T. Lee, S.S. Tseng, R.C. Chang, Tata McGraw Hill.

Course Objectives :

1. Learn key techniques for designing and analyzing algorithms.
2. Study fundamental concepts and notations used in Algorithm design.
3. Study and apply different algorithm design methods namely, greedy method, divide and conquer, dynamic programming and backtracking.
4. Study the Parallel architectures for designing parallel algorithms.
5. Design and analyze the complexities of various algorithms following above methods.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**	Bloom's Cognitive	
		level	Descriptor
CO1	apply algorithmic approaches to solve the real world problems and analyze their complexity.	3,4	Applying Analyzing
CO2	demonstrate tree and graph traversal techniques.	2,3	Understanding Applying
CO3	discuss various algorithm designing techniques with their performance comparisons and use the appropriate one while developing the applications.	4, 5	Analyzing Evaluating

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1	2			3	3		1				
CO2	1	1							1		
CO3				3	1		2	3	3		2

Assessment:

In Semester Evaluation (ISE), and End Semester Examination (ESE) having 50% weightageeach.

Assessment	Marks
ISE	50
ESE	50

ISE is based on performance of student in laboratory, experimental write-up, presentation, oral, and test (surprise/declared/quiz). The course teacher shall use at least two assessment tools as mentioned above for ISE.
ESE: Assessment is based on performance and oral.

Laboratory Experiences:

Students will be given hands on experience to design and implement efficient and effective algorithms for various problems based on syllabus covered in the course Design and Analysis of Algorithm in the practical hours using any suitable programming language like C,C++,Java. The List of experiments may include 12 to 14 experiments from among the following-

1. To implement sorting algorithm using array as a data structure and analyse its time complexity for different values of *n*. The large number of elements may be generated using Random Number generator or may be stored in a file. (Quick Sort, Merge Sort)
2. To implement different search techniques using array and/or trees and analyze their time complexity. (Linear, Binary, Binary recursive)
3. To implement Fractional Knapsack problem and activity selection problem using Greedy method.
4. Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's& Prim’s algorithm and compare.
5. To apply Greedy method to solve problems of
    a) Job sequencing with deadlines
    b) Optimal storage on tapes
6. Implement the following using Dynamic Programming
    a) Matrix-chain multiplication
    b) Longest common subsequence
    c) Optimal binary search trees
7. To implement Strassen’s matrix multiplication algorithm
8. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.
9. Find a subset of a given set S = {s1,s2,.....,sn} of n positive integers whose sum is equal to a given positive integer d. For example, if S= {1, 2, 5, 6, 8} and d = 9 there are two solutions {1,2,6} and {1,8}. A suitable message is to be displayed if the given problem instance doesn't have a solution.
10. Implement any scheme to find the optimal solution for the Traveling Salesperson problem and then solve the same problem instance using any approximation algorithm and determine the error in the approximation.
11. Implement the following using Back Tracking
    a) 8-Queen's problem
    b) Hamiltonian cycle
    c) Graph coloring Problem
12. Write a program to
    a) Print all the nodes reachable from a given starting node in a digraph using BFS method.
    b) Check whether a given graph is connected or not using DFS method.
13. Implement All-Pairs Shortest Paths Problem using Floyd's algorithm. Parallelize this algorithm by creating multiple threads and determine the speed-up achieved.
14. Compare and evaluate the performance of different Randomization and Approximation algorithms.

Title of the Course: Object Oriented Modeling and Design (3CS352)	L	T	P	Cr
	2	-	2	3

Pre-Requisite Courses: Software Engineering, Programming Lab-I (OOP)

Textbooks:

1. Rambaugh, Premerlani, Eddy, Lorenson, *"Object Oriented Modeling and Design"*, ,PHI.
2. Grady Booch, JeamsRambaugh, IvarJacotson, *"The Unified Modeling Language User Guide"*, Addison Wesley

References:

1. Andrew High, *"Object Oriented Analysis and Design"*, TMG
2. Kahate, *"Practical Object Oriented Design with UML"*, Mark Priestley

Course Objectives :

1. Learn Object Oriented notations for building the analysis model for a software application.
2. Articulate learner with various design diagrams of UML for a particular requirement or an application.
3. Learner will be exposed to thorough concepts of UML diagrams for Case studies, using tools like UML-Umbrello/ Rational Rose suite.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**	Bloom's Cognitive	
		level	Descriptor
CO1	explain various object oriented modeling concepts and draw the different models of OMT methodology for an application.	2, 3	Understanding, Applying
CO2	understand and interpret UML Diagrams and compare it with other software development methodologies	2	Understanding
CO3	design UML Diagrams for IT applications using UML Tools.	6	Creating

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1		2	1	3			1		2		
CO2						1		2	2		3
CO3		3	2	2	3	2	2				3

Assessment:

In Semester Evaluation (ISE), and End Semester Examination (ESE) having 50% weightageeach.

Assessment	Marks
ISE	50
ESE	50

ISE is based on performance of student in laboratory, experimental write-up, presentation, oral, and test (surprise/declared/quiz). The course teacher shall use at least two assessment tools as mentioned above for ISE. ESE: Assessment is based on performance and oral.

Course Contents:

Module 1: Principles of Object Orientation Motivations for OOP	**4 Hrs.**
Modeling as a design technique, Objects, classes, links and associations, generalization and Inheritance, Aggregation, abstract classes, generalization as extension and restriction, multiple inheritance.	

Module 2: Dynamic and Functional Modeling Events and states, operations, nested state diagrams, concurrency,advanced dynamic modeling concepts, relation of object and dynamic models, Data Flow Diagrams, relation of functional to object and dynamic models.	**4 Hrs.**
Module 3: Design Methodology Analysis Overview, System design with examples, Object Design, combining the three models, designing Algorithms, design Optimization, implementation of controls, design association	**4 Hrs.**
Module 4: Structural Modeling using UML Classes, Relationships, Common mechanisms, Diagrams, Class Diagrams, Interfaces, Types and Roles, Packages, Instances and Object Diagram	**5 Hrs.**
Module 5: Behavioral Modeling using UML Interactions, Use cases, Use case diagram, Interaction Diagrams and Activity diagrams, Events and signals, State Machines, State chart diagrams	**4 Hrs.**
Module 6: Architectural Modeling using UML Components, Deployment, Collaboration, Component Diagrams, Deployment Diagrams	**4 Hrs.**

Module wise Measurable Students Learning Outcomes :
After the completion of the course the student should be able to:

Module 1: State and explain the stages of Object Modeling technique along with the object model in detail.
Module 2: Describe the elements dealing with flow of control, interactions and sequencing of operations and understand the usage of events and states to specify control
Module 3: After constructing a formal model of a real-world problem, specify the decisions to be made in the system design and object design phase
Module 4: Describe the importance of modeling a software intensive system and identify the things that need to be modeled in any given system and represent relationship between them by way of structural diagrams
Module 5: Describe how behavioral diagrams can be used to represent the functionality provided by the system being modeled
Module 6: Describe how the actual system can be built and organized in terms of different components (deployed on the nodes) inter-related with each other using component diagrams

Practical Assignments
It shall consists of 10-12 experiments from the following

1. Learning Setup and configuration of RSA/Rational Rose.
2. Study of class and instance diagrams in OMT Methodology.
3. Study of different relationships between classes in the object model.
4. Study of issues related to Functional and dynamic model in OMT Methodology.
5. Draw Use Case diagram for particular scenario.
6. Draw sequence diagram.
7. Draw collaboration diagram.
8. Draw class diagram.
9. Working with packages.

10. Adding attributes & operations to classes & relationship. (Types, association, dependencies, aggregation, Generalization, multiplicity)
11. Draw state transition diagram and Activity diagrams.
12. Setting Component view.
13. Setting Deployment View.
14. Code generation and validations.

Title of the Course: Title of the Course: Mini-Project-I (3CS341)	L	T	P	Cr
	-	-	2	1

Pre-Requisite Courses: Nil

Textbook: Nil

References: Nil

Course Objectives :

1. To use latest design and development tools
2. To undergo project management techniques and project design principles.
3. To implement the project with appropriate programming languages and testing tools
4. To develop analytical vision and skills to analyse, compare the outcome with other techniques

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**	Bloom's Cognitive	
		level	Descriptor
CO1	demonstrate present technological trends through seminar and presentation	1	Remembering
CO2	demonstrate the appropriate selection of software tool for project implementation.	2	Understanding
CO3	work in teams and participate in group activity of software development.	3	Applying
CO4	develop a software product and demonstrate its significance	4,5	Analyzing, Evaluating

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1		3									
CO2		3		3							
CO3				2		3	2				
CO4		3									3

Assessment:

In Semester Evaluation (ISE), and End Semester Examination (ESE) having 50% weightageeach.

Assessment	Marks
ISE	50
ESE	50

ISE is based on performance of student in laboratory, experimental write-up, presentation, oral, and test (surprise/declared/quiz). The course teacher shall use at least two assessment tools as mentioned above for ISE.
ESE: Assessment is based on performance and oral.

Course Contents:

1. Students should maintain a project log book containing weekly progress of the project
2. At the end of the semester project group should achieve all the proposed objectives of the problem statement.
3. The work should be completed in all aspects of design, implementation and testing.
4. Project report should be prepared and submitted in soft and hard form along with

all the code and datasets. 5. Group should demonstrate the work with various test cases and results obtained and explain future scope. 6. The group should participate in technical symposiums, paper presentations to demonstrate their work and findings in technical community.	
Module wise Measurable Students Learning Outcomes :---	

Title of the Course: Professional Electives II - Computer Graphics and Visualization (3CS361)	**L**	**T**	**P**	**Cr**
	2	**-**	**2**	**3**

Pre-Requisite Courses: Data Structures & Files, C/C++ Programming, Java Programming

Textbooks:

1. “Mathematical Elements for Computer Graphics”, David F. Rogers, J Alan, Adams, TMGH, 2nd Edition
2. “Procedural Elements for Computer Graphics”, David F. Rogers, TMGH, 2nd Edition
3. “Interactive Comp. Graphics, A Top-Down Approach using OpenGL”, Edward Angel, Pearson, 5th Edition

References:

1. “Principal of Interactive Computer Graphics”, Newman Sproull, MGH, 2nd Edition
2. “Computer Graphics, A Programming Approach”, Steven Harrington, MGH
3. “Computer Graphics”, Hearn, Baker, PHI, 2nd Edition
4. “Multimedia Making it Work”, Tay Vaughan, TMGH, 5th Edition

Course Objectives:

a. Understand components of computer graphics, its visualization and their applications.
b. Undergo the process of generating virtual images from virtual scenes using a pipeline of generate, compute, store/display; and identifying the critical choices applicable to solution designs.
c. Recognize the potential benefits of comp. graphics and be able to apply it within their core fields.
d. Identify the important qualities which distinguish different imaging formats & display technologies.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**	Bloom’s Cognitive	
		Level	Descriptor
CO1	explain theory of computer graphical systems and manipulate graphical data programmatically.	2,3	Understanding, Applying
CO2	graphically demonstrate the themes, ideas, graphics algorithms etc. through implementation and select the appropriate for a particular application.	3,4	Applying, Analyzing
CO3	compare different animation tools and techniques and use the appropriate one to create the animation through designing GUI and simulations for real world applications.	4,6	Applying, Creating.

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**K**
CO1	2	3									
CO2				2				1			
CO3				3		3	1				

Assessment:

Assessment	Marks
ISE	100

ISE is based on performance of student in laboratory, experimental write-up, presentation, oral, and test (surprise/declared/quiz). The course teacher shall use at least two assessment tools as mentioned above for ISE.

Course Contents:

Module 1: Introduction to Computer Graphics Latest input output devices : Graphics Tablets, Bar code readers, Cameras, MIDI keyboard, Touch pads, Scanner, Electronic Whiteboard, Webcam, CCTV, Monitor (LED, LCD, CRT etc), LCD Projection Panels, Head Phone, Speakers, Touch screen etc Scan conversion Techniques- Real Time, RLE, Frame buffers Visualization of basic mathematical objects- Point, Line, Circle – DDA &Bresenham's Techniques Antialising- polygon interiors, simple area antialising Halftoning- patterning, thresholding& error distribution, ordered dither	**5 Hrs.**
Module 2: Geometric Transformations Object representations & Transformations- 2D & 3D Affine transformations- Translation, scaling, rotation, reflection, shearing; multiple transformations Plane Geometric Projections- Parallel and Perspective Viewing	**4 Hrs.**
Module 3: Polygon Filling And Curves Polygon listing & filling criteria- ordered edge list representations Polygon filling algorithms- Edge fill, fence fill, edge flag and seed fill algorithms Curve Representation & Visualization- Non-parametric and parametric curves, Interpolation, Parabolic Blended curves, Bezier curves and B-spline curves	**4 Hrs.**
Module 4: Viewing and Virtual reality Window & Viewport Transformation, Window Clipping –Line subdivision, Midpoint subdivision Visibility & Hidden surface removal -Z Buffer algorithm, Warnock Algorithm Introduction to Virtual Reality: Concept, Forms of VR, VR applications, VR devices, Basics of VRML.	**4 Hrs.**
Module 5: Animation Animation- Principles & Techniques, Techniques of animation: Onion Skinning, Motion Cycling, masking, Flip Book animation, blue-screening, color cycling, morphing, Working with audio, Video streaming & editing, 2D graphics & animation with any one tool , animation on the web**.**	**5 Hrs.**
Module 6: OpenGL: A Primer The OpenGL API- primitives & attributes, indexed & RGB color models, GLUT, interaction, events & callbacks, picking, model-view matrix for geometric transformations Lighting & Shading	**4 Hrs.**

Minimum 8 experiments will be performed to understand functioning of Computer graphics & its visualization. The list contains;

1. Practical based on C/C++ graphics library.
2. Introductory OpenGL programming.
3. Visualization of Data Sets.
4. 2D Transformations.
5. 3D Transformations and animation.
6. Line/Circle generation algorithm.
7. Polygon filling algorithms.
8. Hidden line/surface elimination algorithms (Z Buffer).

9. Curve Generation (Cubic spline, Bezier).
10. Study of Multimedia-file formats. (BMP-JPG/WAV-MP3/DAT-MPG etc).
11. Visualization applications / Case tools/ animation using Multimedia concepts.

Module wise Measurable Students Learning Outcomes :
After the completion of the course the student should be able to:

The student after completing the course will be able to:

Module 1

1. Differentiate Vector & Raster graphics.
2. Draw graphic entities with maximum correctness over screen alias.

Module 2

1. Represent the object in plane/space coordinate system.
2. Animate the object with linear and rotational move.

Module 3

1. Color objects with different filling algorithms and compare their time & space requirements.
2. Display images on discrete computer screens with minimum possible errors.

Module 4

1. View objects in parallel, perspective mode; as well eliminate the invisible edges & surfaces.
2. Decide upon what to and where to display on the comp. screen.

Module 5

1. Draw axis/parameter dependent mathematical curve paths.
2. Understand their applications in the field of design, engineering, manufacturing, animation etc.

Module 6

1. Program with OpenGL platform.
2. Have primary efforts towards lighting, shading, rendering, texturing the objects.

Computer Usage / Lab Tool: C/C++/Java, OpenGL, Any one animation tool (Flash/Blender)

Laboratory Experiences: As listed Above

Independent Learning Experiences: Case Tools, Multimedia packages

Title of the Course: Professional Electives II Advanced Programming (3CS362)	**L**	**T**	**P**	**Cr**
	2	**-**	**2**	**3**

Pre-Requisite Courses: Data Structures, JAVA Programming Concepts

Textbooks:

1. Steven Holzner, “ JAVA 2 Programming Black Book”
2. Peyton-Jones, “The Implementation of Functional Programming Language”, Prentice Hall.

References:

1. Functional programming in Haskell course on NPTEL.
2. O'Reilly, “Learning Perl” ebook.
3. Functional programming in Java 8 at TutorialsPoint

Course Objectives :

1. To explain advanced java programming concepts.
2. To explore functional programming paradigm, advanced java features.
3. To develop small java application using MVC architecture and java beans to solve real world problems.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**	Bloom’s Cognitive	
		level	Descriptor
CO1	describe different java collections and their usage in programming	2	Understanding
CO2	demonstrate concepts of MVC architecture, Java Beans while constructing small application	3	Applying
CO3	make use of functional programming, advanced java concepts to develop solution to real world problems.	3 6	Applying Creating

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1	2										
CO2			2		3			1			
CO3					3						2

Assessment:

Assessment	Marks
ISE	100

ISE is based on performance of student in laboratory, experimental write-up, presentation, oral, and test (surprise/declared/quiz). The course teacher shall use at least two assessment tools as mentioned above for ISE.

Course Contents:

Module 1	**4 Hrs.**
Collections Map, Set, List, Array List, Vector, Linked List, Stack, Hash Set, Tree Set, Hash Map, Tree Map, Comparator, Iterators, SortedSet, SortedMap, Queue, Deque	

Module 2	**4 Hrs.**
Java Beans Introduction, Design Patterns for Properties, Design Patterns for Events, Methods and Design Patterns, Java Beans API, Beans with JSP	
Module 3	**4 Hrs.**
MVC Architecture Architecture, Model, View, Controller, Applications	
Module 4	**6 Hrs.**
Functional Programming in Java 8: Introduction to Functional Programming, Lambda Expressions, Functional Interfaces, Functions as a method argument, Method references, Default Methods.	
Module 5	**4 Hrs.**
Introduction to Haskell: Introduction to Haskell and the ghci interpreter, Defining functions, Lists, strings and tuples, Sorting. OR **Introduction to Perl:** Introduction, Basic I/O, Variables, and Scalar Data, Arrays, Lists, and Hashes, References and Control Structures.	
Module 6	**4 Hrs.**
Functional Programming in Haskell: Higher order functions on lists: map, filter, list comprehension, User defined datatypes, Abstract data types, Modules, Recursive data types, Search trees, Arrays, IO OR **Functional Programming in Perl:** Functions, File I/O and Debugging, Regular expressions, Html and CGI Programming, Packages, Sockets.	

Module wise Measurable Students Learning Outcomes :

After the completion of the course the student should be able to:

Module 1:

- Implement various Java collections classes and interfaces

Module 2:

- Understand use of JavaBeans for designing API.

Module 3:

- Use of MVC architecture in real time applications.

Module 4:

- Understand reasons of how functional programming differs from OOP.

Module 5:

- List and Define fundamental concepts of Functional Programming.

Module 6:

- Understand and learn, develop application using functional programming language.

Title of the Course: Open Elective II – Data Analytics (1OE378)	L	T	P	Cr
	3	0	0	3

Pre-Requisite Courses:
Familiar with high-school level linear algebra, and calculus. Knowledge of probability theory, statistics, and programming is desirable.

Textbooks:

1. Hastie, Trevor, et al.The elements of statistical learning.Vol. 2. No. 1. New York: springer, 2009.
2. Montgomery, Douglas C., and George C. Runger.Applied statistics and probability for engineers. John Wiley &Sons, 2010

References:

1. NPTEL:http://nptel.ac.in. A course by Dr. Balaraman Ravindran, Department of Computer Science and Engineering &Dr. Nandan Sudarsanam, Department of Management Studies, IIT Madras.
2. Basic Data Analysis Tutorial, by Jacob Whitehill, Department of Computer Science, University of the Western Cape, 24/07/2009 [UWCDataAnalysisTutorial.pdf]

Course Objectives :

1. To explore the fundamental concepts of data analytics.
2. To learn how to analyze various types of data using different techniques.
3. To have the hands-on with state of the art data analysis tools.
4. To design effective ways for communicating results to any types of users

Course Learning Outcomes:

After the completion of the course the student should be able to		Bloom's Cognitive	
		level	Descriptor
CO1	demonstrate knowledge of statistical data analysis techniques used in decision making.	2	Understanding
CO2	identify suitable data analysis methods for the analysis tasks at hand from a reasonably wide selection of methods.	3	Applying
CO3	employ cutting edge tools and technologies to analyze Big Data.	4	Analyzing

CO-PO Mapping :

	a	b	c	d	e	f	g	h	i	j	k
CO1	3	2		3							
CO2		3		1							
CO3				2	3						

Assessment:
Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weightage respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment, oral, seminar, test (surprise/declared/quiz), and group discussion.[One assessment tool per ISE. The assessment tool used for ISE 1 shall not be used for ISE 2]
MSE: Assessment is based on 50% of course content (Normally first three modules)
ESE: Assessment is based on 100% course content with70-80% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Module 1 **Introduction**: Classification of digital data, types of analysis, approaches, data pre-processing, Overview of Data Analytics Lifecycle. **Descriptive Statistics:** Descriptive Statistics, Probability Distributions.	**6 Hrs.**
Module 2 **Inferential Statistics :** Inferential Statistics through hypothesis tests, Permutation & Randomization Test **Regression & ANOVA :** Regression, ANOVA(Analysis of Variance)	**6 Hrs.**
Module 3 **Machine Learning: Introduction and Concepts :** Differentiating algorithmic and model based frameworks, Regression : Ordinary Least Squares, Ridge , Regression, Lasso Regression, K Nearest Neighbors, Regression & Classification	**7 Hrs.**
Module 4 **Supervised Learning with Regression and Classification techniques :** Bias-Variance Dichotomy, Model Validation Approaches, Logistic Regression Linear Discriminant Analysis, Quadratic Discriminant Analysis, Regression and Classification Trees, Support Vector Machines, Ensemble Methods: Random Forest Neural Networks, Deep learning	**7 Hrs.**
Module 5 **Unsupervised Learning and Challenges for Big Data Analytics:** Clustering, Associative Rule Mining, Challenges for big data analytics.	**6 Hrs.**
Module 6 **Prescriptive analytics:** Creating data for analytics through designed experiments, Creating data for analytics through Active learning, Creating data for analytics through Reinforcement learning. Introduction/demo of tools like Weka, R, Pig, Hadoop, HiveQL, Python etc.	**7 Hrs.**

Module wise Measurable Students Learning Outcomes :

After the completion of the course the student should be able to:

Module 1:

- Explain the ways of data representation and describe the different distributions of a random variable.

Module 2:

- Interpret the given sample data, analyze it and derive inferences from it using different tests.

Module 3:

- Apply regression techniques to perform data analysis on real world problems.

Module 4:

- Apply supervised learning with regression for classification of data.

Module 5:

- Describe the need of unsupervised learning and explain the challenges for big data analytics.

Module 6:

- Create data for analytics using different ways and use data analysis tools.

Title of the Course: Professional Elective III - Digital Image Processing (3CS331)	**L**	**T**	**P**	**Cr**
	3	**-**	**-**	**3**

Pre-Requisite Courses:

Textbook:

1. R. C. Gonzalez, R. E. Woods, *Digital Image Processing*, 2nd Edition. 2002, PHI
2. A. K. Jain, Fundamentals of Digital Image Processing, PHI

References:

1. Milan Sonka, Vaclav Hlavac, Boyle, *Digital Image Processing and Computer Vision*, Cengage Learning
2. S. Jayaraman, S. Esakkirajan, T. Veerkumar, *Digital Image Processing*, Tata McGrawHill
3. Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, *Digital Image Processing Using MATLAB*, 2nd ed.

Course Objectives :

1. To learn fundamental of digital image processing.
2. To learn the concepts of image enhancement, image segmentation, compression etc and apply the algorithms to build applications.
3. To compare various algorithms and select the appropriate for a particular application.
4. To create initial background of the area of Image Processing to excel in this stream for further research.
5. To develop engineering skills and intuitive understanding of the tools used in Image Processing.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**	Bloom's Cognitive	
		level	Descriptor
CO1	explain fundamental concepts of digital image processing, mathematical transforms, image enhancement, segmentation, morphology, compression and color imaging.	2	understanding
CO2	write algorithms and apply the concepts mathematically to interpret the results with justification	3	Applying,
CO3	compare different algorithms of image processing and apply them to solve real life problems.	4	Analyzing,

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1	1										
CO2				1							
CO3		1		2						3	

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weightage respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment, oral, seminar, test (surprise/declared/quiz), and group discussion.[One assessment tool per ISE. The assessment tool used for ISE 1 shall not be used for ISE 2]

MSE: Assessment is based on 50% of course content (Normally first three modules)
ESE: Assessment is based on 100% course content with70-80% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Module 1: Digital Image Fundamentals ***Introduction*****:** Concept, Fundamental Steps and Components of Image Processing System ***Digital Image Fundamentals*****:** Image Acquisition, A simple image model, Sampling and Quantization, Imaging Geometry, Different types of digital images	**5 Hrs.**
Module 2: Image Transforms 2D systems and Necessary Mathematical preliminaries, 2DOrthogonal and Unitary Transforms, DFT, KL-Transforms, Cosine, Hadamard Transforms, Introduction to Wavelet transforms	**7 Hrs.**
Module 3: Image Enhancement Point Processing, Basic Gray Level Transformations, Histogram Processing, Spatial domain Filtering, Frequency domain filtering	**7 Hrs.**
Module 4: Image Segmentation and Analysis Edge Detection – using first and second order derivatives, LoG, Canny edge detector, Boundary Extraction – Connectivity, Heuristic Graph Search, Hough Transform, Active Contour, Watershed Transform, Region-based Segmentation – region growing, region splitting and merging, Feature Extraction	**9 Hrs.**
Module 5: Binary and Color Image Processing *Binary Image Processing*: Binarisation, Mathematical Morphology, Standard Binary morphological operations *Color Image Processing*: Fundamentals, Color Models, Histogram Processing, Filtering	**5 Hrs.**
Module 6: Image Compression Fundamentals, Compression model, Lossless VsLossy Compression, Fundamentals of Information Theory, Run-length coding, Huffman coding, Dictionary-based compression, Predictive coding, Transform-based coding, Image Compression Standards	**6 Hrs.**

Module wise Measurable Students Learning Outcomes :
After the completion of the course the student should be able to:

Module 1
Describe the fundamental concepts of Image Processing and its applications.

Module 2
Explain Image Processing Transforms which play significant role in image enhancement, filtering, analysis and compression.

Module 3
Implement various techniques to improve the quality of an image.

Module 4
Explain segmentation which is one of the most important steps leading to image analysis, learning and implementing various methods to divide an image into parts or groups of pixels which are homogeneous with respect to some criterion.

Module 5

Describe fundamentals of binary and color image processing and its operations

Module 6

Explain the need of image compression i.e. the technique of reducing the amount of data required to represent a digital image.

Title of the Course: Professional Elective I - Intelligent Systems (3CS332)	**L**	**T**	**P**	**Cr**
	3	**-**	**-**	**3**

Pre-Requisite Courses: Exposure to concepts in discrete structures, probability/statistics and algorithmic analysis

Textbooks:

1. Stuart Russell and Peter Norvig, “Artificial Intelligence – A Modern Approach”, Prentice-Hall, 2010, 3rd edition
2. Elaine Rich and Kelvin Knight ,Nair , “Artificial Intelligence,” McGraw Hill Publication, 3rd edition.

References:

1. Janakiraman et al., “Foundations of Artificial Intelligence and Expert Systems”, Macmilan India Ltd.
2. Townsend,“Introduction to Turbo prolog”

Course Objectives :

1. To introduce the concepts of Artificial Intelligence (AI) with emphasis on its use to solve real world problems.
2. To explain the challenges inherent in building “intelligent systems”.
3. To explain core techniques and algorithms.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**	Bloom’s Cognitive	
		level	Descriptor
CO1	interpret concepts of Artificial Intelligence.	2	Understanding
CO2	apply basic knowledge representation, problem solving, and learning methods in AI	3	Applying
CO3	analyze the applicability of algorithms in solving particular engineering problems and building intelligent systems.	4	Analyzing
CO4	designing artificial intelligence system.	6	Creating

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1	3	2							3		
CO2		2								3	
CO3			3				2				2
CO4				2		1		1			2

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weightage respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment, oral, seminar, test (surprise/declared/quiz), and group discussion.[One assessment tool per ISE. The assessment tool used for ISE 1 shall not be used for ISE 2]
MSE: Assessment is based on 50% of course content (Normally first three modules)
ESE: Assessment is based on 100% course content with70-80% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Module 1:Introduction to Artificial Intelligence Introduction, History, Application, Approaches, Introduction to Agents	**4 Hrs.**
Module 2:Problem Solving Problem solving by searching, Uninformed and informed search, Constraint satisfaction problems	**8 Hrs.**
Module 3: Knowledge Representation and Logic Propositional Logic, Inference rules, First Order Logic, Rule based systems, Semantic nets, Frames	**8 Hrs.**
Module 4: Planning Introduction, Components of planning, Partial-order-planning, Graph plan, SATPLAN	**6 Hrs.**
Module 5: Reasoning Reasoning with uncertainty, Fuzzy reasoning, Bayes networks, Reasoning in semantic net	**6 Hrs.**
Module 6: Expert Systems and Machine Learning ES Characteristics, Architecture, Rule based ES, Rule Induction and Decision Trees, Natural Language Processing	**7 Hrs.**

Module wise Measurable Students Learning Outcomes :
After the completion of the course the student should be able to:

Module 1:

- understand the concept of AI, various approaches and the concept of agent.

Module 2:

- analyze a given problem and identify the most suitable strategy for the problem and find solution.
- formulate a problem description as a CSP

Module 3:

- represent a natural language description as statements in logic
- deduct new sentences by applying inference rules.
- represent knowledge base as a set of rules.
- represent a real life problem in terms of semantic network and frames.
- write elementary programs in prolog.

Module 4:

- use suitable planning algorithm to solve the problem.

Module 5:

- represent a problem in terms of probabilistic statements, Bayes net etc.

Module 6:

- understand steps in building expert systems.

- represent problem as learning problem, apply suitable algorithm and solve it.

design the processing steps required for a NLP task

Title of the Course: Professional Elective-III Computer Modeling and Simulation (3CS333)	L	T	P	Cr
	3	1	0	4

Pre-Requisite Courses: Engineering Mathematics, Programming skills in one or more of the following programming languages: Java, C, or C++

Textbooks:

1. “Mathematical Modeling and Computer Simulation”, by Daniel P. Maki, Maynard Thompson Brooks Cole; 1st edition, 2005
2. *“Simulation with Arena”* (5th Edition) by W. David Kelton, Randall P. Sadowski and Nancy B. Swets, 2010 (McGraw Hill)
3. Law, A.M., Kelton, W.D.: Simulation Modeling and Analysis. McGraw-Hill, New York, 2-nd edition, 1991. ISBN 0-07-100803-9.

References:

1. “*Discrete-Event Simulation,*” (4th Ed) by Banks, Carson, Nelson, and Nicol, 2005 (Prentice Hall)
2. Brewmaud, Markov Chains; With Gibbs Field , Monte Carlo Simulation & Ques, Springer Verlag
3.

Course Objectives :

1. Introduce computer simulation technologies and techniques, provides the foundations for the student to understand computer simulation needs, and to implement and test a variety of simulation and data analysis libraries and programs.
2. It focuses on identifying what is needed to build simulation software environments, and not just building simulations using preexisting packages.
3. Recognize concepts of modeling layers of society's critical infrastructure networks.
4. Build tools to view and control simulations and their results.

Course Learning Outcomes:

CO	After completion of the course student should be able to	Bloom’s Cognitive	
		level	Descriptor
CO1	to identify and explain the different approaches of modeling and simulation	2	Understanding
CO2	to design and analyze simulation models using different strategies	3	Applying
CO3	to perform simulation and validate its results	6	Creating

CO-PO Mapping :

	a	b	c	d	e	f	g	h	i	j	k
CO1	3			1							
CO2		3									2
CO3			2		3						

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weightage respectively.

Assessment	Marks
ISE 1	10

MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment, oral, seminar, test (surprise/declared/quiz), and group discussion.[One assessment tool per ISE. The assessment tool used for ISE 1 shall not be used for ISE 2]
MSE: Assessment is based on 50% of course content (Normally first three modules)
ESE: Assessment is based on 100% course content with70-80% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Module 1 : Simulation Basics:	**Hrs.**
The notion of Simulations, Types of Models and Simulation, Simulation clock, Handling Stepped and Event-based Time in, Discrete versus Continuous modeling, Numerical Techniques, Sources and Propagation of Error,	**7**
Module 2 : Queuing System:	**Hrs.**
Manual Simulations and Discussions, Elements and characteristics of queuing system, Model and types; Performance Criteria- Average Waiting Time, Average Queue Length, Utilization of Resources; Networks of queues, Rough-cut modeling; An illustration	**8**
Module 3 : Parallel and Distributed Simulations:	**Hrs.**
Partitioning the Data, Partitioning the Algorithms, Handling Inter-partition Dependencies; Generating Random Numbers, Generating Random Variates, Modeling Detailed Operations in Arena, More animation.	**7**
Module 4 : Statistical Analysis:	**Hrs.**
Output Central Limit Theorem, Confidence Intervals, Terminating Simulations, Intermediate Modeling and Steady State Statistical Analysis, Statistical Modeling Technique	**6**
Module 5 : Dynamic and Complex Model Simulations:	**Hrs.**
Finite States, Dynamics, Graph or Network Transitions Based Simulations, Actor Based Simulations, Mesh Based Simulations, Hybrid Simulations	**6**
Module 6 : Case Handlings:	**Hrs.**
Case Studies, Simulation Methodology, Verification and Validation, Simulation Results Analysis and Viewing Tools, Display Forms, Tables, Graphs, and Multidimensional Visualization, Terminals, X and MS Windows, and Web Interfaces, Validation of Model Results	**6**

Module wise Measurable Students Learning Outcomes :
After the completion of the course the student should be able to:

Module 1:	Understand the basics of simulation
Module 2:	Comprehend and analyze the impactof queuing system on the model
Module 3:	Solve and examine the issues involved in parallel and distributed simulations
Module 4:	Summarize and apply the statistical techniques for simulation
Module 5:	Apply different types of simulations appropriately
Module 6:	Visualize the simulation results with different case studies

Title of the Course: Professional Elective IV - Soft Computing (3CS334)	**L**	**T**	**P**	**Cr**
	3	**-**	**-**	**3**

Pre-Requisite Courses:

Textbooks:

1. *"Neural Networks, Fuzzy Logic and Genetic Algorithms"*,S. Rajasekaran, G.A.VijayalakshmiPai, PHI (ECE).

References:

1. MIT-OCW
2. *"Introduction to the Theory of Neural Computation"*, Hertz, Krogh, Palmer.
3. *"Artificial Neural Networks"*, B. Yegnanarayana, PHI.
4. *"Genetic Algorithms"*, David E. Goldberg, Addison Wesley

Course Objectives :

1. Understand comparative performance of soft and hard computing approaches.
2. Provide to students a sound foundation of mathematical, scientific and engineering principles to formulate, solve and analyze learning problems using soft computing.
3. Imbibe capability for innovation in soft computing.
4. Understand hybrid applications of ANN, Fuzzy and GA.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**	Bloom's Cognitive	
		level	Descriptor
CO1	interpret soft computing schemes using knowledge of discrete mathematics, data structures, theory of computer science and computer architectures.	2	Understanding
CO2	demonstrate machine learning processes.	3	Applying
CO3	compare and analyze soft computing schemes.	4	Analyzing
CO4	design & Evaluate for better schemes using soft computing	5,6	Evaluating, Creating

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1		2								3	
CO2			3	3					1		
CO3		3									2
CO4		2			3						2

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weightage respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10

ESE	50

ISE 1 and ISE 2 are based on assignment, oral, seminar, test (surprise/declared/quiz), and group discussion.[One assessment tool per ISE. The assessment tool used for ISE 1 shall not be used for ISE 2]
MSE: Assessment is based on 50% of course content (Normally first three modules)
ESE: Assessment is based on 100% course content with70-80% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Module 1 Fundamentals of Neural Networks	**6 Hrs.**
Basics, McCulloch-Pitts Model, Storage capacity, Optimization Problems.	
Module 2 Back propagation Networks	**8 Hrs.**
Back propagation learning, applications: Parity Problem, Encoder Decoder, NETtalk and DEC-talk, Character Recognition, Learning Time Sequences.	
Module 3 Unsupervised Learning	**4 Hrs.**
Introductions, ARTI, Kohonen’s Algorithm	
Module 4 Fuzzy Systems	**6 Hrs.**
Crisp logic, predicate logic, fuzzy logic, fuzzy rule based logic, defuzzification methods, application	
Module 5 Genetic Algorithm	**9 Hrs.**
Fundamentals, mathematical foundations, Data Structure, application	
Module 6 Hybrid Systems	**6 Hrs.**
Integration of neural networks, fuzzy logic and genetic algorithms	

Module wise Measurable Students Learning Outcomes :
After the completion of the course the student should be able to:

Module 1: Pattern memorizing Capacity of ANN Model.
Module 2: Use of back propagation network for pattern recognition.
Module 3: Use of unsupervised network for pattern clustering.
Module 4: Use of fuzzy logic for control and decision making applications.
Module 5: Use of Genetic algorithms for soft computing.
Module 6: Use of hybrid systems for efficient soft computing applications.

Title of the Course: Professional Elective IV - Advanced Computer Network Technology (3CS335)	**L**	**T**	**P**	**Cr**
	3	**-**	**-**	**3**

Pre-Requisite Courses: Data Communication and Computer networking.

Textbooks:

1. James Kurose and Keith Ross, "Computer Networking, A Top-Down Approach".

References:

1.Larry Peterson and Bruce Davie, "Computer Networks, A Systems Approach", Morgan Kauffman, 2011.
2.W. Richard Stevens, "Unix Network Programming", Eastern Economy Edition, PHI, 1992.
3. B.A. Forouzan, Data communication & networking, 5th Edition, Tata Mc-Graw Hills

Course Objectives :

1. To review established key abstractions, concepts and technologies
2. To study router design and mobility support in wireless networks
3. To study new developments and topics in computer networking

Course Learning Outcomes:

CO	After the completion of the course the student should be able to	Bloom's Cognitive	
		level	Descriptor
CO1	explain the key networking abstractions and concepts.	2	Understanding
CO2	design and Configure the routing protocols and understand the mobility in wireless networks.	6	Creating
CO3	demonstrate the knowledge of current and advanced networking technologies.	3	Applying

CO-PO Mapping :

	a	b	C	d	e	f	g	h	i	j	k
CO1		1									
CO2			1	3	1						
CO3				2	3						

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weightage respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment, oral, seminar, test (surprise/declared/quiz), and group discussion.[One assessment tool per ISE. The assessment tool used for ISE 1 shall not be used for ISE 2]
MSE: Assessment is based on 50% of course content (Normally first three modules)
ESE: Assessment is based on 100% course content with70-80% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Module 1 Basic networking concepts revisited	**7 Hrs.**
introduction to networks, layering and link layer, network layer, routing, end-to-end layer, congestion control	
Module 2 Quality of Service Networking	**6 Hrs.**
Routing and router design, scheduling and QoS, integrated and differentiated services, RSVP	
Module 3 Wireless networks and mobility :	**7 Hrs.**
Wireless networks and mobility supports, MAC protocol, routing, AODV, group communication, multicast	
Module 4 Overlay networks:	**7 Hrs.**
RON, P2P, CDN, Web caching, cross-layer optimizations, Emerging network types: DTN, 4G mobile networks (LTE, Wi-Max), Online social networks (OSN), wireless sensor networks (WSN) – cross-layer sensor data dissemination	
Module 5 Network Programming	**6 Hrs.**
TCP sockets, UDP sockets (datagram sockets), Server programs that can handle one connection at a time and multiple connections (using multithreaded server), Remote Method Invocation (Java RMI) -Basic RMI Process, Implementation details - Client-Server Application.	
Module 6 Advanced topics	**6 Hrs.**
Software Defined Networking. Data center networking. Network Virtualization. Network Function Virtualization. **Emerging applications** – VoIP, SIP, video over P2P	

Module wise Measurable Students Learning Outcomes :
After the completion of the course the student should be able to:

Module I
In depth understanding Layering and its design
Module II
Understand the design and implementation of routing algorithms
Module III
Understand wireless networks and mobility supports
Module IV
Familiar with overlay networks
Module V
Design and implementation of applications using socket programming.
Module VI
Familiar with recent developments in computer networking technology.

Title of the Course: Professional Elective IV - Software Testing and Quality Assurance (3CS336)	**L**	**T**	**P**	**Cr**
	3	**-**	**-**	**3**

Pre-Requisite Courses: Software Engineering

Textbook:

1. Dr.K.V.K.K.Prasad, *“Software Testing Tools* “
2. Desikan, Ramesh, “*Software Testing: principles and Practices*”, Pearson Education, ISBN

References:

1. Nina Godbole, “*Software Quality Assurance: Principles And Practice*”, Alpha Scienc International, Ltd (August 1, 2004)

2. Marnei L Hutcheson, “*Software testing fundamentals*- Methods & Metrics”, Wiley Publication
3. Fenton, Pfleeger, Thomson Brooks/Cole, “*Software Metrics: A Rigourous and practical Approach*”, ISBN 981-240-385-X
4. ISBN-10: 1842651765 ISBN-13: 978-1842651766

Course Objectives :

a. To Understand the Software Testing dearth and Quality Assurance aspects and techniques used in IT industry.
b. To Comprehend the hands on knowledge of various Software CASE tools used for Software Testing.
c. To learn the Testing process and stress on performance metrics.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**	Bloom’s Cognitive	
		level	Descriptor
CO1	build proficiency and articulate Software Testing Process.	1,2	Remembering, Understanding,
CO2	apply manual as well as automated Testing mechanisms.	3	Applying
CO3	analyze testing mechanisms by using various software tools.	4, 6	Analyzing, Creating

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1		2	3								2
CO2				2	2						
CO3			2		3						

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weightage respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment, oral, seminar, test (surprise/declared/quiz), and group discussion.[One assessment tool per ISE. The assessment tool used for ISE 1 shall not be used for ISE 2]

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with70-80% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Module 1: **Software Measurement**: Measurement in software engineering, Classifying software measures, applying the framework, software measurement validation	**6 Hrs.**
Module 2: **Software Testing Process and Planning**: Purpose of testing, difference between inspection and testing, testing v/s debugging, testing life cycle, Roles and responsibility in testing, test artifacts, test plan, the V model for testing, techniques, Metrics, Risk based testing, Test Automation, Types of testing.	**9 Hrs.**
Module 3: **Test Case Generation :** Simplify Test creation and Visualization with Storyboard testing, Understanding Test flow, Writing of Test Scripts.	**4 Hrs.**
Module 4: **Software Testing tools:** Need for Automated Testing tools, Taxonomy, Functional, Regression, Performance, Test Management, Source Code Testing and How to select testing tools.	**8 Hrs.**
Module 5: **Study of testing tools**: Win Runner, Load Runner, J-meter, Test director ,QTP, Rational Robot, Clearcase etc.	**7 Hrs.**
Module 6: **Software Quality Assurance**: Quality Concepts, Software Quality Assurance, Planning for SQA, Six Sigma Principles, Malcolm Baldridge Assessment ISO 9000,EdwardDeming's Principles, Total Quality Management, Product Quality Metrics, In-Process Quality Metrics, Software Maintenance	**5 Hrs.**

Module wise Measurable Students Learning Outcomes :

After the completion of the course the student should be able to:

Module 1:

Learner understands the framework imposed for measuring software quality through classification of different software metrics.

Module 2:

Learner unleashes the planning and process paradigm of Software Testing along with various aspects of Software Testing used in IT industry.

Module 3:

Learner exercises the specification of Test Scripts for Test Case generation.

Module 4:

Learner aware various automated Testing software used for black box testing.

Module 5:

Learner demonstrate use of automated testing tool viz Win Runner, Load Runner, J-meter, Test director ,QTP

Module 6:

Learners becomes known about the jargon expectations of Quality measures through mainly Software Testing which is imparted in real life software development.

Title of the Course: Distributed and Cloud Computing (3CS321)	**L**	**T**	**P**	**Cr**
	3	**-**	**-**	**3**

Pre-Requisite Courses: Operating system, network programming

Textbook:

1. “*Mastering Cloud Computing*”, RajkumarBuyya, Christian Vecchieola, S. ThamaraiSelvi, (McGrawHill)
2. “*The Grid Core Technologies*”, Maozhen Li, Mark Baker, (Wiley)

References:

1. “*Grid Computing*”, Joshy Joseph, Craig Fellenstein, (Pearson)
2. “*Cloud Computing for Dummies*”, J. Hurwitz, R. Bloor, M Kaufman, F. Halper, (Wiley)

Course Objectives :

1. To emphasize the fundamental ideas behind distributed /cloud Computing, the evolution of the paradigm, its applicability; benefits, as well as current and future challenges.
2. To explain basic concepts of virtualization, cloud services, cloud models and their role in elastic computing.
3. To investigate efficient cloud deployment model, virtualization technique for developing standard cloud application.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**	**Bloom’s Cognitive**	
		level	**Descriptor**
CO1	describe the architecture and taxonomy of distributed /grid computing, including models of inter-process communication.	2	Understanding
CO2	demonstrate concepts of Infrastructure, Platform and Software as a Service (IaaS, PaaS, SaaS) abstractions, and Public and Private Clouds.	3	Applying
CO3	distinguish among various cloud computing frameworks and virtualization techniques.	4	Analysis

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1	2										
CO2	1	3	3		3	2					
CO3				3						3	1

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weightage respectively.

Assessment	Marks
ISE 1	10
MSE	30

ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment, oral, seminar, test (surprise/declared/quiz), and group discussion.[One assessment tool per ISE. The assessment tool used for ISE 1 shall not be used for ISE 2]
MSE: Assessment is based on 50% of course content (Normally first three modules)
ESE: Assessment is based on 100% course content with70-80% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Module 1: Principles of distributed computing Eras of computing, Elements of distributed computing – General concepts and definitions, components of a distributed system, architectural styles for distributed computing, models for inter-process communication, Technologies for distributed computing – Remote procedure call, distributed object frameworks, service oriented computing.	**6 Hrs.**
Module 2: Principles of grid computing Introduction to grid, Open Grid Service Architecture (OGSA), Open Grid Service Infrastructure (OGSI), The Globus Toolkit 3 (GT3), OGSI.Net Middleware solution.	**7 Hrs.**
Module 3: Introduction to cloud computing Cloud computing at glance, historical developments, building cloud computing environments, computing platforms and technologies.	**6 Hrs.**
Module 4: Virtualization Introduction, characteristics of virtualized environments, Taxonomy of virtualization Techniques, Virtualization and cloud computing, Pros and Cons of virtualization, technology examples.	**7 Hrs.**
Module 5: Cloud Computing Architecture Introduction, cloud reference model, types of clouds, economics of the cloud, open challenges	**7 Hrs.**
Module 6: Aneka – Cloud Application Platform Framework overview, anatomy of the Aneka Container, building Aneka Clouds, cloud programming and management.	**6 Hrs.**

Module wise Measurable Students Learning Outcomes :

After the completion of the course the student should be able to:

Module 1

1. Classify and describe the architecture and taxonomy of distributed computing.
2. Apply different technologies of distributed computing including RPC, RMI and web services.

Module 2

1. Evaluate the different grid computing architectures and open standards.
2. Build and deploy the grid environment to solve complex problem.

Module 3

1. Have a comprehensive knowledge of cloud computing techniques,
2. Articulate the main concepts, key technologies, strengths and limitations of cloud computing.

Module 4

1. Compare and contrast the economic benefits delivered by various virtualization techniques based on application requirements, economic constraints and business requirements.
2. Explain virtualization and their role in elastic computing.
3. Characterize the distinctions between Infrastructure, Platform and Software as a Service (IaaS,

PaaS, SaaS) abstractions, and Public and Private Clouds.

Module 5

1. Classify and describe the architecture and taxonomy of cloud computing
2. Analyze the economics and open challenges in cloud.
3. Critique the consistency of services deployed from a cloud architecture

Module 6

1. Build and deploy the Aneka cloud platform.
2. Understand how to design and implement cloud-based applications.

Title of the Course: Advanced Database Systems (3CS322)	L	T	P	Cr
	3	-	-	3

Pre-Requisite Courses: Database Engineering

Textbooks:

1. Database system concepts – Silberschatz, Korth, Sudarshan – 6th Edition (MGH).
2. Database Management System – Raghu Ramkrishnan (MGH)

References:

1. Database Systems : A practical approach to design, implementation &Managemennt by Thomas Connolly & Carolyn Begg (Pearson) Third Edition
2. RamezElmasri and ShamkantNavathe, Fundamentals of Database Systems 2nd Ed, Benjamin Cummings, 1994.

Course Objectives :

1. To understand the fundamentals of object based databases, the database centric design issues involved in application development and the advances in database system.
2. To implement the complex and real world database applications.
3. To evaluate and analyze the different types of advanced databases.

Course Learning Outcomes:

After the completion of the course the student should be able to		Bloom's Cognitive	
		Level	Descriptor
CO1	describe the fundamental concepts involved in advanced databases.	1,2	Remembering, Understanding
CO2	build and demonstrate the acquired knowledge in specialized database	6,3	Creating , Applying
CO3	evaluate different types of advanced databases and analyze the issues in application development.	4	Analyzing

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1		2									
CO2			3	3							
CO3					1						

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weightage respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment, oral, seminar, test (surprise/declared/quiz), and group discussion.[One assessment tool per ISE. The assessment tool used for ISE 1 shall not be used for ISE 2]

MSE: Assessment is based on 50% of course content (Normally first three modules)
ESE: Assessment is based on 100% course content with70-80% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Module 1 : Object-Based Databases Overview, Complex Data Types, Structure Types and Inheritance in SQL, Table Inheritance, Arrays and Multiset Types in SQL, Object-Identity and Reference Types in SQL, Implementing O-R Features, Persistent Programming Languages, Object-Relational Mapping, Object-Oriented versus Object-Relational	**6Hrs.**
Module 2 : Application development & Administration Application Programs and User Interfaces, Web Fundamentals, Application Architectures, Rapid Application Development, Application Performance, Application Security. Performance Tuning, Performance Benchmarks, Other issues in Application Development, Standardization.	**8Hrs**
Module 3 : Distributed databases Homogeneous & heterogeneous databases, distributed data storage, distributed transactions, commit protocols, concurrency control in distributed databases, availability, distributed query processing, Heterogeneous distributed databases.	**6Hrs.**
Module 4 : Parallel Databases Introduction, I/O parallelism, inter-query parallelism, intra-query Parallelism, intra-operation parallelism, inter-operation parallelism, Query Optimization, Design of parallel systems, Parallelism on Multi-core Processor.	**5Hrs.**
Module 5 : Cloud Databases Introduction, Architecture, Data Models, NoSQL databases : Apache Cassandra, CouchDB and MongoDB, Comparison of Relational databases and Cloud databases, Challenges to develop Cloud Databases.	**8Hrs.**
Module 6 : Spatial, Temporal Data and Mobility Motivation, Time in Databases, Spatial and Geographic Data, Multimedia Databases, Mobility and Personal Databases.	**6Hrs.**

Module wise Measurable Students Learning Outcomes :
After the completion of the course the student should be able to:

Module 1

1. Apply the concept of object-relational model for design of complex data types
2. Compare between object oriented and objet relational data models

Module 2

1. Understand the different application architectures
2. Design and deploy the enterprise application
3. Evaluate the performance of enterprise application

Module 3

1. Understand the distributed database models
2. Study the distributed storage and transaction management
3. Implement the distributed enterprise application

Module 4

1. Apply the basic concept of parallel database to conventional databases
2. Evaluate the different modes of parallelism
3. Design the parallel database system

Module 5

1. Understand the architecture and data models of cloud databases
2. Evaluate the different open source cloud databases.

Module 6

1. Understand the advances in databases – database beyond the traditional models
2. Use the temporal/mobile/spatial data model to store non-relational data
3. Devise the hybrid solution for multimedia data

Title of the Course: Pervasive Computing (3CS323)	**L**	**T**	**P**	**Cr**
	3	**-**	**-**	**3**

Pre-Requisite Courses: Network Engineering and Programming, Programming Lab. –II

Textbooks:
1.SengLoke, "Context-Aware Computing Pervasive Systems", Auerbach Pub., New York, 2007
2.Frank Adelstein, Sandeep KS Gupta, "Fundamentals of Mobile and Pervasive Computing", Golden Richard, McGraw-Hill 2005
3.JochenBurkhardt, *"Pervasive Computing: Technology and Architecture of Mobile Internet Applications"*, Addison-Wesley Professional; 3rd edition, 2007
4.John Krumm, "Ubiquitous Computing Fundamentals", CRC Press

References:

1. Stefan Poslad, "Ubiquitous Computing: Smart Devices, Environments and Interactions", Wiley, 2009
2. James Keogh, *"J2ME: The Complete Reference*", Tata McGraw Hill

Course Objectives :

1. Enrich with the new revolutionary ubiquitous computing knowledge.
2. Unleash the prerequisites and concepts enabling Pervasive Computing.
3. Learn protocols, architectures, communication technologies & devices of Pervasive Computing.
4. Conceive pervasive concepts through designing and experimenting in J2ME/MMIT/Android.
5. Acquaint with the successful case studies of potential applications of Pervasive Computing.

Course Learning Outcomes:

CO	**After the completion of the course the learner should be able to,**	**Bloom's Cognitive**	
		level	**Descriptor**
CO1	cognize the concepts of Pervasive Computing.	1,2	Remembering, Understanding
CO2	explore the architecture, various protocols used in enabling Pervasive Computing frameworks.	3,4	Applying, Analyzing
CO3	articulate and appraise hands on experiments of Pervasive Computing.	5	Evaluating
CO4	prepare aptitude to research and implement ubiquity concepts in real life applications of pervasive technologies.	6	Creating

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1		3	2	3							
CO2					2		1				3
CO3			3	1		2		1			
CO4		2	2	3			2				

Assessment:
Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weightage respectively.

Assessment	Marks

ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment, oral, seminar, test (surprise/declared/quiz), and group discussion.[One assessment tool per ISE. The assessment tool used for ISE 1 shall not be used for ISE 2]
MSE: Assessment is based on 50% of course content (Normally first three modules)
ESE: Assessment is based on 100% course content with70-80% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Module 1: Introduction Pervasive Computing Concepts- Evolution, Principles, Characteristics, Transparencies, Context awareness, Architecture for pervasive computing, Pervasive devices embedded controls- smart Sensors and Actuators -Context communication and access services, Issues.	**7 Hrs.**
Module 2: Protocols& Communications Open protocols- Service discovery technologies- SDP, Jini, SLP, UpnP protocols–data Synchronization-SyncML framework - Context aware mobile services - Context aware sensor networks, addressing and communications- Context aware security. Wireless Communications: Infrared vs radio transmission, Infrastructure and ad-hoc network, IEEE802.11 Standard, HiperLAN, Bluetooth, Zigbee.	**7 Hrs.**
Module 3: Enabling Technologies Ubiquitous computing, Context Awareness, Ambient Intelligence, Wearable Computing.	**5 Hrs.**
Module 4: Supporting Architectures Web Application Design Concepts- Frameworks, WAP and Beyond-Voice Technologies, Personal Digital Assistants Server side programming-Pervasive Web application Architecture-scenarios: Application-Access via PCs-Access via WAP-Access via PDA and Voice.	**7 Hrs.**
Module 5: Applications Development Techniques Architecture and Development Environments, Overview of development tools to program wireless devices, Runtime Environments, Midlet Programming, Wireless Toolkit, Tools and Techniques, Test-bed for Pervasive Applications.	**7 Hrs.**
Module 6: Case Scenarios Smart Tokens, Health care systems, Heating ventilation and Air conditioning, Set top boxes, Appliances and Home networking, Residential gateway, Automotive computing, Onboard Computing Systems, In vehicle networks, Entertainment Systems.	**6 Hrs.**

Module wise Measurable Students Learning Outcomes :
After the completion of the course the student should be able to:
Module 1
Introduction

1. Understand the notion of Pervasive Computing
2. Learn the Pervasive Computing Concepts
3. Know the enablers of Pervasive computing.

4. Aware the challenges of Pervasive Computing

Module 2
Protocols& Communications

1. Understand the Protocols used to enable Pervasive Technology
2. Learn various wireless communication's vital role in pervasive computing

Module 3
Enabling Technologies

Drill down concepts of Ubiquitous computing, Context Awareness, Ambient Intelligence, Wearable Computing.

Module 4
Architecture

5. Get acquainted with Web application framework concepts.
6. Groom with Server side programming Pervasive Web application Architecture.
7. Explore application access techniques via WAP, PDA and Voice.
8. Expedite design for meaningful applications using this architecture and framework.

Module 5
Applications Development Techniques

1. Learn programming practices to enable pervasive experiments
2. Learn J2ME Architecture and Development Environment, Midlet Programming using J2ME Wireless Toolkit .
3. Hands on J2ME programming
4. Learn and experiment using Microsoft MMIT/Android

Module 6
Case Scenarios

1. Get exposure to Pervasive Potential applications
2. Obtain confidence in enabling ubiquity to real life engineering applications

Title of the Course: Advanced Database System Lab (3CS371)	L	T	P	Cr
	0	0	2	1

Pre-Requisite Courses:

Textbooks:

1. Database system concepts – Silberschatz, Korth, Sudarshan – 4th Edition (MGH).
2. Database Management System – Raghu Ramkrishnan (MGH)

References:

1.Database Systems : A practical approach to design, implementation &Managemennt by Thomas Connolly & Carolyn Begg (Pearson) Third Edition

2.RamezElmasri and ShamkantNavathe, Fundamentals of Database Systems 2nd Ed, Benjamin Cummings, 1994.

Course Objectives :

1. To practice the concepts/techniques studied in theory course.
2. To have hands-on with different database servers / platforms / tools.
3. To design and implement the database applications.

Course Learning Outcomes:

After the completion of the course the student should be able to		Bloom's Cognitive	
		Level	Descriptor
CO1	compare, evaluate different DBMS for particular application.	4,5	Analyzing, Evaluating
CO2	use different database servers, development platforms / tools.	3	Applying
CO3	analyze, design and implement a database application.	4,6	Analyzing, Creating

CO-PO Mapping :

	a	b	c	d	e	f	g	h	i	j	k
CO1		2									
CO2					3						
CO3			3	2							

Assessment:

In Semester Evaluation (ISE), and End Semester Examination (ESE) having 50% weightageeach.

Assessment	Marks
ISE	50
ESE	50

ISE is based on performance of student in laboratory, experimental write-up, presentation, oral, and test (surprise/declared/quiz). The course teacher shall use at least two assessment tools as mentioned above for ISE.
ESE: Assessment is based on performance and oral.

Course Contents:

- Minimum 10 assignments should be perform to have through understanding and practice of the theory covered in the subject.
- The detail list of assignments will be display by subject teacher.
- Use Oracle 11g Express Edition / IBM DB2 9.7 as database servers
- Use Microsoft IIS as web server
- Use Windows .NET framework 4 / C# as development platform, PL/SQL for database programming

Title of the Course: Internet and Web Programming(3CS372)	L	T	P	Cr
	2	-	2	3

Pre-Requisite Courses: Java Programming, Basics of HTML

Textbook:
1.“A Complete Guide To Internet And Web Programming” Deven N. Shah, Dreamtech Press, 2009

References:
1. “Learning PHP, MySQL, JavaScript, and CSS, 2nd Edition” A Step-by-Step Guide to Creating Dynamic Websites Robin Nixon, O'Reilly Media , August 2012
2. “The XML Handbook” Charls Goldfarb.
3. “Head First” Servlet and JSP - Bryan Basham, SPD O’REILLY, 2ndEdition.

Course Objectives :

1. To introduce students to the major internet programming concepts such as client server programming model, protocols, server configuration, performance, fault tolerance, security etc.
2. To discuss the fundamentals of web programming languages like HTML, client-side scripting language (JavaScript), server-side programming (Servlets, JSP) and XML/web services.
3. To illustrate techniques for developing internet-based applications, mainly focusing on web programming.
4. To demonstrate the MVC architecture and learn the process of web development with the use of best suitable framework.

Course Learning Outcomes:

	After the completion of the course the student should be able to	**Bloom’s Cognitive**	
		level	**Descriptor**
CO1	summarize the concepts of HTML, Servlet, JSP,XML and server configuration. .	1,2	Remembering Understanding
CO2	demonstrate understanding of frameworks, tools, programming languages while constructing web pages	3	Apply
CO3	create and compile advanced dynamic web projects for social and commercial needs using client and server technologies.	**6**	create

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1	2										
CO2					3						
CO3				1		2					3

Assessment:

In Semester Evaluation (ISE), and End Semester Examination (ESE) having 50% weightageeach.

Assessment	Marks
ISE	50
ESE	50

ISE is based on performance of student in laboratory, experimental write-up, presentation, oral, and test (surprise/declared/quiz). The course teacher shall use at least two assessment tools as mentioned above for ISE.
ESE: Assessment is based on performance and oral.

Course Contents:

Module 1: Networks, TCP-IP, Web Organization and Addressing, Ports, protocols, Web Browsers and Web Servers, HTML ,DHTML,Client/Server Architectures, Security,	**4 Hrs.**

Installation of web server etc.	
Module 2: Servlet Basics, Handling the Client Request: Form Data, Handling the Client Request: HTTP Request Headers, Generating the Server Response: HTTP Status Codes, Generating the Server Response: HTTP Response Headers, Handling Cookies, Session Tracking	**5 Hrs.**
Module 3: Understanding the JSP lifecycle, Installing JSP pages, Looking at JSP in the real world, Invoking Java Code with JSP Scripting Elements, Using JavaBeans Components in JSP Documents, Integrating Servlets and JSP: The Model View Controller (MVC) Architecture	**4Hrs.**
Module 4: JSF 2 Overview, Installation, Setup, Configuration, and Getting Started, JSF 2 Programming Basics, Managed Beans I: Using Java Classes to Represent Form Info Managed Beans II: Advanced Features. OR Introduction to struts architecture, action mappings, Hibernate basics, Introduction to spring.	**5 Hrs.**
Module 5: Java script basics, Validation using java script, The basic Ajax process, Using dynamic content and JSP, Using dynamic content and servlets, Sending GET data, Sending POST data, Displaying HTML results, Parsing and displaying XML results.	**4 Hrs.**
Module 6: Introduction, Benefits, components of XML, XML schemas, DTD, XLS, XHTML, WML, CSS, X Link, X pointer, X Include, XBase, XML Technologies & applications viz. E-Commerce, etc.	**4 Hrs.**

Module wise Measurable Students Learning Outcomes :

After the completion of the course the student should be able to:

Module 1

Explain client server architecture, configuration steps of web servers , application servers.

Module 2

Demonstrate lifecycle of Servlet and design web applications with the help of the servlet.

Module 3

Apply JSP for designing client-server programs. As well as able to write a program that interact with database like MS-Access, DB2, Oracle etc.

Module 4

Demonstrate process of web development with different frameworks..

Module 5

Apply usage of java script for form validation.

Module 6

Usage of AJAX and XML , XML parser, DOM tree concepts in web applications.

Title of the Course: Mini-Project-II (3CS342)	L	T	P	Cr
	-	-	2	1

Pre-Requisite Courses: Nil

Textbook: Nil

References: Nil

Course Objectives :

1. To use latest design and development tools
2. To undergo project management techniques and project design principles.
3. To implement the project with appropriate programming languages and testing tools
4. To develop analytical vision and skills to analyze, compare the outcome with other techniques

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**	Bloom's Cognitive	
		level	Descriptor
CO1	demonstrate present technological trends through seminar and presentation	1	Remembering
CO2	demonstrate the appropriate selection of software tool for project implementation.	2	Understanding
CO3	work in teams and participate in group activity of software development.	3	Applying
CO4	develop a software product and demonstrate its significance	4,5	Analyzing, Evaluating

CO-PO Mapping :

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CO1		3									
CO2		3		3							
CO3				2		3	3				
CO4		3									3

Assessment:

In Semester Evaluation (ISE), and End Semester Examination (ESE) having 50% weightageeach.

Assessment	Marks
ISE	50
ESE	50

ISE is based on performance of student in laboratory, experimental write-up, presentation, oral, and test (surprise/declared/quiz). The course teacher shall use at least two assessment tools as mentioned above for ISE.
ESE: Assessment is based on performance and oral.

Course Contents:

1. Students should maintain a project log book containing weekly progress of the project.
2. At the end of the semester project group should achieve all the proposed objectives of the problem statement.
3. The work should be completed in all aspects of design, implementation and testing.
4. Project report should be prepared and submitted in soft and hard form along

with all the code and datasets. 5. Group should demonstrate the work with various test cases and results obtained and explain future scope. 6. The group should participate in technical symposiums, paper presentations to demonstrate their work and findings in technical community.		
Module wise Measurable Students Learning Outcomes :--NIL		