21UCSC100/200	Problem Solving and Programming in C	(3-0-0) 3

Contact Hours: 39 CIE: 50 Marks SEE:100 Marks Exam Duration:3 Hrs.

Course Learning Objectives (CLOs): The course focuses on the following learning results:

- Developing the problem solving skills that can be applied to problems in different areas which enables students to take-up subsequent course work and professional career.
- Provides a comprehensive study of the features of C programming language.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1,12)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	Design a solution by analyzing the given problem scenario and represent it using algorithm / flowchart.	-	1,2,3	-
CO-2	Explain the C language primitives, language principles and use them in writing simple programs.	-	1,2,3	-
CO-3	Write a C program using proper control structures to solve simple problems.	-	1,2,3	-
CO-4	Write a C program using arrays, and strings to solve simple problems.	-	2,6	-
CO-5	Explain the usage of pointers and the need for writing modular programs and demonstrate its use in writing programs.	-	-	1,2,3

POs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Mapping Level	1.7	1.8	1.7	-	-	2.0	-	-	-	-	-	-

Prerequisites: NIL

KURAKUNDI RAGHAVENDRARAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 08:46:06 +05'30'

Contents:

Unit I

Flow-Chart and Algorithm: Solving various scientific, engineering and business related problems of varying complexity.

Fundamentals of C Programming Language: Program structure and execution. Character set, data types, operators, type conversion, expression evaluation. Input and output statements. **08 Hrs**

Unit II

Decision making and Branching: if statement and its different forms, switch statement. **08 Hrs**

Unit III

Decision making and Looping:loops and their behavior – entry and exit controlled loops, conditional and unconditional jump statements, Nested loops. **08 Hrs**

Unit IV

Arrays: Single and multidimensional arrays, advantages and disadvantages of arrays, searching and sorting

Strings: Definition, Different ways of reading and printing strings, string handling functions, applications. **08 Hrs**

Unit V

Modular Programming: Declaration, definition and use of functions, passing parameters to function. **07 Hrs**

Question Paper Pattern:

1. Each question will carry 20 marks with maximum of four sub divisions
2. Each unit will consists of two full questions
3. Students have to answer one full question from each unit and total five questions to be answered.
4. The question paper will have built in choice in the unit.

Reference Books:

1) E Balagurusamy, “Programming in ANSI C”, 6th Edition, Tata McGraw Hill, 2012.
2) Brian W Kernighan & Dennis M Ritchie, “The C programming language”, 2nd Edition, Prentice-Hall India, 2004.
3) R.G. Dromey., “How to solve it by Computer”, Prentice-Hall India, 2008

21UCSL100/200	Computer Programming Lab	(0-0-2) 1

Contact Hours: 26 CIE: 50 Marks SEE:50 Marks Exam Duration:3 Hrs.

Course Learning Objectives (CLOs): The course focuses on the following learning results through practice:

- Conceptualization of the solutions for the given simple problems.
- Representation of the solutions using algorithm and flow chart.
- Writing modular C program to solve simple problems.
- Practicing coding and debugging standards to understand maintainability, testability and other quality parameters.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		Mapping to POs(1,12)		
		Substantial Level (3)	Moderate Level (2)	Slight Level (1)
CO-1	Design a solution by analyzing the given problem scenario and represent it using algorithm / flowchart.	-	1,2,3	-
CO-2	Explain the C language primitives, language principles and use them in writing simple programs.	-	1,2,3	-
CO-3	Write a C program using proper control structures to solve simple problems.	-	1,2,3	-
CO-4	Write a C program using arrays, and strings to solve simple problems.	-	1,2,3	-
CO-5	Explain the usage of pointers and the need for writing modular programs and demonstrate its use in writing programs.	-	-	1,2,3

POs	1	2	3	4	5	6	7	8	9	10	11	12
Mapping Level	1.7	1.7	1.7	-	-	-	-	-	-	-	-	-

Prerequisites: NIL

Working Platform: Linux Operating System

KURAKUNDI RAGHAVENDRARAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 08:46:32 +05'30'

Expected Coding Practices:

1. Use of Good Programming practices: Declaration of variables, Indentation, Documentation, Simplicity of logic, Efficiency of logic, uniformity etc.
2. Generic and Reusable code.
3. Inclusions of exceptional cases.
4. Better usability

21UAEE200	Cyber Law	(2-0-0) 2

Contact Hours: 26 CIE: 50 Marks SEE: 50 Marks Exam Duration: 2 Hrs.

Course Learning Objectives (CLOs): This course will cover the basics of cyber-crimes & spread awareness of this field to help the students understand the importance of security and related laws.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs (1,12)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	Explain the meaning of cyber-crime and its implications. [usage] [BL-3]	-	1, 2, 3, 5	-
CO-2	Identify and Elaborate the taxonomy and classifications of cyber-crimes. **[Familiarity] [BL-2]**	2	1	-
CO-3	Identify the scope and applicability of IT Act-2000; [Familiarity] [BL-2]	-	3	1, 2
CO-4	Explore the Legal Protection against Cyber Crimes. [Familiarity]	2	4, 5, 6	-
CO-5	Study recent trends in cyber law and development. [Familiarity] [BL-3]	2	4, 5, 6	

KURAKUNDI RAGHAVENDRARAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 08:46:41 +05'30'

POs	1	2	3	4	5	6	7	8	9	10	11	12
Mapping Level	1.66	2	2	2	2	2	-	-	-	-	-	-

Prerequisites: NIL

Contents:

Unit I

Introduction:

Cyber Crime: Meaning and Definition, Meaning of Crime, Meaning of Cyber Crime. **04 Hrs.**

Unit II

Classification of Cyber Crimes:

Taxonomy of Cyber Crime, Classifications of Cyber Crimes, Cyber Crimes against Persons, Crimes against Persons' Property, Cybercrimes Against Government, Cybercrimes Against Society at large, Causes of Cyber Crime, Impact and Effects of Cyber Crimes, Cyber Crime: Some Landmark Occurrence. **06 Hrs.**

Unit III

Information Technology Law:

A Bird's Eye View, Cyber World vis-a-vis need of Legal Protection, Information Technology Act, 2000: A Beginning, Objectives of Information Technology Act, 2000, Scope of Information Technology Act, 2000, Applicability of Information Technology Act, 2000, Information Technology Act, 2000: A Snapshot, Information Technology (Amendment) Act, 2008, Recompense of Information Technology Law, Limitation of Information Technology Law. **06 Hrs.**

Unit IV

Legal Protection against Cyber Crimes:

Criminal Liabilities under Information Technology Act, 2000 Common Cyber Crimes and Applicable Legal Provisions: A Snapshot, Civil Liabilities under Information Technology Act, 2000, Civil Liability for Corporate: Cyber Crimes under IPC and Special Laws, The Indian Penal Code, 1860, Cyber Crimes under the Special Acts**.** **06 Hrs.**

Unit V

Cyber Laws – Recent Trends

Different types of cyber law trends and developments of India. **04 Hrs.**

Question Paper Pattern:

1. Each question will carry 10 marks with maximum of two sub divisions.
2. Each unit will consists of two full questions
3. Students have to answer one full question from each unit and total five questions to be answered.
4. The question paper will have built in choice in the unit.

Reference Books:

1) CYBER CRIME LAW AND PRACTICE, Published by: THE INSTITUTE OF COMPANY SECRETARIES OF INDIA, 2016
2) https://cybercrimelawyer.wordpress.com/category/66cpunishment-for-identity theft /
3) www.cyberlawsindia.net
4) http://www.enotes.com/research-starters/social-impactscyber-crime

21UHUC101/201 Society, Environment and Engineering (2-0-0) 2

Contact Hours: 26 CIE: 50 Mark SEE: 50 Marks Exam Duration: 02 Hrs.

Course Learning Objectives (CLOs):

The student is expected to learn the societal structure, development processes, concern towards environment, appropriate technology and role of Engineers in providing engineering solutions for societal comfort.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1,12)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	Analyse the social structure and development needs	6,7		8
CO-2	Create awareness about the need of balanced ecosystems and identify the reasons for environment degradation.	6,7		8
CO-3	Apply mitigation techniques for conservation of environment	6,7		8
CO-4	Evaluate the need and impact of technology on social system and climate	6,7		8

CO-5	Recite his/her role as a facilitator for sustainable development	6,7		8

POs	1	2	3	4	5	6	7	8	9	10	11	12
Mapping Level						3	3	1				

Prerequisites: Nil

Contents:

Unit I

Societal Structures and Dynamics: An analysis of basic sociology concepts and their applications to contemporary society; cultural heritage, occupation mobility and income distribution, social tensions and their causes; societal responsibility and social institutions.

Transformation of industrial society into information society, Development processes: parameters for development of interrelationship between social economic and scientific factors. Role of science and technology in development planning; its objectives and assessment. **06 Hrs.**

Unit II

Ecosystems: Natural ecosystems, Principles of eco-balance, Biosphere cycle, carbon dioxide cycle, causes of eco-imbalance - its effects and remedies.

Environmental Degradation: Causes of degradation– its effects, Control of air, water, soil, and pollutions, Solid waste management, Protection of ozone layer.

05 Hrs.

Unit III

Conservation of environment: Optimum utilization of natural resources, Renew able and non renewable resources, Conflict of resources, Global environmental issues, Climate change as a threat to human civilization and Mitigation measures.

05 Hrs.

Unit IV

Technology: Definition, Impact of technology on environment & society, Benefits of technology due to new inventions, Conflict of technology, technology creation for societal change, Appropriate technology, Intermediate technology, labor based and labor intensive technology, Shifts in employment due to technological advancement, Role of technology to unmask social problems, Impact of technology on culture, tradition and social values. **05 Hrs.**

Unit V

Technology for Sustainable development: Definition and concept, Technology for sustainable energy and materials. Agricultural age, industrial age and information age, Characteristics of information society, Information as power and wealth. Community management, Engineers role as facilitator. **05 Hrs.**

Question Paper Pattern:

1) Each question will carry 10 marks with maximum of two sub divisions
2) Each unit will consists of two full questions
3) Students have to answer one full question from each unit and total five questions to be answered.
4) The question paper will have built in choice in the unit.

Reference Books:

1. B. C. Punmia, Ashok Kumar Jain and Arun Kumar Jain, "Environmental Engineering", 16th Edition, Laxmi Publications (P) Ltd., New Delhi, 2016
2. H.G. Wells, "Brief History of Civilization",
3. J. Neharu, "Glimps of World History", 2004

21UCSM300	Engineering Mathematics – III	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- The basic terminologies of mathematical and logical reasoning, functions, and relations associated with its properties and corresponding practical examples.
- Various counting principle methods to solve complex problems in combinatorics.
- Demonstration with examples, the basic terminologies of graphs and its types.
- Identify the applications of mathematical structures in other fields of computer science such as data structures and algorithms, databases, networks, operating systems etc.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Verify** the correctness of an argument using various techniques and strategies.	1	2	4,15
CO-2	**Solve** problems using counting techniques and combinatorics.	1	2	4,15
CO-3	**Solve** the problems on different	1	2	4,13

	types of functions, relations, and Generating functions.			
CO-4	**Solve** the problems pertaining to graphs and related discrete structures.	1	2	4,13
CO-5	**Explain** the concepts and properties of algebraic structures such as groups and coding theory.	1	2	4,15

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	3.0	2.0	-	1.0	-	-	-	-	-	-	-	-	1.0	-	1.0	-

Pre-requisites: Knowledge of Elementary Mathematics

Contents:

Unit-I

Fundamentals of Logic: Basic Connectives and Truth Tables, Logic Equivalence – The Laws of Logic, Logical Implication – Rules of Inference, The Use of Quantifiers, Quantifiers, Definitions and the Proofs of Theorems **9 Hrs**

Unit-II

Fundamental Counting: The Rules of Sum and Product, Permutations and Combinations, The Binomial coefficients, The Pigeonhole Principle.
Relations: Cartesian Products and Relations, Properties of Relations, Equivalence Relations and Partitions **9 Hrs**

Unit-III

Functions: Definition, Plain and One-to-One, Onto Functions, Function Composition, Inverse Functions, Directed Graphs, Hasse Diagrams. **7 Hrs**

Unit-IV

Generating Functions: Definitions and examples, Exponential Generating Functions.
Recurrence Relations: Recursive definitions. First Order Linear Recurrence Relations, Second order linear homogeneous recurrence relation with constant coefficients. **7 Hrs**

Unit-V

Graphs: Elements of graph theory, Graphs and its properties, Directed graphs, Sub-graph, Complements, Planar graphs, Euler Graph, Hamiltonian Graphs, Graph Colouring, Representation of graphs, Trees. Application to engineering. **7 Hrs**

Reference Books:

1) Ralph P Grimaldi & B.V.Ramana “Discrete and Combinatorial Mathematics”, 5th Edition, Pearson Education, 2006.
2) Kenneth H Rosen, “Discrete Mathematics and its Applications”, 7th Edition, McGraw Hill, 2012.

3) Kolman B & Busby R C, “Discrete and Mathematical Structures for Computer Science”, 5th Edition, Prentice Hall of India 2004.
4) Thomas Kosay, “Discrete Mathematics with Applications”, Elsevier, 2005, Reprint 2008.

21UCSC300	Digital Electronics	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- To introduce the fundamental principles of digital electronics commonly used in Computer Science and Engineering.
- To facilitate them to gain experience with the design of logic devices.
- To provide the student with an understanding of basic digital electronics abstractions on which analysis and design of electronic circuits/systems are based and the capability to model and analyze complex circuits.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Solve** the given Boolean expression to obtain simplified optimal solutions.	2	4	-
CO-2	**Conceptualize** and **solve** the given real time application by employing suitable reduction techniques.	13	3	15
CO-3	**Analyze** the operation of Flip-flops and Shift registers.	3	15	13
CO-4	**Design** sequential circuits for the given problem statement.	3	13	-
CO-5	**Design** a finite state machine by **modeling** different states of a system in a given problem scenario.	13	3	15

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	3.0	2.5	2.0	-	-	-	-	-	-	-	-	2.3	-	1.3	-

Pre-requisites: Knowledge of Basic Electronics.

Contents:

Unit-I

Introduction: Revision of Digital Signals and Waveforms, Digital Logic, logic gates, 7400 TTL Series, Boolean laws, Simplification of Boolean expressions – Minterm and Maxterm representations.

Design of Combinational Logic Circuits: Sum-of-Products simplification, Truth Table to Karnaugh Map, Pairs, Quads, and Octets, K-map Simplifications, Don’t-care

Conditions, Product-of-sums simplification, Simplification by Quine-McClusky Method, 0's & 1's Hazards. **7 Hrs**

Unit-II

Data-Processing Circuits: Multiplexers, De-multiplexers, Decoders, Encoders, Parity Generators and Checkers, Magnitude Comparators. **8 Hrs**

Unit-III

Clocks, Flip-Flops: Clock Waveforms, Clocked D Flip-Flop, Edge-triggered D Flip-Flop, Edge-triggered JK Flip-Flop, Flip-Flop Timing, JK Master-slave Flip-Flop, Switch Contact Bounce Circuits, Various Representation of Flip-Flops.

Registers: Types of Registers, Serial In - Serial Out, Serial In - Parallel out, Parallel In - Serial Out, Parallel In - Parallel Out, Universal Shift Register, Applications of Shift Registers. **8 Hrs**

Unit-IV

Counters: Asynchronous Counters, Decoding Gates, Synchronous Counters, Changing the Counter Modulus, Decade Counters, Pre-settable Counters, Counter Design as a Synthesis problem. **8 Hrs**

Unit-V

Non-linear Applications of Op-amp- Comparator, 555 Timers: working of Astable and Monostable mode

D/A Conversion and A/D Conversion: D/A Converters: Binary Ladder (R:2R),Binary-weighted, A/D Converter-Simultaneous Conversion, A/D Converter-Counter Method, Accuracy and Resolution **8 Hrs**

Reference Books:

1) Donald P Leach, Albert Paul Malvino & Goutam Saha, "Digital Principles and Applications", 7th Edition, Tata McGraw Hill, 2010.
2) R D Sudhaker Samuel, "Illustrative Approach to Logic Design", Sanguine-Pearson, 2010.
3) Charles H. Roth Jr., "Fundamentals of Logic Design", 5th Edition, Cengage Learning, 2004.
4) M Morris Mano, "Digital Logic and Computer Design", 10th Edition, Pearson Education, 2008.

21UCSC301	Data Structures and Applications	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Working of various basic data structures and their implementation.
- Implementation issues of data structure in programming language.

- Selection of the appropriate data structure for solving a given problem.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Write** programs to solve problems using Pointers and Structures/Unions.	-	14	1,3,15,16
CO-2	**Write** programs to solve problems using files.	-	14	1,3,15,16
CO-3	**Write** programs to solve simple problems using stack and **explain** its working principles.	-	14	1,3,15,16
CO-4	**Write** programs to solve problems using queue and **explain** its working principles.	-	14	1,3,15,16
CO-5	**Write** programs to solve problems using Linked Lists and **explain** its working principles.	-	14	1,3,15,16
CO-6	**Write** programs to solve problems using trees and **explain** its working principles.	-	14	1,3,15,16

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.0	-	1.0	-	-	-	-	-	-	-	-	-	-	2.0	1.0	1.0

Pre-requisites: Problem Solving skills and knowledge of Programming in C language.

Contents:

Unit-I

Structures, unions and Pointers: Motivation for using structures. Pointer, access data from memory through pointer, pointer to structures. Motivation for dynamic memory requirement. Realizing arrays using pointer and dynamic memory allocation. Importance of memory management during allocation and de-allocation of memory. **8 Hrs**

Unit-II

Stack: Realization of stack and its operations using static and dynamic structures. Application of stack in converting an expression from infix to postfix and evaluating a postfix expression. Heterogeneous stack using Unions. Applications of Stacks. **8 Hrs**

Unit-III

Queues: Realization of queues (FIFO, Double-ended queue, Priority queue) and its operations using static and dynamic data structures, Applications of Queues. **8 Hrs**

Unit-IV

Lists: Constructing dynamic data structures using self-referential structure (using the same realized linked Lists), operations on lists. Doubly Linked list. Application of Lists in sorting. **8 Hrs**

Unit-V

Trees: Types of trees and their properties, Realization of trees using static and dynamic data structures. Operations on Binary trees and their application in searching (BST and AVL Tree), Binary heap as priority queues, Applications of Trees. **7 Hrs**

Reference Books:

1) Aaron M. Tenenbaum, Yedidyah Langsam & Moshe J. Augenstein, "Data Structures using C and C++", Pearson Education, 2006
2) E. Balagurusamy, "Programming in ANSI C", 7th Edition, Tata McGraw-Hill, 2016
3) Behrouz A. Forouzan & Richard F. Gilberg, "Computer Science: A Structured Programming Approach Using C", 2nd Edition, Cengage Learning, 2003.

21UCSC302 Computer Organization and Architecture (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Basics of sub systems of a computer, their organization, structure and operation.
- Basic concept of programs as sequences of machine instructions.
- Operational aspects of I/O devices and standard I/O interfaces.
- Memory hierarchy and concept of virtual memory.
- Arithmetic and logical operations with integer and floating-point operands.
- Organization of a simple processor, pipelined processor and other computing systems.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the basic working principles of various sub-systems of a computer system.	-	-	1,2,8
CO-2	**Explain** the working principles of different sub systems, such as processor, Input/output, and memory.	-	3,8	1,2,4,6
CO-3	**Design** the required memory bank using basic memory units.	-	3	-

CO-4	**Explain** hardwired control and micro programmed control, pipelining, embedded and other computing systems.	3,4	1	2
CO-5	**Design** simple arithmetic and logical units for a given operational features.	4,8	1,2,3	-

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.5	1.3	2.3	2.3	-	1.0	-	2.0	-	-	-	-	-	-	-	-

Pre-requisites: Knowledge of Digital Electronics and Programming language.

Contents:

Unit-I

Basic Structure and Machine Instructions: Basic Operational Concepts, Bus Structures, Performance - Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement.

Machine Instructions and Programs: Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues, Subroutines, Additional Instructions, Encoding of Machine Instructions. **8 Hrs**

Unit-II

Input / Output Organization: Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Controlling Device Requests, Exceptions, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces – PCIBus, SCSI Bus, USB. **8 Hrs**

Unit-III

Memory System: Basic Concepts, Semiconductor RAM and ROM Memories, Speed, Size and Cost, Cache Memories – Mapping Functions, Replacement Algorithms, Performance Considerations. **8 Hrs**

Unit-IV

Arithmetic: Numbers, Arithmetic Operations and Characters, Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed Operand Multiplication, Fast Multiplication, Integer Division, Floating-point Numbers and Operations. **8 Hrs**

Unit-V

Basic Processing Unit: Some Fundamental Concepts, Execution of a Complete Instruction, Multiple Bus Organization, Hard-wired Control, Micro programmed Control. Pipelining, Embedded Systems and Large Computer Systems: Basic Concepts of pipelining, Examples of Embedded Systems, Processor chips for embedded

applications, Simple Microcontroller. **7 Hrs**

Reference Books:

1) Carl Hamacher, Zvonko Vranesic & SafwatZaky, "Computer Organization", 5th Edition, Tata Mc Graw Hill, 2011.
2) William Stallings, "Computer Organization & Architecture", 9th Edition, Prentice Hall of India, 2012.
3) Vincent P. Heuring & Harry F. Jordan, "Computer Systems Design and Architecture", 2nd Edition, Pearson education, 2004.

21UCSC303 Operating Systems (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): The contents of this course deal with the structure and working principles of generic operating systems at introductory level, focusing on process management, memory management, file system and device management. It also focuses on architecture and programming aspects of Linux based OS at fundamental level.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the working principles of processes and threads and solve problems related to process scheduling.	-	13,14,1,3	-
CO-2	**Solve** Critical Section problem required for process synchronization.	-	1,2,3,14	13
CO-3	**Explain** and **solve** the issues pertaining to deadlock.	-	1,2,3,14	13
CO-4	**Compare** and **contrast** different main memory management strategies.	-	1,2,3,13	15
CO-5	**Explain** the working principles of file system and its management.	-	1,2,13	14
CO-6	**Explain** the structure and working principles of secondary storage management.	-	1,2,13	14

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	2.0	2.0	2.0	-	-	-	-	-	-	-	-	-	1.6	1.6	1.0	-

Pre-requisites: Knowledge of Computer Organization, Digital Electronics and Computer Programming at introductory level.

Contents:

Unit-I

Introduction to Operating Systems: What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Process management; Memory management; Storage management; Protection and security; Distributed system; Special-purpose systems; Computing environments.

Process Management: Process concept; Process scheduling; Operations on processes; Multithreading models; Thread Libraries; Threading issues. Process Scheduling: Basic concepts; Scheduling criteria; Scheduling algorithms; Multiple-Processor scheduling; Thread scheduling. **8 Hrs**

Unit-II

Process Synchronization: Synchronization: The Critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors. **8 Hrs**

Unit-III

Deadlocks: System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock. **8 Hrs**

Unit-IV

Memory Management: Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing. **8 Hrs**

Unit-V

File System, Implementation of File System: File System: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection. Implementing File System: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management

Secondary Storage Structures & Protection: Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management. Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights, Capability-Based systems **7 Hrs**

Reference Books:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 9th Edition, Wiley India, 2018.
2. William Stallings: Operating Systems: Internals and Design Principles, 6th Edition, Prentice Hall, 2013.

3. Tariq Mahmood, Imran Saeed and Tasleem Nustafa: Operating Systems and Networks, 5th Edition, IT Series, 2017

21UAEE324 UNIX Administration and Programming (2-0-0) 2

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- To provide the student with an exposure on UNIX platform so that various domain specific project activities can be performed with ease and comfort.
- To provide the student with an exposure on the structure and working principles of UNIX operating system at introductory level, focusing on OS services, commands, and scripting language for administration of UNIX operating system.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the structure, working principles and basic commands of UNIX operating system.	-	-	13
CO-2	**Use** different UNIX commands and system calls to perform system administration and user specified tasks.	-	14, 15	-
CO-3	**Write** shell scripts to perform different system administrative tasks.	-	5, 13, 14, 15	-
CO-4	**Write** awk scrips to perform different system administrative tasks.	-	5, 13, 14, 15	-
CO-5	**Write** perl scrips to perform different system administrative tasks.	-	5, 13, 14, 15	-

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	-	-	-	-	2.0	-	-	-	-	-	-	-	1.7	2.0	2.0	-

Pre-requisites: Knowledge of basic UNIX commands.

Contents:

Unit-I

Unix System Architecture, commands and System calls: Unix System structure, Commands for performing various activities related to process, files, directories, devices, password protection, vi editors, and other administrative task. System calls related to process, files and directories. **5 Hrs**

Unit-II

Filters in Unix: Paginating Files, head: Displaying the Beginning of a File, tail: Displaying the End of a File, cut: Splitting a File Vertically, paste: Pasting Files, sort Ordering a File, uniq Locate Repeated and Non repeated Lines, tr Translating Characters, An Example: Displaying a Word count List. grep Searching for a Pattern, Basic Regular Expressions (BRE) – An Introduction, Extended Regular Expressions (ERE) and egrep. **6 Hrs**

Unit-III

Shell Programming: Environment Variables, Aliases (bash), Command History (bash). Shell Scripts, read and readonly commands, Using Command Line Arguments, exit and Exit Status of Command, The Logical Operators && and || Conditional Execution, The if Conditional, Using test and [ ] to Evaluate Expressions, The case Conditional, expr: Computation and String Handling, $0: Calling a Script by Different names, while: Looping, for: Looping with a List, set and shift: Manipulating the Positional Parameters. **6 Hrs**

Unit-IV

Awk Scripting Language: awk program line and script structure, awk's operational mechanism, Records and fields, special variables $0, $1, $2, etc., patterns, The BEGIN and END, Variables, built in variables, built in functions, length, split, getline, print, printf, sprintf, index, system, substr, etc., control structures, operators in awk, associative arrays, writing simple awk scripts, Running awk scripts from the shell.

5 Hrs

Unit-V

Perl - The Master Manipulator: Preliminaries, The chop function, Variables and Operators, String handling functions, Specifying filenames in a command line, $_, $. and .., Lists and arrays, argv[]: command line arguments, foreach, split, join, grep, associative arrays. **4 Hrs**

Reference Books:

1. Sumitabha Das UNIX Concepts and Applications, 3rd Edition, Tata McGraw Hill, 2003
2. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 8th edition, Wiley India, 2009.
3. Behrouz A. Forouzan and Richard F.Gilberg UNIX and Shell Programming A Text book, Thomson, edition 2003.

21UHUC300	Universal Human Values - I	(2-0-0) 2

Contact Hours: 26

Course Learning Objectives (CLOs): This course provides an opportunity for the students to enhance their life skills like right understanding leading to the harmonious living in relationship

with the self and family enhancing holistic development of the students.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Recite** and **follow** interpersonal relations with peers and the others	6	-	-
CO-2	**Comprehend** happiness, prosperity and **distinguish** between body and self	-	6, 9	-
CO-3	**Comprehend** harmony and **practice** Sanyam and Svasthya	-	9	-
CO-4	**Demonstrate** the values of human-human interaction and universal values such as *Nyaya*, *Visvasa*, and *Sammana*	7	-	-
CO-5	Clearly **visualize** the co-relation between lack of Human Values and the prevailing problems and **use** tangible steps and a roadmap for moving in the cherished direction.	8	9	-

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	-	-	-	-	-	2.5	3.0	3.0	2.0	-	-	-	-	-	-	-

Pre-requisites: None

Contents:

Unit-I

troduction to Value Education: Understanding the need, basic guidelines, content and process for Value Education, Self-exploration – its content and process; 'Natural Acceptance' and Experiential Validation – as the mechanism for self-exploration , Continuous Happiness and Prosperity – A look at basic human aspirations, Right understanding, Relationship and Physical Facilities – The basic requirements for fulfillment of aspirations of every human being. **6 Hrs**

Unit-II

ıderstanding Happiness and Prosperity: Understanding Happiness and Prosperity correctly – A critical appraisal of the current scenario and Method to fulfill the above human aspirations: Understanding and living in harmony at various levels, Understanding human being as a co-existence of the sentient 'I' and the material 'Body' and the needs of Self ('I') and 'Body' - Sukh and Suvidhã, Understanding the Body as an instrument of 'I' (I being the doer, seer and enjoyer). **5 Hrs**

Unit-III

ɪrmony in the Human Being: Understanding the characteristics and activities of 'I' and

harmony in 'I', Understanding the harmony of 'I' with the Body: Sanyam and Svãsthya; correct appraisal of physical needs, meaning of prosperity in detail, Programs to ensure Sanyam and Svãsthya. **5 Hrs**

Unit-IV

ıarmony in the Family: Understanding harmony in the Family – the basic unit of human interaction, Understanding values in human-human relationship; meaning of Nyãya and program for its fulfillment to ensure Ubhaya –tripti; Trust (Visvãsa) and Respect (Sammãna) as the foundational values of relationship. **5 Hrs**

Unit-V

ıderstanding Intention and Competence: Understanding the meaning of Visvãsa; Difference between intention and competence, Understanding the meaning of Sammãna, Difference between respect and differentiation; the other salient values in relationship. **5 Hrs**

Reference Books:

1. R.R.Gaur, R Asthana, and G.P Bagaria, "A Foundation Course in HUMAN VALUES and professional Ethics", 2nd Revised Edition. EXCEL BOOKS, New Delhi. 2019
2. Videos on UHV by AICTE

21UCSL305	Digital Electronics Laboratory	(0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This laboratory course focuses on the following learning perspectives:

- To gain an intuitive understanding of the role of the electronic components and learn how to employ them to model elementary electronic circuits.
- To take measurements of circuit behavior and compare the performance with predicted circuit models and explain discrepancies.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Design** and **implement** independently combinational circuits for real time scenarios.	1,13	2,3	12
CO-2	**Design** and **implement** independently sequential circuits for real time scenarios.	1,13	2,3	12

CO-3	**Design** and **implement** any application circuit to simulate real time problems.	1,2	15	16

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	3.0	2.3	2.0	-	-	-	-	-	-	-	-	1.0	3.0	-	2.0	1.0

Pre-requisites: Basic Electronics.

List of Experiments

Experiment No. 1 **Logic Gates**

Aim To study and verify the truth table of various logic gates.

Learning Objective
- Identify various ICs for different logic operations.
- Understand ICs nomenclature and their specifications.

Experiment No. 2 **Realization of Boolean Functions**

Aim To simplify the given Boolean expression and to realize it using Basic gates and Universal gates.

Learning Objective
- To understand need for simplifying Boolean expressions.
- To simplify the given Boolean expression and to build the same using logic gates.
- Given a truth table, to derive the Boolean expressions and build the logic circuit to realize it.

Experiment No. 3 **Adders and Subtractors**

Aim To realize Half Adder and Full Adder, Half Subtractor and Full Subtractor using basic gates and NAND gates.

Learning Objective
- To realize the adder and subtractor circuits using basic gates and universal gates.
- To realize full adder using two half adders
- To realize a full subtractor using two half subtractors
- To realize arithmetic operations

Experiment No. 4 **Multiplexers / MEV Technique**

Aim To simplify given Boolean expression using Map Entered Variable(MEV) technique and realize the simplified expression using 8:1 Multiplexers

Learning Objective
- Understand importance of MEV technique.
- Understand multiplexer usage to implement given Boolean expression
- To verify the various functions of IC 74153 and 74151

Experiment No. 5 **Decoders**

Aim To implement given Boolean function using decoders.

Learning Objective
- To learn about working principle of decoders and their usage
- Understand decoder usage to implement given Boolean expression
- To verify the various functions of IC 74138(Decoder)

Experiment No. 6 **Code Converters**

Aim To design and realize the code converters (Binary to Gray code and BCD to Excess – 3 code) using basic gates

Learning Objective
- Understand need for code converters
- To learn to realize various code converters

Experiment No. 7 **Comparators**

Aim To realize Two Bit Comparator using basic gates

Learning Objective
- To learn about various applications of comparator
- To realize logical operations

Experiment No. 8 **Flip-Flops**

Aim To realize the flip-flop conversions (JK to D, JK to T, D to JK, D to T).

Learning Objective
- To learn about various types of Flip-Flops
- Conversions of one type of Flip flop to another

Experiment No. 9 **Applications Flip-Flops**

Aim To design and implement mod-n synchronous counter

Learning Objective
- To learn about applications of flip-flops.
- To learn design procedure for counters.

Experiment No. 10 **Shift Registers / Ring and Johnson counters**

Aim To realize and study Ring Counter and Johnson counter

Learning Objective
- To learn about Ring Counter and its applications
- To learn about Johnson Counter and its application

Experiment No. 11 **Asynchronous counter**

Aim Design and implement a mod-n (n<=9) asynchronous counter using Decade counter IC 7490.

Learning Objective
- To learn about asynchronous counters
- To understand usage of IC 7490

Reference Books:

1. Donald P Leach, Albert Paul Malvino and Goutam Saha: Digital Principles and Applications, 7th Edition, Tata McGraw Hill, 2010.
2. R D Sudhaker Samuel: Illustrative Approach to Logic Design, Sanguine-Pearson, 2010.

3. Charles H. Roth: Fundamentals of Logic Design, Jr., 5th Edition, Cengage Learning, 2004.
4. M Morris Mano: Digital Logic and Computer Design, 10th Edition, Pearson Education, 2008.

21UCSL306 Data Structures and Applications Laboratory (0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This laboratory course focuses on the following learning perspectives:

- Realization of fundamental data structures like stacks, queues, linked lists and trees.
- Compare and contrast the benefits of dynamic and static data structure implementations.
- Selection of the appropriate data structure for solving a given problem.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Write** programs to solve problems using Pointers and Structures/Unions.	-	14	1,3, 15, 16
CO-2	**Write** programs to solve problems using files.	-	14	1,3, 15, 16
CO-3	**Write** programs to solve problems using stack.	-	14	1,3, 15, 16
CO-4	**Write** programs to solve problems using queue.	-	14	1,3, 15, 16
CO-5	**Write** programs to solve problems using Linked Lists.	-	14	1,3, 15, 16
CO-6	**Write** programs to solve problems using trees.	-	14	1,3, 15, 16

POs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.0	-	1.0	-	-	-	-	-	-	-	-	-	-	2.0	1.0	1.0

Suggested list of term works:

The list of experiments is based on the following concepts:

1. Pointers and Structures / Unions.
2. Files
3. Stack
4. Queue
5. Linked Lists
6. Trees

Reference Books:

1. Aaron M. Tenenbaum, Yedidyah Langsam and Moshe J. Augenstein: Data Structures using C and C ++, Pearson Education / PHI, 2006,
2. E. Balagurusamy: Programming in ANSI C, 7th Edition, Tata McGraw-Hill, 2016.
3. Behrouz A. Forouzan and Richard F. Gilberg: Computer Science: A Structured Programming Approach Using C, 2nd Edition, Ceng

18UCSM400	Engineering Mathematics – IV	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- To have an insight into Fourier series, Fourier transforms, Difference equations and Z-transforms.
- Apply discrete and continuous probability distributions in analyzing the probability models arising in engineering field.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Express** periodic function as a Fourier series and **obtain** the various harmonics of the Fourier series expansion for the given numerical data.	-	-	1
CO-2	**Transform** the given function using Fourier transforms depending on the nature of engineering applications. Solve difference equations using Z-transform.	-	-	1
CO-3	Make **Use** of the correlation and regression analysis to fit a suitable mathematical model for the statistical data.	-	1,2	-
CO-4	**Apply** discrete and continuous probability distributions in analyzing the probability models arising in engineering field.	-	1,2	-
CO-5	Estimate the correlation, covariance using joint probability distributions. Also use student's t-distribution, Chi-square distribution as a test of goodness of it.	-	1,2	-

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Mapping Level	3.0	2.0	-	1.0	-	-	-	-	-	-	-	-	1.0	-	1.0	-

Pre-requisites: Knowledge of

1. Differentiation of function.
2. Integration of function.
3. Basic Probability theory.
4. Statistical averages

Course Contents:

Unit I

Fourier Series: Periodic functions, Dirichlet's condition, Fourier series of periodic functions of period 2π and arbitrary period. Half-range Fourier series, Practical harmonic analysis, Examples from engineering field. **8 Hrs**

Unit II

Fourier Transforms: Infinite Fourier transforms, Fourier sine and cosine transforms, Inverse Fourier transforms, Simple problems.

Z-Transforms and Difference equations: Z-transform – definition, Standard Z – transforms, Damping and shifting rules, Initial value and Final value theorems (without proof) with problems, Inverse Z-transform, Simple Problems, Difference equations – Basic definition, Application of Z-transform to solve Difference equation. **8 Hrs**

Unit III

Statistical Methods: Correlation and regression-Karl Pearson's coefficient of correlation and rank correlation-problems. Regression analysis- lines of regression–problems.

Curve Fitting: Curve fitting by the method of least squares- fitting the curves of the form= $\boldsymbol{ax+b; y=ax^2+bx+c; y=ax^b}$. **8 Hrs**

Unit IV

Probability Distributions: Review of basic probability theory, Random variables (discrete and continuous), probability mass/density functions. Binomial, Poisson, exponential and normal distributions- problems (No derivation for mean and standard deviation)-Illustrative examples. **8 Hrs**

Unit V

Joint probability distribution: Joint Probability distribution for two discrete random variables, expectation and covariance.

Sampling Theory: Introduction to sampling distributions, standard error, Type-I and Type-II errors. Test of hypothesis for means, student's t-distribution, Chi-square

distribution as a test of goodness of fit. **7 Hrs**

Reference Books:

1) E. Kreyszig: Advanced Engineering Mathematics, John Wiley & Sons, 10th Ed.(Reprint) 2016.
2) B.S. Grewal: Higher Engineering Mathematics, Khanna Publishers, 44th Ed., 2017.
3) Srimanta Pal et al: Engineering Mathematics, Oxford University Press, 3rd Edition, 2016.

21UCSC400 Programming Computer Peripherals and Interfacing (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Understand the Basic architecture and Peripherals associated with Micro-processers & Micro-controllers.
- Understand the internal architecture, instruction set of ARM7 microcontroller, assembling process & implement small programs.
- Design & develop Assembly Language Program /& C program for a given real time application.
- Demonstrate working knowledge of the necessary steps and methods used to interface ARM7 to devices such as motors, LCD, ADC, and DAC etc.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the Basic architecture and Peripherals associated with Micro-processers & Micro-controllers	-	1	-
CO-2	**Explain** the features of embedded systems, architecture of ARM7 and applications.	-	1	-
CO-3	**Illustrate** the ARM and THUMB instruction sets.	-	2,5	13
CO-4	**Write** an ASM / Embedded C	-	3	-

	program using the instruction set of ARM and THUMB to solve the engineering problems.			
CO-5	**Design** and **Write** ARM (LPC2148) program for specific applications.	-	5	3,12

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.0	1.5	-	2.0	-	-	-	-	-	-	1.0	1.0	-	-	-

Pre-requisites: Number systems, Digital systems and Computer Organization

Course Contents:

Unit-I

Introduction to Processor & Controllers: Evolution of Microprocessor, Block diagram and Features of Microprocessor & Micro-controller, Comparison of Microprocessor and Microcontroller, The RISC & CISC design philosophy. **7 Hrs**

Unit-II

Peripheral Interfacing with Microprocessor: Static and Dynamic memories, Vector interrupt table, Interrupt service routine, Interfacing of microprocessor with Programmable Interrupt Controller 8259, DMA controller 8257, Programmable peripheral Interface-8255. **9 Hrs**

Unit-III

ARM Embedded Systems and ARM Processor Fundamentals: ARM design philosophy, embedded system hardware- AMBA bus protocol, embedded system software- applications. ARM core data flow model, Registers, CPSR-Processor modes, Banked registers. Pipeline- Characteristics **7 Hrs**

Unit-IV

ARM Instruction Set: Fundamentals of ARM instructions, Barrel shifter, Classification and explanation of instructions with examples-Data processing, Branch, Load-store, SWI and Program Status. IRQ and FIQ exceptions, Comparison between exception and interrupts. Interrupt handling schemes- nested interrupt handler, non-nested interrupt handler. Basic interrupt stack design.

Introduction to THUMB and ARM Programming: Introduction to THUMB, Differences between ARM and THUMB, Register usage in Thumb, ARM Thumb Interworking. General Structure of ARM assembly module, Assembler directives.

Simple ALP programs on Arithmetic & logical operations, Factorial, string operation, sorting, searching, and Scan. **8 Hrs**

Unit-V

Peripheral Interfacing: Salient features of LPC2148 ARM CPU, applications, block diagram, memory mapping. Functional features of Interrupt controller, RTC, USB, UART, I2C, SPI, SSP controllers, watch dog timers and other system control units. GPIO, PLL & Timers: Features, Register description with example and Applications. Interfacing of Stepper motor, DC Motor, LED interface. **8 Hrs**

Reference Books:

1. Ramakant A Gayakwad, “Microprocessor and Interfacing ", 4th edition, 2009 Tata McGraw Hill
2. B.Ram, Microprocessors and Interfacing,
3. Atul P Godse and Mrs. Deepali A Godse, Microprocessors and Interfacing,
4. William Hohl, ARM Assembly Language, CRC Press.
5. Steve Furber, ARM System-on-chip Architecture, Pearson Education, 2012
6. LPC 2148 User Manual

21UCSC401 Object Oriented Programming (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Object Oriented (OO) concepts/philosophy and its benefits and drawbacks in system development.
- Basic features of Java programming language to implement Object Oriented (OO) Key concepts like ADT/Encapsulation, reusability (Inheritance/Composite Objects), polymorphism etc., and other core basic features.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Prepare** an abstract data type for the given business scenario and **write** simple programs to represent ADT and **use** in the given application scenario**.**	13	1	-
CO-2	**Write** programs to solve given	2, 14	1, 16	3

	problem using different reusability features like inheritance and composite objects.			
CO-3	**Write** a program to solve given problem using utility classes.	2, 14	1	3
CO-4	**Write** a program to solve given problem using abstract classes and differentiate with interfaces.	2, 14, 16	1	3
CO-5	**Write** a program to solve given problem using packages.	2, 14, 16	1	3
CO-6	**Write** a program to solve given problem using exception handling in construction of robust systems.	2, 14	1	3
CO-7	**Use** multithreading concept to solve conflicts due to interleaved execution of threads and write simple programs.	2, 14	1	3
CO-8	**Use** streams concept in developing system that needs facility for storage and retrieval of data.	2, 14	1	3
CO-9	**Design and Develop** GUI based system using applet, frames, events and other support available in AWT / Swings components.	2, 8, 14	1	3

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	2.0	3.0	1.0	-	-	-	-	3.0	-	-	-	-	3.0	3.0	-	2.3

Pre-requisites: Knowledge of Programming language (any)

Contents:

Unit-I

Introduction to Object Oriented Paradigm (OOP): Object Oriented Philosophy, Key Concepts of OOP, Encapsulation, Polymorphism, Inheritance. **7 Hrs**

Unit-II

Basic Features of Java - 1: Introduction to JAVA, Data Types, Variables and Arrays, String Handling in Java, Control Structures. **8 Hrs**

Unit-III

Basic Features of Java - 2: Classes, Objects, Methods, Constructors, Overloading

methods, Methods and Classes, Inheritance, Packages and Interfaces. **9 Hrs**

Unit-IV

Core Features of Java - 1: Exception Handling, Multi-Threaded Programming, Streams. **8 Hrs**

Unit-V

Core Features of Java - 2: AWT and Swings, Applets, Events **7 Hrs**

Reference Books:

1) Herbert Schildt, "Java-The Complete Reference", 9th Edition, Tata McGraw Hill, 2014.
2) Grady Booch, "Object-Oriented Analysis and Design with Applications", 3rd Edition, Pearson Education, 2007.
3) Raj Buyya et al "Object Oriented Programming with Java", McGraw Hill Publications, 2009
4) Paul Tripti, "Object Oriented Programming using Java", 1st Edition, Cengage Learning India, 2018.

21UCSC402 Analysis and Design of Algorithms (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Analyze the performance of algorithms.
- Demonstrate familiarity with major algorithms and data structures.
- Apply important algorithmic design paradigms and methods of analysis.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the importance of algorithmic/mathematical approach in solving different types of problems.	-	1	-
CO-2	**Analyze** time and space complexity for a given algorithm.	2	-	1
CO-3	**Apply** and **analyze** brute force technique and compare it with other	2	5	3,13

	techniques.			
CO-4	**Apply** and **analyze** divide and conquer technique and compare it with other techniques.	2	5	3,13
CO-5	**Apply** and **analyze** greedy technique and compare it with other techniques.	2	5	3,13
CO-6	**Apply** and **analyze** dynamic programming technique and compare it with other techniques.	2	5	3,13
CO-7	**Apply** and **analyze** backtracking and branch & bound technique and compare it with other techniques.	2	5	3,13

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.5	3.0	1.0	-	2.0	-	-	-	-	-	-	-	1.0	-	-	-

Pre-requisites: Knowledge of Discrete Mathematics and Data Structures

Contents:

Unit-I

Introduction: Algorithm, Fundamentals of problem solving, Problem types, Principles of Algorithm Design. Analysis framework, Asymptotic notations, Mathematical analysis of Non recursive algorithms, Recurrence relations; Mathematical analysis of recursive analysis.

Brute force strategy: Selection Sort, Bubble sort, String matching **7 Hrs**

Unit-II

Divide and Conquer: Introduction and General method, Binary search, Merge sort, Quick sort, Matrix multiplication using Stressen's Matrix multiplication.

Basic Traversal and search techniques: Depth First search, Breadth First Search, Topological Sorting. **8 Hrs**

Unit-III

Dynamic Programming: Introduction and General method, Computing a binomial coefficient, Warshall's algorithm, Floyd's algorithm, Knapsack problem. **8 Hrs**

Unit-IV

Greedy Strategy: Introduction and General Method, Knapsack problem, Job sequencing with dead-lines, min cost spanning tree (Prim's & Kruskal's), Single Source Shortest Path. Huffman Tree. **8 Hrs**

Unit-V

Back tracking and Branch and Bound: Introduction General Method for both strategies Back Tracking: Sum of Sub sets, Knapsack problem, Traveling Sales person (TSP).
Limitations of Algorithm Power: Lower bound arguments, decision trees, P, NP and NP Complete Problems**.** **8 Hrs**

Reference Books:

1. Anany Levitin, "Introduction to the Design and analysis of algorithms", 3rd Edition, Pearson Education, 2011
2. Horowitz, Sahani et.al "Fundamentals of Computer Algorithms", 2nd Edition, Galgotia Publication, 2004.
3. Marks Allen Weiss, "Data Structure and Algorithm Analysis", 3rd Edition, Pearson Education, 2009
4. Thomas H.Cormen, Charles E.Leiserson & Ronald L. Rivest, "Introduction to Algorithms", 2nd Edition, Prentice Hall of India, 2003.

21UCSC403	Software Engineering	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This is a 3 credit, 39 contact hours course at undergraduate level focusing on knowing the process of software system development and enables students to develop software system using engineering techniques.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Describe** the properties of various critical systems and the quality parameters.	-	-	16
CO-2	**Describe** the different key practices of process models.	-	3,13	
CO-3	**Identify** various system requirements and **prepare** system specification reports to solve real life problems in various domains and **develop** domain expertise.	1,2,13	-	-
CO-4	**Conceptualize** the system through **design** and **modeling** the system	1,2,3,13	5	10,12

	architecture, components and processes with quality and standards.			
CO-5	**Develop** software system using engineering techniques, industry relevant tools and programming features/techniques.	1,2,3,13,14	5,15	10,11,12,16
CO-6	**Verify** and **validate** the given system using standard tools and techniques.	-	5,15	10
CO-7	**Manage** project in terms of risk, configuration/versions, Cost and Resources.	-	9,11	10

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	3.0	3.0	2.7	-	2.0	-	-	-	2.0	1.0	1.5	1.0	2.8	3.0	2.0	1.0

Pre-requisites: Knowledge of

- Basics of computer systems and its uses
- Programming language (any)

Contents:

Unit-I

Overview: Introduction: FAQ's about software engineering, Professional and ethical responsibility. Socio-Technical systems: Emergent system properties; Systems engineering; Organizations, people and computer systems; Legacy systems.

Critical Systems, Software Processes: Critical Systems: A simple safety critical system; System dependability; Availability and reliability. Software Processes: Models, Process iteration, Process activities; The Rational Unified Process; Computer Aided Software Engineering. **7 Hrs**

Unit-II

Requirement Engineering: Software Requirements: Functional and Non-functional requirements; User requirements; System requirements; Interface specification; The software requirements document. Requirements Engineering Processes: Feasibility studies; Requirements elicitation and analysis; Requirements validation; Requirements management. **7 Hrs**

Unit-III

System models: System Models: Context models; Behavioral models; Data models; Object models; Structured methods.
Software Design and Development: Architectural Design: Architectural design decisions; System organization; Modular decomposition styles; Control styles. Object-Oriented design. UI Design Issues.
Rapid Software Development: Agile methods; Extreme programming; Rapid application development. Software Evolution: Program evolution dynamics; Software maintenance; Evolution processes; Legacy system evolution. **9 Hrs**

Unit-IV

Verification and Validation: Verification and Validation: Planning; Software inspections; Automated static analysis; Verification and formal methods. Software testing: System testing; Component testing; Test case design; Test automation. Testing Techniques: Equivalence Partitioning, Boundary Value Analysis,, Cause Effect Graphing, Test Generation from Predicates, Statement testing, Branch Testing, Condition Testing, Path Testing, Procedural Call Testing, Data Flow Testing. **9 Hrs**

Unit-V

Software Quality & Project Management: Various Software quality parameters and associated standards and procedures, Project Management activities; Project planning; Project scheduling; Risk management. Configuration Management, Managing People: Selecting staff; Motivating people; Managing people; The People Capability Maturity Model. Software Cost Estimation: Productivity; Estimation techniques, Project duration and staffing. **7 Hrs**

Reference Books:

1. Ian Somerville, “Software Engineering”, 10/E, Person Education, 2016.
2. Roger Pressman, “Software Engineering, Practitioners approach”, 7/E, McGraw-Hill, 2010.
3. Bharat Bhushan Agarwal & Sumit Prakash Tayal, “Software Engineering”, 2nd Edition, Firewal Media Publications, 2007.
4. A.A.Puntambekar, “Software Engineering & Quality Assurance”, 1st Edition, Technical Publications Pune, 2010

21UHUA400 Constitution of India and Professional Ethics (2-0-0) Audit

Contact Hours: 26

Course Learning Objectives (CLOs): The students being citizens of India are to learn about the basic constituents of the constitution of India. Further, they are expected to know about professional ethics and code of practice and also to understand about the society & Engineering and human rights.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		Mapping to POs(1-12) / PSOs (13-16)		
		Substantial Level (3)	Moderate Level (2)	Slight Level (1)
CO-1	**Recite** the Preamble, Salient features of the Constitution, fundamental Rights & limitations.	-	6,7	8
CO-2	**Explain** the directive Principles and the functions of Union executives	-	6,7	8
CO-3	**Explain** the functions of the state executives.	-	6,7	8
CO-4	**Exhibit** the knowledge of Special Provisions and Human Rights.	-	6,7	8
CO-5	**Practice** professional ethics and safe guard the societal comfort.	-	6,7	8

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	-	-	-	-	2.0	2.0	1.0	-	-	-	-	-	-	-	-

Pre-requisites: None

Contents:

Unit-I

troduction to the Constitution of India: The Making of the Constitution and Salient features of the Constitution.

Preamble to the Indian Constitution: Fundamental Rights & its limitations. **5 Hrs**

Unit-II

rective Principles: State Policy & Relevance of Directive Principles State Policy Fundamental Duties.

nion Executives: President, Prime Minister, Parliament and Supreme Court of India. **5 Hrs.**

Unit-III

State Executives: Governor, Chief Minister, State Legislature and High Court of State. Electoral Process in India, Amendment Procedures. **5 Hrs**

Unit-IV

Special Provision for SC & ST, Special Provision for Women, Children & Backward Classes and Emergency Provisions. Powers and functions of Municipalities, Panchyats and Co -Operative Societies.

uman Rights: Meaning and Definitions, Legislation Specific Themes in Human Rights-Working of National Human Rights Commission in India **6 Hrs**

Unit-V

:ope & **Aims**: Engineering Ethics, Responsibility of Engineers, Impediments to Responsibility.

Risks, Safety and liability: Engineers, Honesty, Integrity & Reliability in Engineering.

5 Hrs.

Reference Books:

1. Durga Das Basu: "Introduction to the Constitution on India", (Students Edition), Prentice –Hall EEE, 19th / 20th Edn., 2001
2. Charles E. Haries, Michael S Pritchard and Michael J. Robins "Engineering Ethics" Thompson Asia, 2003.
3. M.V.Pylee, "An Introduction to Constitution of India", Vikas Publishing, 2002.
4. M.Govindarajan, S.Natarajan, V.S.Senthilkumar, "Engineering Ethics", Prentice Hall of India Pvt. Ltd. New Delhi, 2004
5. Brij Kishore Sharma,"Introduction to the Constitution of India", PHI Learning Pvt. Ltd., New Delhi, 2011.
6. Latest Publications of Indian Institute of Human Rights, New Delhi.

21UHUC402 Universal Human Values (2-0-0) 2

Contact Hours: 26

Course Learning Objectives (CLOs): This course provides an opportunity for the students to enhance their life skills like right understanding leading to the harmonious living in relationship with the self and family enhancing holistic development of the students.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Recite** and **follow** interpersonal relations with peers and the others	6	-	-
CO-2	**Demonstrate** the concept of harmony in nature and need of self-regulation.	-	6, 9	-
CO-3	**Recite** and **follow** natural acceptance and differentiate between intention and competence.	-	9	-
CO-4	**Differentiate** between the	7	-	-

	characharesticis and activities of different orders existing in nature and demonstrtate the role of human beings in mutual fulfilment with all the orders of Nature.			
CO-5	**Visualize** and involve in the strategic preparation for Universal Human Order.	8	9	-

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	-	-	-	-	2.5	3.0	3.0	2.0	-	-	-	-	-	-	-

Pre-requisites: None

Contents:

Unit-I

Harmony in the Society: Understanding the harmony in the society (society being an extension of family): Samãdhãna, Samriddhi, Abhaya, Sah-astitva as comprehensive Human Goals. **Visualizing a universal harmonious order in society** – Undivided Society (AkhandSamãj), Universal Order (SãrvabhaumaVyavasthã) - from family to world family. **5 Hrs**

Unit-II

Harmony in the Nature (Existence): Understanding the harmony in the Nature, Interconnectedness and mutual fulfillment among the four orders of nature – recyclability and self-regulation in nature, Understanding existence as co-existence (Sah-astitva) of mutually interacting units in all-pervasive space, Holistic perception of harmony at all levels of existence. **6 Hrs**

Unit-III

Implications of the Holistic Understanding: Natural acceptance of human values, Definitiveness of Ethical Human Conduct, Basis for Humanistic Education, Humanistic Constitution and Universal Human Order. **5 Hrs**

Unit-IV

A Look at Professional Ethics: Competence in Professional Ethics:

(a) Ability to utilize the professional competence for augmenting universal human order

(b) Ability to identify the scope and characteristics of people-friendly and eco-friendly production systems, technologies and management models **5 Hrs**

Unit-V

Strategy for transition towards Universal Human Order:

- Case studies of typical holistic technologies, management models and production systems
- Strategy for transition from the present state to Universal Human Order:
    a. At the level of individual: as socially and ecologically responsible engineers, technologists and managers
    b. At the level of society: as mutually enriching institutions and organizations. **5 Hrs**

Reference Books:

1. R.R.Gaur, R Asthana, and G.P Bagaria, "A Foundation Course in HUMAN VALUES and professional Ethics", 2nd Revised Edition. EXCEL BOOKS, New Delhi. 2019
2. Videos on UHV by AICTE

21UCSL404 Object Oriented Programming Laboratory (0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This laboratory course focuses on the following learning perspectives:

- Object Oriented (OO) concepts/philosophy and its benefits and drawbacks in system development.
- Basic features of Java programming language to implement Object Oriented (OO) Key concepts like ADT/Encapsulation, reusability (Inheritance/Composite Objects), polymorphism etc., and other core basic features.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Prepare** an abstract data type for the given business scenario and **write** simple programs to represent ADT and **use** in the given application scenario**.**	13	1	-
CO-2	**Write** programs to solve given problem using different reusability features like inheritance and	2, 14	1, 16	3

	composite objects.			
CO-3	**Write** a program to solve given problem using utility classes.	2, 14	1	3
CO-4	**Write** a program to solve given problem using abstract classes and differentiate with interfaces.	2, 14, 16	1	3
CO-5	**Write** a program to solve given problem using packages.	2, 14, 16	1	3
CO-6	**Write** a program to solve given problem using exception handling in construction of robust systems.	2, 14	1	3
CO-7	**Use** multithreading concept to solve conflicts due to interleaved execution of threads and write simple programs.	2, 14	1	3
CO-8	**Use** streams concept in developing system that needs facility for storage and retrieval of data.	2, 14	1	3
CO-9	**Design and Develop** GUI based system using applet, frames, events and other support available in AWT / Swings components.	2, 8, 14	1	3

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	3.0	1.0	-	-	-	-	3.0	-	-	-	-	3.0	3.0	-	2.3

Pre-requisites: Knowledge of: Registration/Completion of the course Object Oriented Programming.

Suggested Platforms:
Notepad (Non IDE), IDE (JCreator, Net Beans, Eclipse etc) in Windows OS and Linux OS

All programs should:

1. Be written to realize the Object Oriented Philosophy and core Java features.
2. Be written with Java Naming & Coding conventions and well documented.
3. Handle exceptions.
4. Be tested for all possible scenarios.

Course Contents:
Minimum one exercise to cover each course outcome specified above. Minimum

8 experiments to be completed by each student independently covering all course outcomes defined for this course. Course teacher has to publish list of experiments along with individual outcome for every experiments, on the first day of the semester. Examiner may set any problem based on the published term work during tests.

Reference Books:

1) Herbert Schildt, "Java: The Complete Reference: 7th Edition, Tata McGraw Hill, 2007.
2) Kathy Sierra & Bert Bates, "Head First Java", 2nd Edition, O'Reilly, 2009
3) Patrick Niemeyer & Daniel Leuck, "Learning Java", 4th Edition, O'Reilly, 2013
4) Laura Lemay & Charles L. Perkins, "Teach Yourself Java in 21 Days", 7th Edition, Sams Publishing, 2016

21UCSL405 Programming Computer Peripherals and Interfacing Laboratory (0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This laboratory course focuses on the following learning perspectives:

- Understand the internal architecture, instruction set of ARM7 microcontroller, assembling process & implement small programs.
- Design & develop Assembly Language Program /& C program for a given real time application.
- Understand the use of interrupts & other advanced concepts related to ARM7
- Demonstrate working knowledge of the necessary steps and methods used to interface ARM7 to devices such as motors, LCD, ADC, and DAC etc.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Execute** assembly level codes for a given specific problem using ARM processor.	-	2, 4	3,15
CO-2	**Execute** embedded C programs for a given specific problem using ARM processor.	-	4,14	15,16

CO-3	**Implement** programs for interfacing with real world devices such as LCD's Keyboards, DAC, ADC, Relays Motors and Serial Interface - RTC, USB, UART, I2C.	13	4,5,16	3,12

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	2.0	1.0	2.0	2.0	-	-	-	-	-	-	1.0	3.0	2.0	1.0	1.5

Course Contents:

PART A: Conduct the following experiments to learn ALP using ARM:

- Arithmetic and logical operations
- Interrupts related operations
- Timer related applications.

PART B: Conduct interfacing experiments to learn embedded C for ARM:

- LCD- interfacing
- Stepper Motor Interfacing
- Real time sensors Interfacing
- 7-segment LED interface
- Serial Interface (USB, UART, I2C)
- Timer /Counters
- Interrupt Controller

Reference Books:

1. William Hohl, ARM Assembly Language, CRC Press.
2. Steve Furber, ARM System-on-chip Architecture by, Pearson Education, 2012
3. James K. Peckol, Embedded Systems: A Contemporary Design Tool, 2008
4. Jonathan W. Valvano, Brookes / Cole, Embedded Microcomputer Systems, Real Time Interfacing, 1999
5. LPC 2148 User Manual.

21UCSL406 Introductory Project (0-0-2) 1

Contact Hours: 24

Course Learning Objectives (CLOs): This course enables the student to identify the community expectations in terms of possible engineering solutions and prepare project proposal.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Identify** the societal problems	-	-	2,6,7,9,12,13
CO-2	**Analyze** real environment and **Formulate** the problem statement.	-	-	9,12,13
CO-3	**Conduct** exhaustive literature survey	-	-	9,12,13
CO-4	**Propose** sustainable engineering solutions	-	-	5,7,12,13
CO-5	**Prepare** the report and communicate effectively through presentation.	-	-	9,10,12,13
CO-6	**Manage** the project in terms of various resources in a particular discipline or in a multi-disciplinary domain.	-	-	11

POs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	-	1.0	-	-	1.0	1.0	1.0	-	1.0	1.0	1.0	1.0	1.0	-	-	-

Guidelines for conduction:

1. Team consists of 4-5 students.
2. Students have to choose a guide among the faculty members who are teaching their semester.
3. In consultation with guide, the team should carry out their project work.
4. Final evaluation is based on seminar and report submission.
5. This requires designated committee to monitor the process of conduction

KURAKUNDI RAGHAVENDRA RAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 09:49:16 +05'30'

18UCSC100/200 Problem Solving and Programming in C (3-2-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This is a 4 credit course, to be covered in 52 hours for UG program at introductory level for all branches of Engineering. It focuses on the following learning results:

- Developing the problem solving skills that can be applied to problems in different areas which enables students to take-up subsequent course work and professional career.
- Provides a comprehensive study of the features of C programming language.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Design** a solution by **analyzing** the given problem scenario and **represent** it using algorithm / flowchart.	-	1,2,3	-
CO-2	**Explain** the C language primitives, language principles and use them in writing simple programs.	-	1,2,3	-
CO-3	**Write** a C program using proper control structures to solve simple problems.	-	1,2,3	-
CO-4	**Write** a C program using arrays, and strings to solve simple problems.	-	1,2,3	-
CO-5	**Explain** the usage of pointers and the need for writing modular programs and **demonstrate** its use in writing programs.	-	-	1,2,3
CO-6	**Write** a C program using structures and unions, to solve simple problems.	-	-	1,2,3
CO-7	**Write** C program using file processing features and libraries to solve simple problems.	-	-	1,2,3

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.6	1.6	1.6	-	-	-	-	-	-	-	-	-	-	-	-	-

KURAKUNDI RAGHAVENDRARAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 09:49:40 +05'30'

Pre-requisites: None

Contents:

Unit-I

Flow-Chart and Algorithm: Solving various scientific, engineering and business related problems of varying complexity.

Fundamentals of C Programming Language: Program structure and execution. Character set, data types, operators, type conversion, expression evaluation. Input and output statements. **10L + 2T Hrs**

Unit-II

Decision making and Branching: if statement and its different forms, switch statement. **6L + 2T Hrs**

Unit-III

Decision making and Looping: loops and their behavior – entry and exit controlled loops, conditional and unconditional jump statements, Nested loops. **8L + 2T Hrs**

Unit-IV

Arrays: Single and multidimensional arrays, advantages and disadvantages of arrays, searching and sorting.

Strings: Definition, Different ways of reading and printing strings, string handling functions, applications. **8L + 2T Hrs**

Unit-V

Modular Programming: Declaration, definition and use of functions, passing parameters to function.

Building Blocks of Data Structure: structures, unions, pointer and file operations. **10L + 2T Hrs**

Reference Books:

1) E Balagurusamy, “Programming in ANSI C”, 6th Edition, Tata McGraw Hill, 2012.
2) Brian W Kernighan & Dennis M Ritchie, “The C programming language”, 2nd Edition, Prentice-Hall India, 2004.
3) R.G.Dromey., “How to solve it by Computer”, Prentice-Hall India, 2008
4) B A Forouzan and R F Gilberg, “Computer Program: A structured programming approach using C”, 3rd Edition, Thomson Learning, 2005
5) Brain W. Kernighan and Rob Pike, “The Practice of Programming”, Pearson Education Inc. 2008.

18UCSL100/200 Problem Solving and Programming in C Lab (0-0-2) 1
Contact Hours: 24

Course Learning Objectives (CLOs): This is a 1 credit laboratory course, to be covered in 36 hours for UG program at introductory level for all branches of Engineering. It focuses on the following learning results through practice:

- Conceptualization of the solutions for the given simple problems.
- Representation of the solutions using algorithm and flow chart.
- Writing modular C program to solve simple problems.
- Practicing coding and debugging standards to understand maintainability, testability and other quality parameters.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Design** a solution by **analyzing** the given problem scenario and **represent** it using algorithm / flowchart.	-	1,2,3	-
CO-2	**Explain** the C language primitives, language principles and use them in writing simple programs.	-	1,2,3	-
CO-3	**Write** a C program using proper control structures to solve simple problems.	-	1,2,3	-
CO-4	**Write** a C program using arrays, and strings to solve simple problems.	-	1,2,3	-
CO-5	**Explain** the usage of pointers and the need for writing modular programs and **demonstrate** its use in writing programs.	-	-	1,2,3
CO-6	**Write** a C program using structures and unions, to solve simple problems.	-	-	1,2,3
CO-7	**Write** C program using file processing features and libraries to solve simple problems.	-	-	1,2,3

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.6	1.6	1.6	-	-	-	-	-	-	-	-	-	-	-	-	-

KURAKUNDI RAGHAVENDRARAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 09:50:31 +05'30'

Pre-requisites: None

Working Platform: Linux Operating System

Expected Coding Practices:

1. Use of Good Programming practices: Declaration of variables, Indentation, Documentation, Simplicity of logic, Efficiency of logic, uniformity etc.
2. Generic and Reusable code.
3. Inclusions of exceptional cases.
4. Better usability

Course Contents:

Programming exrcises of varying complexity, to meet the learning results stated in course outcomes for this course.

Reference Books:

1) E Balagurusamy, “Programming in ANSI C”, 6th Edition, Tata McGraw Hill, 2012.
2) Brian W Kernighan & Dennis M Ritchie, “The C programming language”, 2nd Edition, Prentice-Hall India, 2004.
3) R.G.Dromey., “How to solve it by Computer”, Prentice-Hall India, 2008
4) B A Forouzan and R F Gilberg, “Computer Program: A structured programming approach using C”, 3rd Edition, Thomson Learning, 2005
5) Brain W. Kernighan and Rob Pike, “The Practice of Programming”, Pearson Education Inc. 2008.

III Semester

18UMAC300	Engineering Mathematics – III	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- To have an insight into Laplace transforms, Fourier series, Fourier Transforms, Difference equations and Z-transforms.
- To develop the proficiency in variational calculus.
- To solve ODE's arising in engineering applications, using numerical methods.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Transform** the given function using Laplace / Fourier transforms depending on the nature of engineering applications.	-	-	1
CO-2	**Express** periodic function as a Fourier series and **obtain** the various harmonics of the Fourier series expansion for the given numerical data.			1,2
CO-3	**Solve** difference equations using Z-transform.			1
CO-4	**Solve** first and second order differential equations arising in engineering problems using single step and multistep numerical methods.		1,2	
CO-5	**Determine** the extremals of functional using calculus of variations and **solve** problems arising in engineering.			1,2

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.2	1.3	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Pre-requisites: Knowledge of Differentiation and Integration of Functions.

KURAKUNDI RAGHAVENDRARAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 09:51:43 +05'30'

Contents:

Unit-I

Laplace Transforms: Definition and Properties, Laplace transforms of elementary functions, Laplace transforms of Periodic functions and unit-step function - Problems
Inverse Laplace Transforms: Inverse Laplace Transform – Problems, Convolution Theorem (without proof) to find the inverse laplace transform and problems, Solution of linear differential equations using Laplace Transform **8Hrs**

Unit-II

Fourier Series: Periodic functions, Dirichlet's condition, Fourier series of periodic functions of period 2π and arbitrary period. Half-range Fourier series, Practical harmonic analysis, Examples from engineering field. **8 Hrs**

Unit-III

Fourier Transforms: Infinite Fourier transforms, Fourier sine and cosine transforms, Inverse Fourier transforms, Simple problems.
Z-Transforms and Difference equations: Z-transform – definition, Standard Z – transforms, Damping and shifting rules, Initial value and Final value theorems (without proof) with problems, Inverse Z-transform, Simple Problems, Difference equations – Basic definition, Application of Z-transform to solve Difference equation. **8 Hrs**

Unit-IV

Numerical Solutions of Ordinary Differential Equations (ODE's): Numerical solution of ODE's of first order and first degree - Taylor's series method, Modified Euler's method, Runge – Kutta method of fourth order, Milne's predictor and corrector method (Noderivations of formulae), Problems. **7 Hrs**

Unit-V

Numerical Solution of Second Order ODE's: Runge-Kutta method and Milne's predictor and Corrector method (No derivations of formulae).
Calculus of Variations: Variation of function and functional, Variational problems, Euler's equation (without proof), Geodesics (plane), Hanging chain problems. **8 Hrs**

Reference Books:

1) B.S.Grewal, "Higher Engineering Mathematics", 44th Edition, Khanna Publishers, 2017.
2) E.Kreyszig, "Advanced Engineering Mathematics", 10th Edition (Reprint), John Wiley & Sons, 2016.
3) Srimanta Pal et al, "Engineering Mathematics", 3rd Edition, Oxford University Press, 2016.

4) C.Ray Wylie & Louis C Barrett, “Advanced Engineering Mathematics”, 6th Edition, McGraw-Hill Book Co, 1995.
5) S.S.Sastry, “Introductory Methods of Numerical Analysis”, 4th Edition, Prentice Hall of India, 2010.
6) B.V.Ramana, “Higher Engineering Mathematics”, 11th Edition, Tata McGraw-Hill, 2010.
7) N.P.Bali & Manish Goyal, “A Text Book of Engineering Mathematics”, 7th Edition, Laxmi Publishers, 2014.
8) Veeraranjan T, “Engineering Mathematics for First Year”, Tata McGraw-Hill, 2008.
9) Thomas G.B. & Finney R.L., “Calculus and Analytical Geometry”, 9th Edition, Pearson, 2012.

18UCSC300 Digital Electronics (4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- To introduce the fundamental principles of digital electronics commonly used in Computer Science and Engineering.
- To facilitate them to gain experience with the design of logic devices.
- To provide the student with an understanding of basic digital electronics abstractions on which analysis and design of electronic circuits/systems are based and the capability to model and analyze complex circuits.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Solve** the given boolean expression to obtain simplified optimal solutions.	2	4	-
CO-2	**Conceptualize** and **solve** the given real time application by employing suitable reduction techniques.	13	3	15
CO-3	**Design** combinational subsystems to perform different logical and arithmetic operations.	3	15	13
CO-4	**Design** sequential circuits for the	3	13	-

	given problem statement.			
CO-5	**Design** a finite state machine by **modeling** different states of a system in a given problem scenario.	13	3	15
CO-6	**Use** MSI chips to **build system** for the given real time application.	5	4	14

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	3.0	2.5	2.0	3.0	-	-	-	-	-	-	-	2.3	1.0	1.3	-

Pre-requisites: Knowledge of Basic Electronics.

Contents:

Unit-I

Introduction: Revision of logic gates. Definitions for Digital Signals, Digital Waveforms, Digital Logic, 7400 TTL Series Working of Op-amp, Non-linear applications – Comparator, Schmitt Trigger.

Combinational Logic Circuits: Sum-of-Products Method, Truth Table to Karnaugh Map, Pairs, Quads, and Octets, Karnaugh Simplifications, Don't-care Conditions, Product-of-sums Method, Product-of-sums simplifications, Simplification by Quine-McClusky Method, Hazards. **9 Hrs**

Unit-II

Data-Processing Circuits: Multiplexers, De-multiplexers, 1-to-16 Decoder, Encoders, Exclusive-OR Gates, Parity Generators and Checkers, Magnitude Comparator, Programmable Array Logic, Programmable Logic Arrays.

Clocks, Flip-Flops: Clock Waveforms, TTL Clock, Clocked D FLIP-FLOP, Edge-triggered D FLIP-FLOP, Edge-triggered JK FLIP-FLOP, FLIP-FLOP Timing, JK Master-slave FLIP-FLOP, Switch Contact Bounce Circuits, Various Representation of FLIP-FLOPs, Analysis of Sequential Circuits. **11 Hrs**

Unit-III

Registers: Types of Registers, Serial In - Serial Out, Serial In - Parallel out, Parallel In - Serial Out, Parallel In - Parallel Out, Universal Shift Register, Applications of Shift Registers.

Counters: Asynchronous Counters, Decoding Gates, Synchronous Counters, Changing the Counter Modulus, Decade Counters, Pre-settable Counters, Counter Design as a Synthesis problem. **11 Hrs**

Unit-IV

Design of Synchronous Sequential Circuits: Synchronous Sequential Circuit: Model Selection, State Transition Diagram, State Synthesis Table, Design Equations and

Circuit Diagram, Algorithmic State Machine, State Reduction Technique. **10 Hrs**

Unit-V

Design of Asynchronous Sequential Circuits: Analysis of Asynchronous Sequential Circuit, Problems with Asynchronous Sequential Circuits, Design of Asynchronous Sequential Circuit.

D/A Conversion and A/D Conversion: Variable, Resistor Networks, Binary Ladders, D/A Converters, D/A Accuracy and Resolution, A/D Converter-Simultaneous Conversion, A/D Converter-Counter Method, Continuous A/D Conversion, A/D Techniques, Dual-slope A/D Conversion, A/D Accuracy and Resolution. **11 Hrs**

Reference Books:

6) Donald P Leach, Albert Paul Malvino & Goutam Saha, "Digital Principles and Applications", 7th Edition, Tata McGraw Hill, 2010.
7) R D Sudhaker Samuel, "Illustrative Approach to Logic Design", Sanguine-Pearson, 2010.
8) Charles H. Roth Jr., "Fundamentals of Logic Design", 5th Edition, Cengage Learning, 2004.
9) M Morris Mano, "Digital Logic and Computer Design", 10th Edition, Pearson Education, 2008.

18UCSC301 Discrete Structures in Computer Science (3-2-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- The basic terminologies of mathematical and logical reasoning, functions, and relations associated with its properties and corresponding practical examples.
- Various counting principle methods to solve complex problems in combinatorics.
- Demonstration with examples, the basic terminologies of graphs and its types.
- Identify the applications of mathematical structures in other fields of computer science such as data structures and algorithms, databases, networks, operating systems etc.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		Mapping to POs(1-12) / PSOs (13-16)		
		Substantial Level (3)	Moderate Level (2)	Slight Level (1)
CO-1	**Verify** the correctness of an argument using various techniques and strategies.	1	2	4,15
CO-2	**Solve** problems using counting techniques and combinatorics.	1	2	4,15
CO-3	**Solve** the problems on different types of functions, relations, and Generating functions.	1	2	4,13
CO-4	**Solve** the problems pertaining to graphs and related discrete structures.	1	2	4,13
CO-5	**Explain** the concepts and properties of algebraic structures such as groups and coding theory.	1	2	4,15

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	3.0	2.0	-	1.0	-	-	-	-	-	-	-	-	1.0	-	1.0	-

Pre-requisites: None

Contents:

Unit-I

Fundamentals of Logic: Basic Connectives and Truth Tables, Logic Equivalence – The Laws of Logic, Logical Implication – Rules of Inference, The Use of Quantifiers, Quantifiers, Definitions and the Proofs of Theorems **9L +3T Hrs**

Unit-II

Fundamental Counting: The Rules of Sum and Product, Permutations and Combinations, The Binomial coefficients, The Pigeon-hole Principle.

Relations: Cartesian Products and Relations, Properties of Relations, Equivalence Relations and Partitions **8L+3T Hrs**

Unit-III

Functions: Definition, Plain and One-to-One, Onto Functions, Function Composition, Inverse Functions, Directed Graphs, Hasse Diagrams. **7L+3T Hrs**

Unit-IV

Generating Functions: Definitions and examples, Exponential Gnerating Functions.

KURAKUNDI RAGHAVENDRARAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 09:54:06 +05'30'

Recurrence Relations: Recursive definitions. First Order Linear Recurrence Relations, Second order linear homogenious recurrence retelation with constant coefficients. **7L+3T Hrs**

Unit-V

Graphs: Elements of graph theory, Graphs and its properties, Directed graphs, Subgrapgh, Complements, Planar graphs, Euler Graph, Hamiltonian Graphs, Graph Coloring, Representation of graphs, Trees.

Groups: Definitions, Examples, and Elementary Properties, Homomorphisms, Isomorphisms **7L+2T Hrs**

Reference Books:

5) Ralph P Grimaldi & B.V.Ramana "Discrete and Combinatorial Mathematics", 5th Edition, Pearson Education, 2006.
6) Kenneth H Rosen, "Discrete Mathematics and its Applications", 7th Edition, McGraw Hill, 2012.
7) Kolman B & Busby R C, "Discrete and Mathematical Structures for Computer Science", 5th Edition, Prentice Hall of India 2004.
8) Thomas Kosay, "Discrete Mathematics with Applications", Elsevier, 2005, Reprint 2008.

18UCSC302 Data Structures and Applications (4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Working of various basic data structures and their implementation.
- Implementation issues of data structure in programming language.
- Selection of the appropriate data structure for solving a given problem.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Write** programs to solve problems using Pointers and Structures/Unions.	-	14	1,3,15,16
CO-2	**Write** programs to solve problems using files.	-	14	1,3,15,16
CO-3	**Write** programs to solve simple	-	14	1,3,15,16

	problems using stack and **explain** its working principles.			
CO-4	**Write** programs to solve problems using queue and **explain** its working principles.	-	14	1,3,15,16
CO-5	**Write** programs to solve problems using Linked Lists and **explain** its working principles.	-	14	1,3,15,16
CO-6	**Write** programs to solve problems using trees and **explain** its working principles.	-	14	1,3,15,16
CO-7	**Write** programs to solve problems using Hashing and **explain** its working principles.	-	14	1,3,15,16

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.0	-	1.0	-	-	-	-	-	-	-	-	-	-	2.0	1.0	1.0

Pre-requisites: Problem Solving skills and knowledge of Programming in C language.

Contents:

Unit-I

Structures, unions and Pointers: Motivation for using structures. Pointer, access data from memory through pointer, pointer to structures. Motivation for dynamic memory requirement. Realizing arrays using pointer and dynamic memory allocation. Importance of memory management during allocation and de-allocation of memory. **10 Hrs**

Unit-II

Stack: Realization of stack and its operations using static and dynamic structures. Application of stack in converting an expression from infix to postfix and evaluating a postfix expression. Heterogeneous stack using Unions. Applications of Stacks. **10 Hrs**

Unit-III

Queues: Realization of queues (FIFO, Double-ended queue, Priority queue) and its operations using static and dynamic data structures, Applications of Queues. **10 Hrs**

Unit-IV

Lists: Constructing dynamic data structures using self-referential structure (using the same realized linked Lists), operations on lists. Doubly Linked list. Application of Lists in sorting. **12 Hrs**

KURAKUNDI RAGHAVENDRA RAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 09:55:07 +05'30'

Unit-V

Trees: Types of trees and their properties, Realization of trees using static and dynamic data structures. Operations on Binary trees and their application in searching (BST and AVL Tree), Binary heap as priority queues, Applications of Trees.
Hash Table: Realizing effective hash table with proper data stricture and hash function, its application. **10 Hrs**

Reference Books:

4) Aaron M. Tenenbaum, Yedidyah Langsam & Moshe J. Augenstein, "Data Structures using C and C++", Pearson Education, 2006
5) E. Balagurusamy, "Programming in ANSI C", 7th Edition, Tata McGraw-Hill, 2016
6) Behrouz A. Forouzan & Richard F. Gilberg, "Computer Science: A Structured Programming Approach Using C", 2nd Edition, Cengage Learning, 2003.

18UCSC303 Computer Organization and Architecture (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Basics of sub systems of a computer, their organization, structure and operation.
- Basic concept of programs as sequences of machine instructions.
- Operational aspects of I/O devices and standard I/O interfaces.
- Memory hierarchy and concept of virtual memory.
- Arithmetic and logical operations with integer and floating-point operands.
- Organization of a simple processor, pipelined processor and other computing systems.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the basic working principles of various sub-systems of a computer system.	-	-	1,2,8
CO-2	**Explain** the working principles of different sub systems, such as	-	3,8	1,2,4,6

KURAKUNDI RAGHAVENDRA RAO GOPINATH
Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 09:56:33 +05'30'

	processor, Input/output, and memory.			
CO-3	**Design** the required memory bank using basic memory units.	-	3	-
CO-4	**Explain** hardwired control and micro programmed control, pipelining, embedded and other computing systems.	3,4	1	2
CO-5	**Design** simple arithmetic and logical units for a given operational features.	4,8	1,2,3	-

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.5	1.3	2.3	2.3	-	1.0	-	2.0	-	-	-	-	-	-	-	-

Pre-requisites: Knowledge of Digital Electronics and Programming language.

Contents:

Unit-I

Basic Structure and Machine Instructions: Basic Operational Concepts, Bus Structures, Performance - Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement.

Machine Instructions and Programs: Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues, Subroutines, Additional Instructions, Encoding of Machine Instructions. **8 Hrs**

Unit-II

Input / Output Organization: Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Controlling Device Requests, Exceptions, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces – PCIBus, SCSI Bus, USB. **8 Hrs**

Unit-III

Memory System: Basic Concepts, Semiconductor RAM and ROM Memories, Speed, Size and Cost, Cache Memories – Mapping Functions, Replacement Algorithms, Performance Considerations. **8 Hrs**

Unit-IV

Arithmetic: Numbers, Arithmetic Operations and Characters, Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed

Operand Multiplication, Fast Multiplication, Integer Division, Floating-point Numbers and Operations. **8 Hrs**

Unit-V

Basic Processing Unit: Some Fundamental Concepts, Execution of a Complete Instruction, Multiple Bus Organization, Hard-wired Control, Micro programmed Control. Pipelining, Embedded Systems and Large Computer Systems: Basic Concepts of pipelining, Examples of Embedded Systems, Processor chips for embedded applications, Simple Microcontroller. **7 Hrs**

Reference Books:

4) Carl Hamacher, Zvonko Vranesic & SafwatZaky, "Computer Organization", 5th Edition, Tata Mc Graw Hill, 2011.
5) William Stallings, "Computer Organization & Architecture", 9th Edition, Prentice Hall of India, 2012.
6) Vincent P. Heuring & Harry F. Jordan, "Computer Systems Design and Architecture", 2nd Edition, Pearson education, 2004.

18UCSC304 Introduction to UNIX Operating Systems (2-0-2) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- To provide the student with an exposure on UNIX platform so that various other domain specific project activities can be performed with ease and comfort.
- To provide the student with an exposure on the structure and working principles of UNIX operating systems at introductory level, focusing on OS services, commands and scripting language for administration of UNIX operating system.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the structure and working principles of UNIX operating system.	-	-	13
CO-2	**Use** different UNIX commands and System Calls to perform system	-	14,15	-

	administration and user specified tasks.			
CO-3	**Write** shell scripts to perform different system administrative task.	-	5,13,14,15	-
CO-4	**Write** awk scripts to perform different system administrative task.	-	5,13,14,15	-

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	-	-	-	2.0	-	-	-	-	-	-	-	1.7	2.0	2.0	-

Pre-requisites: None
Contents:

Unit-I

Introduction to Operating Systems: Computer System organization/ architecture; Operating System structure; Operating System operations; Introduction to Process management; Memory management; Storage management; Protection and security; Distributed system; Computing environments. Operating System Services; System calls; System programs; Operating System design and implementation; Operating System structure; Virtual machines; System boot **4L Hrs**

Unit-II

Unix System Architecture, commands and System calls: Unix System structure, Commands for performing various activities related to process, files, directories, devices, password protection, vi editors, and other administrative task. Daemon process. System calls related to process, files and directories. **5L + 3 P Hrs**

Unit-III

Filters in Unix: Paginating Files, head: Displaying the Beginning of a File, tail: Displaying the End of a File, cut: Splitting a File Vertically, paste: Pasting Files, sort Ordering a File, uniq Locate Repeated and Non repeated Lines, tr Translating Characters, An Example: Displaying a Word count List. grep Searching for a Pattern, Basic Regular Expressions (BRE) – An Introduction, Extended Regular Expressions (ERE) and egrep. **6L + 3P Hrs**

Unit-IV

Shell Programming: Environment Variables, Aliases (bash), Command History (bash). Shell Scripts, read and readonly commands, Using Command Line Arguments, exit and Exit Status of Command, The Logical Operators && and || Conditional Execution, The if Conditional, Using test and [ ] to Evaluate Expressions, The case Conditional, expr: Computation and String Handling, $0: Calling a Script by Different names, while: Looping, for: Looping with a List, set and shift: Manipulating the Positional Parameters. **6L +3P Hrs**

KURAKUNDI RAGHAVENDRARAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 09:58:28 +05'30'

Unit-V

Awk Scripting Language: awk program line and script structure, awk's operational mechanism, Records and fields, special variables $0, $1, $2, etc., patterns, The BEGIN and END, Variables, built in variables, built in functions, length, split, getline, print, printf, sprintf, index, system, substr, etc., control structures, operators in awk, associative arrays, writing simple awk scripts, Running awk scripts from the shell.

6L + 3P Hrs

Note: Course teachers' assessment (CTA) consists of study of various commands, mastery over vi editors, structure of UNIX operating system, in depth writing shell and awk scripts for simple administrative task. The report is to be submitted by individual students. Students are expected to spend approximately 26 hours on practice based learning and its evaluation.

Reference Books:

1. Sumitabha Das UNIX Concepts and Applications, Third edition, Tata McGraw Hill, 2003
2. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 8th edition, Wiley India, 2009.
3. Behrouz A. Forouzan and Richard F.Gilberg UNIX and Shell Programming A Text book, Thomson, edition 2003.

18UCSL305 Digital Electronics Laboratory (0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This laboratory course focuses on the following learning perspectives:

- Combinational circuit design and simplification techniques used for realizing them.
- Sequential circuit design and working of a basic storage element.
- Simple circuits using passive elements (resistors, capacitors, inductors).

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		Mapping to POs(1-12) / PSOs (13-16)		
		Substantial Level (3)	Moderate Level (2)	Slight Level (1)
CO-1	**Design and implement** combinational circuit for the problem scenarios.	1,13	2,3	12
CO-2	**Design and implement** sequential circuit for problem scenarios.	1,13	2,3	12
CO-3	**Design and implement** an application circuit to simulate given problem.	1,2	15	16

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	3.0	2.3	2.0	-	-	-	-	-	-	-	-	1.0	3.0	-	2.0	1.0

Pre-requisites: Basic Electronics.

List of Experiments:

1 Study and verification of the truth table of various logic gates.

2 Realization of Boolean Functions:
   i) Simplify the given Boolean expression and to realize it using Basic gates and Universal gates.
   ii) Realize the adder and subtract or circuits using basic gates and universal gates.
   iii) Simplify given Boolean expression using Map Entered Variable (MEV) technique and realize the simplified expression using 8:1 Multiplexers.
   iv) To implement given Boolean function using decoders.
   v) To design and realize the following code converters using basic gates

3 Flip-Flops (Sequential Circuits):
   i) To realize flip-flop conversions.
   ii) Applications Flip-Flops:
      a) To design and implement mod-n synchronous counter.
      b) Design and implement a mod-n asynchronous counter.
      c) To realize and study Shift Registers / Ring counter and Johnson counter.

Reference Books:

5. Donald P Leach, Albert Paul Malvino and Goutam Saha: Digital Principles and Applications, 7th Edition, Tata McGraw Hill, 2010.

6. R D Sudhaker Samuel: Illustrative Approach to Logic Design, Sanguine-Pearson, 2010.
7. Charles H. Roth: Fundamentals of Logic Design, Jr., 5th Edition, Cengage Learning, 2004.
8. M Morris Mano: Digital Logic and Computer Design, 10th Edition, Pearson Education, 2008.

18UCSL306 Data Structues and Applications Laboratory (0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This laboratory course focuses on the following learning perspectives:

- Realization of fundamental data structures like stacks, queues, linked lists and trees.
- Compare and contrast the benefits of dynamic and static data structure implementations.
- Selection of the appropriate data structure for solving a given problem.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Write** programs to solve problems using Pointers and Structures/Unions.	-	14	1,3, 15, 16
CO-2	**Write** programs to solve problems using files.	-	14	1,3, 15, 16
CO-3	**Write** programs to solve problems using stack.	-	14	1,3, 15, 16
CO-4	**Write** programs to solve problems using queue.	-	14	1,3, 15, 16
CO-5	**Write** programs to solve problems using Linked Lists.	-	14	1,3, 15, 16
CO-6	**Write** programs to solve problems using trees.	-	14	1,3, 15, 16
CO-7	**Write** programs to solve problems using Hashing.	-	14	1,3, 15, 16

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.0	-	1.0	-	-	-	-	-	-	-	-	-	-	2.0	1.0	1.0

Suggested list of term works:

The list of experiments is based on the following concepts:

7. Pointers and Structures / Unions.
8. Files
9. Stack
10. Queue
11. Linked Lists
12. Trees
13. Hashing

Reference Books:

4. Aaron M. Tenenbaum, Yedidyah Langsam and Moshe J. Augenstein: Data Structures using C and C ++, Pearson Education / PHI, 2006,
5. E. Balagurusamy: Programming in ANSI C, 7th Edition, Tata McGraw-Hill, 2016.
6. Behrouz A. Forouzan and Richard F. Gilberg: Computer Science: A Structured Programming Approach Using C, 2nd Edition, Ceng

IV Semester

18UMAC400	Engineering Mathematics – IV	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- To provide an insight into applications of conformal mapping, integration of complex functions and application of probability distributions in Engineering.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Construct** and **use** the concepts of analytic function to solve the problems arising in Engineering field.	-	-	1
CO-2	**Utilize** conformal transformation and complex integral to transform irregular domain onto a relatively simple domain.	-	1	-
CO-3	**Apply** discrete and continuous probability distributions in analyzing the probability models arising in engineering field.	-	1	-
CO-4	**Make use** of the correlation and regression analysis to fit a suitable mathematical model for the statistical data.	-	1,2	-
CO-5	**Estimate** the correlation, covariance using joint probability distributions. Also use student's t-distribution, Chi-square distribution as a test of goodness of fit.	-	1,2	-

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.8	2.0	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Pre-requisites: Knowledge of

- Differentiation of Functions
- Integration of Functions.
- Basics of Probability
- Statistical Averages

Contents:

Unit-I

Calculus of complex functions: Review of function of a complex variable, limits, continuity, and differentiability. Analytic functions: Cauchy-Riemann equations in Cartesian and polar forms. Construction of analytic functions: Milne-Thomson method-Problems. **7 Hrs**

Unit-II

Conformal transformations: Introduction. Discussion of transformations: $= e^{z}; w = z^{2}$, $w = z + \frac{1}{z}, z \neq o$). Bilinear transformations- Problems.

Complex integration: Line integral of a complex function, Cauchy's theorem and Cauchy's Integral theorem. **8 Hrs**

Unit-III

Probability Distributions: Review of basic probability theory. Random variables (discrete and continuous), probability mass/density functions. Binomial, Poisson, exponential and normal distributions- problems (No derivation for mean and standard deviation)-Illustrative examples. **8 Hrs**

Unit-IV

Statistical Methods: Correlation and regression-Karl Pearson's coefficient of correlation and rank correlation-problems. Regression analysis- lines of regression–problems.

Curve Fitting: Curve fitting by the method of least squares- fitting the curves of the form= $ax + b; y = ax^{2} + bx + c; y = ax^{b}$. **8 Hrs**

Unit-V

Joint probability distribution: Joint Probability distribution for two discrete random variables, expectation and covariance.

Sampling Theory: Introduction to sampling distributions, standard error, Type-I and Type-II errors. Test of hypothesis for means, student's t-distribution, Chi-square distribution as a test of goodness of fit. **8 Hrs**

Reference Books:

4) E. Kreyszig: Advanced Engineering Mathematics, John Wiley & Sons, 10th Ed.(Reprint) 2016.
5) B.S. Grewal: Higher Engineering Mathematics, Khanna Publishers, 44th Ed., 2017.
6) Srimanta Pal et al: Engineering Mathematics, Oxford University Press, 3rd Edition, 2016.
7) C. Ray Wylie, Louis C. Barrett : “Advanced Engineering Mathematics", 6th Edition, McGraw-Hill Book Co., New York, 1995.
8) S. S. Sastry, Introductory Methods of Numerical Analysis, Prentice Hall of India, 4th Edition 2010.
9) B. V. Ramana: "Higher Engineering Mathematics" 11th Edition, Tata McGraw-Hill, 2010.
10) N. P. Bali and Manish Goyal : A Text Book of Engineering Mathematics, Laxmi Publishers, 7th Ed., 2014.

18UCSC400	ARM Processor	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Understand the internal architecture, instruction set of ARM7 microcontroller, assembling process & implement small programs.
- Design & develop Assembly Language Program /& C program for a given real time application.
- Understand the use of interrupts & other advanced concepts related to ARM7
- Demonstrate working knowledge of the necessary steps and methods used to interface ARM7 to devices such as motors, LCD, ADC, and DAC etc.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the features of embedded systems, architecture of ARM7 and applications.	-	1	-
CO-2	**Write** a program using the instruction set of ARM and THUMB	-	2,5	13

KURAKUNDI RAGHAVENDRARAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 10:03:10 +05'30'

	state to solve the engineering problems.			
CO-3	**Explain** the exception, interrupts and interrupt handling schemes and **write** program to solve simple problems.	-	3	-
CO-4	**Explain** the architectural features of LPC2148 microcontrollers.	-	1,2	13,15
CO-5	**Write** a program to interface hardware to LPC2148 microcontrollers.	-	5	3,12

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.0	1.5	-	2.0	-	-	-	-	-	-	1.0	1.0	-	1.0	-

Pre-requisites: None

Contents:

Unit-I

ARM Embedded Systems and ARM Processor Fundamentals: Evolution of Microcontroller and Microprocessor, The RISC design philosophy, ARM design philosophy, embedded system hardware- AMBA bus protocol, embedded system software- applications. ARM core data flow model, Registers, CPSR-Processor modes, Banked registers. Pipeline- Characteristics. **9 Hrs**

Unit-II

ARM Instruction Set: Fundamentals of ARM instructions, Barrel shifter, Classification and explanation of instructions with examples-Data processing, Branch, Load-store, SWI and Program Status. **7 Hrs**

Unit-III

Introduction to THUMB and ARM Programming: Introduction to THUMB, Differences between ARM and THUMB, Register usage in Thumb, ARM Thumb Interworking. General Structure of ARM assembly module, Assembler directives. Simple ALP programs on Arithmetic & logical operations, Factorial, string operation, sorting, searching, and Scan. **8 Hrs**

Unit-IV

Exception and Interrupt handling schemes: Exception handling- ARM processor exceptions and modes, vector table, exception priorities, link register offsets.

Interrupts- assigning interrupts, interrupt latency, IRQ and FIQ exceptions with example- code for enabling and disabling IRQ and FIQ exceptions, Comparison between exception and interrupts. Interrupt handling schemes- nested interrupt handler, non-nested interrupt handler. Basic interrupt stack design. **7 Hrs**

Unit-V

LPC2148 ARM CPU: Salient features, applications, block diagram, memory mapping. Functional features of Interrupt controller, RTC, USB, UART, I2C, SPI, SSP controllers, watch dog timers and other system control units.

Peripherals - GPIO, PLL & Timers: Features, Register description with example and Applications. **8 Hrs**

Reference Books:

1) Andrew N. Sloss, ARM System Developer's guide, ELSEVIER Publications, 2016
2) William Hohl, ARM Assembly Language, CRC Press.
3) Steve Furber, ARM System-on-chip Architecture by, Pearson Education, 2012
4) James K. Peckol, Embedded Systems: A Contemporary Design Tool, 2008
5) Jonathan W. Valvano, Brookes / Cole, Embedded Microcomputer Systems, Real Time Interfacing, 1999
6) LPC 2148 USER MANUAL.

18UCSC401 Finite Automata and Formal Languages (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Study abstract computing machines, Language representation techniques, regular expressions, grammar constructions and associated theories and tools to realize formal language.
- Employ finite state machines to solve problems in computing.
- Comprehend the hierarchy of problems arising in the computer sciences.
- Understand the Turing theory and its significance.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		Mapping to POs(1-12) / PSOs (13-16)		
		Substantial Level (3)	Moderate Level (2)	Slight Level (1)
CO-1	**Construct** a finite automaton for a given pattern and **explain** its working principles.	-	1,2,3,13	-
CO-2	**Write** regular expressions for given patterns and **explain** different techniques and principles used.	-	1,2,3,13	-
CO-3	**Verify** the properties of given languages using standard procedures and **explain** the language properties.	-	1,15	2, 4
CO-4	**Design** grammar for a given language specification and **explain** the design principles.	-	1,2,3,13	-
CO-5	**Write** lexical analyzer and parser for simple programming constructs using standard compiler writing tools.	-	1,2,3,5	-
CO-6	**Design and verify** pushdown automata for a given language specification and **explain** its underlying working principles.	-	1,2,3,15	-
CO-7	**Design and verify** Turing Machine for a given language specification and **explain** its underlying working principles.	-	1,2,3,15	-

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	1.9	2.0	1.0	2.0	-	-	-	-	-	-	-	2.0	-	2.0	-

Pre-requisites: Knowledge of Programming language (any)

Contents:

Unit-I

Introduction to Finite Automata: Structural Representation. The central concepts of Automata theory – Alphabet, Strings & Languages. Finite Automata: Introduction, Deterministic Finite Automata (DFA), Non-Deterministic Finite Automata (NFA), Equivalence of NFA and DFA, Applications of Finite automata, FA with Epsilon (ϵ)

transitions. **8 Hrs**

Unit-II

Regular Expressions and languages: Regular Expressions, Finite Automata and Regular Expressions, Applications of Regular Expressions. Properties of Regular Languages (RL): Proving Languages not to be Regular. Closure properties of Regular Languages, Decision properties of Regular Languages, Equivalence and Minimization of Automata. **7 Hrs**

Unit-III

Context-Free Grammars (CFG) and Languages (CFL): Context-Free Grammars, Parse Trees, Applications of Context-Free Grammars, Ambiguity in Grammars and Languages.

Compiler Writing Tools: Structure of compiler writing tools like LEX and YACC. Simple programs on LEX and YACC. **9 Hrs**

Unit-IV

Pushdown Automata (PDA): Definition of Pushdown Automata, The languages of a PDA, Equivalence of PDA's and CFG'S, Deterministic Pushdown Automata. Properties of Context Free Languages: Normal forms for Context Free Grammar, Pumping lemma for Context Free Languages, Closure properties of Context Free languages. **8 Hrs**

Unit-V

Turing Machines (TM): Introduction, Design of Turing Machine, Extensions to Basic Turing Machine. **7 Hrs**

Reference Books:

1) John E. Hopcroft, Rajeev Motwani & Jeffrey D. Ullman, “Introduction to Automata Theory, Languages and Computation”, Pearson Education, 3rd Edition, 2013.
2) Elaine A.Rich, “Automata, Computability, and Complexity”, Pearson Publication, 2007.
3) Alfred V Aho, Monica S. Lam, Ravi Sethi & Jeffrey D. Ullman “Compilers Principles, Techniques and Tools”, Pearson Education, 2nd Edition, 2008.
4) John R. Levine and Tony Mason & Doug Brown, “UNIX programming tools”, 2nd Edition, 1992.
5) Peter Linz, “An Introduction to Formal Languages and Automata”, Narosa Publishing House, 5th Edition, 2011.
6) John Martin, “Introduction to languages and theory of computation”, Tata McGraw-Hill, 4th Edition, 2010.

18UCSC402 Object Oriented Programming (4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Object Oriented (OO) concepts/philosophy and its benefits and drawbacks in system development.
- Basic features of Java programming language to implement Object Oriented (OO) Key concepts like ADT/Encapsulation, reusability (Inheritance/Composite Objects), polymorphism etc., and other core basic features.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Prepare** an abstract data type for the given business scenario and **write** simple programs to represent ADT and **use** in the given application scenario**.**	13	1	-
CO-2	**Write** programs to solve given problem using different reusability features like inheritance and composite objects.	2, 14	1, 16	3
CO-3	**Write** a program to solve given problem using utility classes.	2, 14	1	3
CO-4	**Write** a program to solve given problem using abstract classes and differentiate with interfaces.	2, 14, 16	1	3
CO-5	**Write** a program to solve given problem using packages.	2, 14, 16	1	3
CO-6	**Write** a program to solve given problem using exception handling in construction of robust systems.	2, 14	1	3
CO-7	**Use** multithreading concept to solve conflicts due to interleaved execution of threads and write simple programs.	2, 14	1	3
CO-8	**Use** streams concept in developing system that needs facility for storage and retrieval of data.	2, 14	1	3

CO-9	**Design and Develop** GUI based system using applet, frames, events and other support available in AWT / Swings components.	2, 8, 14	1	3

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	3.0	1.0	-	-	-	-	3.0	-	-	-	-	3.0	3.0	-	2.3

Pre-requisites: Knowledge of Programming language (any)

Contents:

Unit-I

Introduction to Object Oriented Paradigm (OOP): Object Oriented Philosophy, Key Concepts of OOP, Encapsulation, Polymorphism, Inheritance. **9 Hrs**

Unit-II

Basic Features of Java - 1: Introduction to JAVA, Data Types, Variables and Arrays, String Handling in Java, Control Structures. **9 Hrs**

Unit-III

Basic Features of Java - 2: Classes, Objects, Methods, Constructors, Overloading methods, Methods and Classes, Inheritance, Packages and Interfaces. **12 Hrs**

Unit-IV

Core Features of Java - 1: Exception Handling, Multi-Threaded Programming, Streams. **10 Hrs**

Unit-V

Core Features of Java - 2: AWT and Swings, Applets, Events **12 Hrs**

Reference Books:

5) Herbert Schildt, "Java-The Complete Reference", 9th Edition, Tata McGraw Hill, 2014.
6) Grady Booch, "Object-Oriented Analysis and Design with Applications", 3rd Edition, Pearson Education, 2007.

18UCSC403	Analysis and Design of Algorithms	(3-0-2) 4

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Analyze the performance of algorithms.
- Demonstrate familiarity with major algorithms and data structures.
- Apply important algorithmic design paradigms and methods of analysis.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the importance of algorithmic/mathematical approach in solving different types of problems.	-	1	-
CO-2	**Analyze** time and space complexity for a given algorithm.	2	-	1
CO-3	**Apply** and **analyze** brute force technique and compare it with other techniques.	2	5	3,13
CO-4	**Apply** and **analyze** divide and conquer technique and compare it with other techniques.	2	5	3,13
CO-5	**Apply** and **analyze** greedy technique and compare it with other techniques.	2	5	3,13
CO-6	**Apply** and **analyze** dynamic programming technique and compare it with other techniques.	2	5	3,13
CO-7	**Apply** and **analyze** backtracking and branch & bound technique and compare it with other techniques.	2	5	3,13

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.5	3.0	1.0	-	2.0	-	-	-	-	-	-	-	1.0	-	-	-

Pre-requisites: Knowledge of Discrete Mathematics and Data Structures

KURAKUNDI RAGHAVENDRA RAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 10:07:29 +05'30'

Contents:

Unit-I

Introduction: Algorithm, Fundamentals of problem solving, Problem types, Principles of Algorithm Design. Analysis framework, Asymptotic notations, Mathematical analysis of Non recursive algorithms, Recurrence relations; Mathematical analysis of recursive analysis.

Brute force strategy: Selection Sort, Bubble sort, string matching **6L + 2P Hrs**

Unit-II

Divide and Conquer: Introduction and General method, Binary search, Merge sort, Quick sort, Matrix multiplication using Stressen's Matrix multiplication.

Basic Traversal and search techniques: Depth First search, Breadth First Search, connected components, labeling of components, Path. **6L + 2P Hrs**

Unit-III

Dynamic Programming: Introduction and General method, Computing a binomial coefficient, Warshall's algorithm, Floyd's algorithm, knapsack problem. **6L + 2P Hrs**

Unit-IV

Greedy Strategy: Introduction and General Method, Knapsack problem, Job sequencing with dead-lines, min cost spanning tree (Prim's & Kruskal's), single source shortest path. Huffman Tree. **6L + 2P Hrs**

Unit-V

Back tracking and Branch and Bound: Introduction General Method for both strategies Back Tracking: Sum of Sub sets, Knapsack problem, Traveling Sales person (TSP).

Limitations of Algorithm Power: Lower bound arguments, decision trees, P, NP and NP Complete Problems**.** **5L + 2P Hrs**

Reference Books:

5. Anany Levitin, "Introduction to the Design and analysis of algorithms", 3rd Edition, Pearson Education, 2011
6. Horowitz, Sahani et.al "Fundamentals of Computer Algorithms", 2nd Edition, Galgotia Publication, 2004.
7. Marks Allen Weiss, "Data Structure and Algorithm Analysis", 3rd Edition, Pearson Education, 2009
8. Thomas H.Cormen, Charles E.Leiserson & Ronald L. Rivest, "Introduction to Algorithms", 2nd Edition, Prentice Hall of India, 2003.

18UCSC404	Operating Systems	(4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): The contents of this course deal with the structure and working principles of generic operating systems at introductory level, focusing on process management, memory management, file system and device management. It also focuses on architecture and programming aspects of Linux based OS at fundamental level.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the fundamental concepts of operating system and **Write** programs to demonstrate working principles of process/threads, related issues using system calls and standard libraries.	-	13,14	1,15
CO-2	**Compare** different scheduling algorithms.	-	2	3,13
CO-3	**Compare** and contrast various memory allocation strategies.	-	2	3,13
CO-4	**Explain** the structure and working principles of a file organization and **Write** programs to demonstrate the various file operations using system calls.	-	13,14	1,15
CO-5	**Explain** the structure and working principles of secondary storage and issues related to protection/access strategies.	-	13	1
CO-6	**Explain** the architecture and working principles of industry standard OS.	-	13	1

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.0	2.0	1.0	-	-	-	-	-	-	-	-	-	1.7	2.0	1.0	-

Pre-requisites: Knowledge of Computer Organization, Digital Electronics and Computer Programming at introductory level.

Contents:

KURAKUNDI RAGHAVENDRA RAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 10:08:56 +05'30'

Unit-I

Process Management: Process concept; Process scheduling; Operations on processes; Multithreading models; Thread Libraries; Threading issues. Process Scheduling: Basic concepts; Scheduling criteria; Scheduling algorithms; Multiple-Processor scheduling; Thread scheduling. **10 Hrs**

Unit-II

Process Synchronization: Synchronization: The Critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors.

Case Study: UNIX process synchronization **10 Hrs**

Unit-III

Deadlocks: System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.

Case Study: UNIX process/thread management and programming related to process creation, conflict management and IPCs using system calls and p-thread libraries.

10 Hrs

Unit-IV

Memory Management: Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing.

Case Study:UNIX memory management. **10 Hrs**

Unit-V

File System, Implementation of File System: File System: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection. Implementing File System: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management

Case Study: UNIX file structure and programming on various file operations like creation, listing attributes, directory listing and lock operations.

Secondary Storage Structures & Protection: Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management. Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights, Capability-Based systems **12 Hrs**

Reference Books:

4. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 8th edition, Wiley India, 2009.

5. William Stallings: Operating Systems: Internals and Design Principles, 6th Edition, Prentice Hall, 2013.

18UCSL405 Object Oriented Programming Laboratory (0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This laboratory course focuses on the following learning perspectives:

- Object Oriented (OO) concepts/philosophy and its benefits and drawbacks in system development.
- Basic features of Java programming language to implement Object Oriented (OO) Key concepts like ADT/Encapsulation, reusability (Inheritance/Composite Objects), polymorphism etc., and other core basic features.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Prepare** an abstract data type for the given business scenario and **write** simple programs to represent ADT and **use** in the given application scenario**.**	13	1	-
CO-2	**Write** programs to solve given problem using different reusability features like inheritance and composite objects.	2, 14	1, 16	3
CO-3	**Write** a program to solve given problem using utility classes.	2, 14	1	3
CO-4	**Write** a program to solve given problem using abstract classes and differentiate with interfaces.	2, 14, 16	1	3
CO-5	**Write** a program to solve given problem using packages.	2, 14, 16	1	3
CO-6	**Write** a program to solve given problem using exception handling in construction of robust systems.	2, 14	1	3
CO-7	**Use** multithreading concept to solve conflicts due to interleaved execution of threads and write simple programs.	2, 14	1	3

CO-8	**Use** streams concept in developing system that needs facility for storage and retrieval of data.	2, 14	1	3
CO-9	**Design and Develop** GUI based system using applet, frames, events and other support available in AWT / Swings components.	2, 8, 14	1	3

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	3.0	1.0	-	-	-	-	3.0	-	-	-	-	3.0	3.0	-	2.3

Pre-requisites: Knowledge of: Registration/Completion of the course Object Oriented Programming.

Suggested Platforms:
Notepad (Non IDE), IDE (JCreator, Net Beans, Eclipse etc) in Windows OS and Linux OS

All programs should:

5. Be written to realize the Object Oriented Philosophy and core Java features.
6. Be written with Java Naming & Coding conventions and well documented.
7. Handle exceptions.
8. Be tested for all possible scenarios.

Course Contents:
Minimum one exercise to cover each course outcome specified above. Minimum 8 experiments to be completed by each student independently covering all course outcomes defined for this course. Course teacher has to publish list of experiments along with individual outcome for every experiments, on the first day of the semester. Examiner may set any problem based on the published term work during tests.

Reference Books:

5) Herbert Schildt, "Java: The Complete Reference: 7th Edition, Tata McGraw Hill, 2007.
6) Kathy Sierra & Bert Bates, "Head First Java", 2nd Edition, O'Reilly, 2009
7) Patrick Niemeyer & Daniel Leuck, "Learning Java", 4th Edition, O'Reilly, 2013
8) Laura Lemay & Charles L. Perkins, "Teach Yourself Java in 21 Days", 7th Edition, Sams Publishing, 2016

18UCSL406	ARM Processor Laboratory	(0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This laboratory course focuses on the following learning perspectives:

- Understand the internal architecture, instruction set of ARM7 microcontroller, assembling process & implement small programs.
- Design & develop Assembly Language Program /& C program for a given real time application.
- Understand the use of interrupts & other advanced concepts related to ARM7
- Demonstrate working knowledge of the necessary steps and methods used to interface ARM7 to devices such as motors, LCD, ADC, and DAC etc.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Execute** assembly level codes for a given specific problem using ARM processor.	-	2, 4	3,15
CO-2	**Execute** embedded C programs for a given specific problem using ARM processor.	-	4,14	15,16
CO-3	**Implement** programs for interfacing with real world devices such as LCD's Keyboards, DAC, ADC, Relays Motors etc.	13	4,5,16	3,12

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	2.0	1.0	2.0	2.0	-	-	-	-	-	-	1.0	3.0	2.0	1.0	1.5

Course Contents:

PART A: Conduct the following experiments to learn ALP using ARM:

- Arithmetic and logical operations
- Interrupts related operations
- Timer related applications.

PART B: Conduct interfacing experiments to learn embedded C for ARM:

- LCD- interfacing
- Stepper Motor Interfacing
- Real time sensors Interfacing
- 7-segment LED interface

Reference Books:

6. Andrew N. Sloss, ARM System Developer's guide, ELSEVIER Publications, 2016
7. William Hohl, ARM Assembly Language, CRC Press.
8. Steve Furber, ARM System-on-chip Architecture by, Pearson Education, 2012
9. James K. Peckol, Embedded Systems: A Contemporary Design Tool, 2008
10. Jonathan W. Valvano, Brookes / Cole, Embedded Microcomputer Systems, Real Time Interfacing, 1999
11. LPC 2148 USER MANUAL.

18UCSL407 Introductory Project (0-0-2) 1

Contact Hours: 24

Course Learning Objectives (CLOs): This course enables the student to identify the community expectations in terms of possible engineering solutions and prepare project proposal.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Identify** the societal problems	-	-	2,6,7,9,12,13
CO-2	**Analyze** real environment and **Formulate** the problem statement.	-	-	9,12,13
CO-3	**Conduct** exhaustive literature survey	-	-	9,12,13
CO-4	**Propose** sustainable engineering solutions	-	-	5,7,12,13
CO-5	**Prepare** the report and communicate effectively through presentation.	-	-	9,10,12,13
CO-6	**Manage** the project in terms of various resources in a particular discipline or in a multi-disciplinary domain.	-	-	11

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	1.0	-	-	1.0	1.0	1.0	-	1.0	1.0	1.0	1.0	1.0	-	-	-

Guidelines for conduction:

6. Team consists of 4-5 students.
7. Students have to choose a guide among the faculty members who are teaching their semester.
8. In consultation with guide, the team should carry out their project work.
9. Final evaluation is based on seminar and report submission.
10. This requires designated committee to monitor the process of conduction

KURAKUNDI RAGHAVENDRA RAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 10:13:31 +05'30'

18UHUC500 Management, Enterprenuership, and IPR (4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- The evolution of IT management and related aspects.
- The scope of entrepreneurship in digital firms.
- The issues and procedures related to intellectual property rights.

.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Study** the principles of management in a given organization.	-	2	-
CO-2	**Describe** and **analyze** the role of staffing and the need for motivation in management	-	2	-
CO-3	**Explain** the role of entrepreneur in establishing an organization.	-	2,6	-
CO-4	**Describe** the importance and provisions of institutional support in establishing an enterprise.	-	2,6	-
CO-5	**Explain** the core principles, procedures and related laws and **apply** IPR for given new idea/invention.	-	5,8,10	-

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	-	2.0	-	-	2.0	2.0	-	2.0	-	2.0	-	-	-	-	-	-

Pre-requisites: Knowledge of humanities course.

Contents:

Unit-I

Engineering and Management: Historical Development of Engineering and Management, Management as synthesis.

Planning, Forecasting and Decision Making: Nature of Planning, foundation of planning, some planning concepts, forecasting, nature of decision making, management science, tools for decision **10Hrs**

Unit-II

Organizing and staffing: Nature of organizing, traditional organizational theory, technology and modern organization structures, staffing technical organization, authority and power; delegation, meeting and committees. Motivation: Motivation, leadership, motivating and leading technical professionals. Controlling: Process of control, financial and non-financial controls. **11 Hrs**

Unit-III

Foundations of Entrepreneurship: Meaning of entrepreneur, functions of entrepreneur, types of entrepreneur, concept of entrepreneurship, role of entrepreneurs in economic development& barriers of entrepreneurship. Small Scale Industry: Definition, characteristics of SSI, role of SSI in economic development, advantages of SSI, steps to start an SSI, impact of liberalization, privatization, and globalization on SSI, definition of ancillary and tiny industry. **11 Hrs**

Unit-IV

Government and Institutional Support: Nature of support from government, objectives and functions of SSI, SIDBI, DIC, single window agency, KIADB, KSSIDC, KSFC. Preparation of Project: Meaning of project identification, project report, contents and formulation, identification of business opportunities, feasibility studies, types and purpose. **10 Hrs**

Unit-V

Intellectual Property Rights: Meaning and forms of intellectual property rights, competing rationale for protection, international conventions and security. Copyright: Meaning of copyright, content of copy right, ownership and rights, period of copyright, assignment and relinquishment of copyright, license, infringement of copy right, fair use, offenses and penalties. Patents: Concept of patent, patentable inventions, procedure for obtaining patent, rights and obligations of patent holders, infringements and remedies, offenses and penalties. Industrial Designs: Definition of design, procedure for registration, rights conferred by registration, infringements, Trademark and related issues. **10 Hrs**

Reference Books:

10) Kenneth C. Laudon and Jane P. Laudon, “Management Information Systems - Managing the Digital Firm”, 8th Edition, Pearson Publications, 2017.

11) Making Intellectual Property Work for Business - Handbook for Chambers of Commerce and Business Associations Setting Up Intellectual Property Services by ICC and WIPO, Paperback, 2012.

18UCSC500	Data Communication	(4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Evolution of network and internet.
- Protocols, applications pertaining to network and internet communication.
- Layered architecture and services.
- Network performance measurement and emerging technologies.

.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the evolution and core operating principles of computer network in terms of architecture, functions, switching techniques and protocols.	-	1	2
CO-2	**Explain** the fundamental concepts of analog and digital communication techniques; analyze the communication channels for errors.	-	1	2, 13
CO-3	**Explain** the core working principles of switching techniques and their applications.	-	1	2,13
CO-4	**Explain** the working principles of peer to peer and logical link control protocols in building networked space.	1	2	3, 13
CO-5	**Compare** and **contrast** the different Medium access control protocols for effective channel utilization.	-	1,2	3,13

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.2	1.4	1.0	-	-	-	-	-	-	-	-	-	1.0	-	-	-

Pre-requisites: NIL

Contents:

Unit-I

Communication Networks, Services, applications and layered architectures: Evolution of Network Architecture and Services: Telegraph Networks and Message Switching, Telephone Networks and Circuit Switching, and the Internet, Computer Networks and Packet Switching, examples of Protocols, Services, and Layering: HTTP, DNS, and SMTP, TCP and UDP Transport Layer Services; The OSI Reference Model: The seven layer OSI Reference Model, Unified View of Layers, Protocols, and Services

Overview of TCP/IP Architecture: TCP/IP Architecture, TCP/IP Protocol: How the layer work together, Protocol Overview; Application Layer Protocols and TCP/IP Utilities **10Hrs**

Unit-II

Digital Transmission Fundamentals: Digital Representation of Information: Block-Oriented Information, Stream Information; Why Digital Communications ?: Comparison of Analog and Digital Transmission, Basic properties of Digital Transmission Systems; Digital Representation of Analog Signals: Bandwidth of Analog Signals, Sampling of an Analog Signal, Digital Transmission of Analog Signals; Characterization of Communication Channels: Frequency Domain Characterization, Time Domain Characterization; Fundamental Limits in Digital Transmission: The Nyquist Signaling Rate, The Shannon Channel Capacity; Line Coding ;Modems and Digital Modulation: Binary Phase Modulation, QAM and Signal Constellations, Twisted Pair, Coaxial Cable, Optical Fiber, Radio Transmission, Infrared Light; Error Detection and Correction: Error Detection, Two Dimensional Parity Checks, Internet Checksum, Polynomial Codes, Standardized Polynomial Codes, Error Detecting Capability of a Polynomial Code. **12 Hrs**

Unit-III

Circuit Switching Networks: Multiplexing: Frequency Division Multiplexing, Time Division Multiplexing, Wavelength-Division Multiplexing; SONET: SONET Multiplexing, SONET Frame Structure; Transport Networks: SONET Networks, Optical Transport networks; Circuit Switches: Space Division Switches, Time Division Switches; The Telephone Network: Transmission Facilities, End to End Digital Services. **10 Hrs**

Unit-IV

Peer-to-Peer Protocols and Data Link Layer: Peer-to-Peer Protocols: Peer –to-Peer Protocols and Service Models; ARQ Protocols and Reliable Data Transfer Service: Stop-and-Wait ARQ, Go-Back-N ARQ, Selective Repeat ARQ; Other Peer-to-Peer Protocols; Data Link Controls: Framing; Point to Point Protocol; HDLC Data link Control; Link Sharing using Packet Multiplexers: Statistical Multiplexing, Speech Interpolation and the Multiplexing of Packetized Speech. **10 Hrs**

Unit-V

Medium Access Control Protocols and Local Area Networks: The Medium Access Control Protocols: Multiple Access Communications; Random Access: ALOHA, Slotted ALOHA, CSMA, CSMA-CD; Scheduling Approaches to Medium Access Control: Reservation Systems, Polling, Token-Passing Rings; Channelization: FDMA, TDMA, CDMA. **10 Hrs**

Reference Books:

1) Alberto Leon Garcia & Indra Widjaja, "Communication Networks: Fundamental Concepts and Key architectures", 2nd Edition, Tata McGraw-Hill, 2001
2) Behrouz A. Forouzan, "Data Communications and Networking", 5th Edition, Tata McGraw-Hill, 2017
3) William Stallings, "Data and Computer Communication", 9th Edition, Pearson Publications, 2013

18UCSC501 Database Management Systems (4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Data models and relational theories
- Database design, programming using SQL/PL-SQL, database architecture and transaction concepts.

.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Compare** the traditional file system and Data Base approach	-	2,13	-
CO-2	**Identify** entities, attributes, their relationships and **prepare** ER model for the given application	2,3,13	-	15

	scenario.			
CO-3	**Write** the queries using relational algebra for the given data manipulation requirement of an RDBMS.	2,3,13,14	-	15
CO-4	**Write** SQL queries using all the standard clauses, correlated queries, aggregate and date related functions for the given application scenario.	2,3,13,14	-	15
CO-5	**Write** triggers, stored procedures and functions for the given application scenario.	2,3,13,14	-	15
CO-6	**Design** database in appropriate normal form for a given application scenario.	2,3,13,14	-	15
CO-7	**Explain** the strategies to deal with the issues related to transaction management and to ensure ACID properties.		1,13	

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.8	3.0	-	-	-	-	-	-	-	-	-	2.7	3.0	1.0	-

Pre-requisites: Knowledge of

- Programming languages
- Set Theory
- File Systems
- Abstract application development process.

Contents:

Unit-I

Introduction: Introduction; An example; Characteristics of Database approach; Actors on the screen; Workers behind the scene; Advantages of using DBMS approach; A brief history of database applications; when not to use a DBMS. Data models, schemas and instances; Three-schema architecture and data independence; Database languages and interfaces; The database system environment; Centralized and client-server architectures;

Entity-Relationship Model: Using High-Level Conceptual Data Models for Database Design; An Example Database Application; Entity Types, Entity Sets, Attributes and

Keys; Relationship types, Relationship Sets, Roles and Structural Constraints; Weak Entity Types; Refining the ER Design; ER Diagrams, Naming Conventions and Design Issues; Relationship types of degree higher than two. **10Hrs**

Unit-II

Relational Model and Relational Algebra: Relational Model Concepts; Relational Model Constraints and Relational Database Schemas; Update Operations, Transactions and dealing with constraint violations; Unary Relational Operations: SELECT and PROJECT; Relational Algebra Operations from Set Theory; Binary Relational Operations: JOIN and DIVISION; Examples of Queries in Relational Algebra; Relational Database Design ER-to-Relational Mapping. **10 Hrs**

Unit-III

SQL: Data Definition and Data Types; DDL statements like creation and specification od table, DCL statements for Schema change, alter, delete etc; DML statements like Insert, Delete and Update statements in SQL etc and more complex statements for Basic queries Nested sub queries, Correlated sub queries. PL/SQL; Specifying constraints as Assertion and Trigger; Views (Virtual Tables) in SQL; Additional features of SQL; Database programming issues and techniques; Embedded SQL, Dynamic SQL; Database stored procedures and SQL. **12 Hrs**

Unit-IV

Database Design – 1: Informal Design Guidelines for Relation Schemas; Functional Dependencies; Inference rule, Equivalence of sets, Minimal set cover. Normal Forms Based on Primary Keys; General Definitions of Second and Third Normal Forms; Boyce-Codd Normal Form. **10 Hrs**

Unit-V

Database Design –2: Properties of Relational Decompositions; Dependency preservation, Lossless (non additive) join properties, Problem with null values and dangling tuples.

Transaction Management: The ACID Properties; Transactions and Schedules; Concurrent Execution of Transactions; Lock- Based Concurrency Control; Performance of locking; Transaction support in SQL; Introduction to crash recovery; 2PL, Serializability and Recoverability; Lock Management; Introduction to ARIES; The log; Other recovery-related structures; The write-ahead log protocol; Check pointing.

10 Hrs

Reference Books:

1. Elmasri & Navathe, “Fundamentals of Database Systems”, 6th Edition, Addison-Wesley, 2012.
2. Raghu Ramakrishnan & Johannes Gehrke, “Database Management Systems”, 3rd Edition, McGraw-Hill, 2003.

3. Silberschatz, Korth and Sudharshan, “Data base System Concepts”, 6th Edition, Mc-Graw Hill, 2010.
4. C.J. Date, A. Kannan & S. Swamynatham, “An Introduction to Database Systems”, 8th Edition, Pearson Education, 2006.

18UCSC502 Compiler Design and System Software (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This is a 3 credit course at undergraduate level enabling the students to understand structure of a compiler,representation of patterns and syntax using lexical rules and grammars respectively, working of parsers, translation schemes, code optimization and code generation, working of assemblers, loaders, linkers and macro processor.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the basic structure and working principles of phases of compiler.	-	13	1
CO-2	**Write** a parser for the given input based on the appropriate parsing technique and **validate** the design	13,14	1,2,3	15
CO-3	**Generate** an optimized intermediate code.	-	1,2,3,13	15
CO-4	**Explain** the working principles of run time environments that include stack allocation, heap management and garbage collection technique used in compiler.	-	1,2,3,13	15
CO-5	**Generate** optimized code for the given intermediate code	-	1,2,3,13	15
CO-6	**Design** Assembler for the given language specification and **validate** the design.	-	1,2,3,13,14	15
CO-7	**Design** Macroprocessor for the given language specification and **validate** the design.	-	1,2,3,13,14	15
CO-8	**Explain** the working principles of Linkers & Loaders for the given language specification.	-	1,2,3,13,14	15

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.9	2.0	2.0	-	-	-	-	-	-	-	-	-	2.1	2.3	1.0	-

Pre-requisites: Knowledge of

- Finite Automata and Formal languages
- Programming language (any)

Contents:

Unit-I

Introduction: Different Phases of Compilers, Comparison of Compilers and Interpreters. Top-down Parsing: RDP and Predictive parsing. **7 Hrs**

Unit-II

Bottom-up Parsing: Simple LR, LALR, CLR, parsers ambiguous grammars. **8 Hrs**

Unit-III

Intermediate Code Generation and Optimizations: Syntax-directed translation; Syntax-directed translation schemes, Variants of syntax trees; Three-address code; Types and declarations; Translation of expressions; Type checking; Control flow; Various techniques of machine independent optimization. **8 Hrs**

Unit-IV

Run-Time Environments: Storage Organization; Stack allocation of space; Access to non-local data on the stack; Heap management; Introduction to garbage collection.
Code Generation: Issues in the design of Code Generator; The Target language; Addresses in the target code; Basic blocks and Flow graphs; Optimization of basic blocks; A Simple Code Generator **8 Hrs**

Unit-V

Ancillary Code Processing Techniques: Generic description of Assembler, Loader, Linker and Macro's. Assemblers: Basic Assembler Features & Functions and Design of assembler. Loaders and Linkers: Basic Loader Functions - Design of Loaders and Linkers Macro Processor: Design of Macro Processors. **8 Hrs**

Reference Books:

1) Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D Ullman,“Compilers-Principles, Techniques and Tools”, 2/E, Addison-Wesley, 2007.
2) D.M.Dhamdhere, “System Programming and Operating Systems”, 2nd revised edition, Tata McGraw - Hill, 2009 reprint.
3) Leland L Beck, “System Software : An Introduction to Systems Programming” 3rd Edition Pearson Education 2007
4) John J Donovan, “System Programming”, Tata McGraw-Hill 2017

18UCSC503 Software Engineering (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This is a 3 credit, 39 contact hours course at undergraduate level focusing on knowing the process of software system development and enables students to develop software system using engineering techniques.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Describe** the properties of various critical systems and the quality parameters.	-	-	16
CO-2	**Describe** the different key practices of process models.	-	3,13	
CO-3	**Identify** various system requirements and **prepare** system specification reports to solve real life problems in various domains and **develop** domain expertise.	1,2,13	-	-
CO-4	**Conceptualize** the system through **design** and **modeling** the system architecture, components and processes with quality and standards.	1,2,3,13	5	10,12
CO-5	**Develop** software system using engineering techniques, industry relevant tools and programming features/techniques.	1,2,3,13,14	5,15	10,11,12,16
CO-6	**Verify** and **validate** the given	-	5,15	10

	system using standard tools and techniques.			
CO-7	**Manage** project in terms of risk, configuration/versions, Cost and Resources.	-	9,11	10

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	3.0	3.0	2.7	-	2.0	-	-	-	2.0	1.0	1.5	1.0	2.8	3.0	2.0	1.0

Pre-requisites: Knowledge of

- Basics of computer systems and its uses
- Programming language (any)

Contents:

Unit-I

Overview: Introduction: FAQ's about software engineering, Professional and ethical responsibility. Socio-Technical systems: Emergent system properties; Systems engineering; Organizations, people and computer systems; Legacy systems.

Critical Systems, Software Processes: Critical Systems: A simple safety critical system; System dependability; Availability and reliability. Software Processes: Models, Process iteration, Process activities; The Rational Unified Process; Computer Aided Software Engineering. **7 Hrs**

Unit-II

Requirement Engineering: Software Requirements: Functional and Non-functional requirements; User requirements; System requirements; Interface specification; The software requirements document. Requirements Engineering Processes: Feasibility studies; Requirements elicitation and analysis; Requirements validation; Requirements management. **7 Hrs**

Unit-III

System models: System Models: Context models; Behavioral models; Data models; Object models; Structured methods.

Software Design and Development: Architectural Design: Architectural design decisions; System organization; Modular decomposition styles; Control styles. Object-Oriented design. UI Design Issues.

Rapid Software Development: Agile methods; Extreme programming; Rapid application development. Software Evolution: Program evolution dynamics; Software maintenance; Evolution processes; Legacy system evolution. **9 Hrs**

Unit-IV

Verification and Validation: Verification and Validation: Planning; Software inspections; Automated static analysis; Verification and formal methods. Software testing: System testing; Component testing; Test case design; Test automation. Testing Techniques: Equivalence Partitioning, Boundary Value Analysis,, Cause Effect Graphing, Test Generation from Predicates, Statement testing, Branch Testing, Condition Testing, Path Testing, Procedural Call Testing, Data Flow Testing. **9 Hrs**

Unit-V

Software Quality & Project Management: Various Software quality parameters and associated standards and procedures, Project Management activities; Project planning; Project scheduling; Risk management. Configuration Management, Managing People: Selecting staff; Motivating people; Managing people; The People Capability Maturity Model. Software Cost Estimation: Productivity; Estimation techniques, Project duration and staffing. **7 Hrs**

Reference Books:

5. Ian Somerville, "Software Engineering", 10/E, Person Education, 2016.
6. Roger Pressman, "Software Engineering, Practitioners approach", 7/E, McGraw-Hill, 2010.
7. Bharat Bhushan Agarwal & Sumit Prakash Tayal, "Software Engineering", 2nd Edition, Firewal Media Publications, 2007.
8. A.A.Puntambekar, "Software Engineering & Quality Assurance", 1st Edition, Technical Publications Pune, 2010

18UCSL504 Database Management Systems Lab (0-0-3)1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This laboratory course focuses on hands on experience on creation of data models, database design, programming using SQL/PL-SQL and development of an application using any high level language.

Course outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Identify** entities, attributes, their relationships and **prepare** ER model for the given problem.	2, 3, 13	-	15
CO-2	**Design** database in appropriate	2,3,13	-	15

	normal form for the given problem.			
CO-3	**Write** SQL queries using all the standard clauses, correlated queries, aggregate and date related functions for the given application scenario.	2,3,13,14	-	15
CO-4	**Write** the programs using advanced features of data base programming that includes PL/SQL, Cursors, Triggers, Stored procedures and Functions for given application scenario.	2,3,13,14	-	15

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	3.0	3.0	-	-	-	-	-	-	-	-	-	3.0	3.0	1.0	-

Prerequisites: Registration for / Completion of: DBMS theory course

Contents:

Part A: Multiple standalone preparatory exercises to meet the course outcomes.

Part B: Project work (Standalone application or web enabled application) based on knowledge gained from theory course and part-A hands on experience.

18UCSL505 Compiler Design and System Software Lab (0-0-3)1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This laboratory course focuses on representation of patterns and syntax using lexical rules and grammars respectively, Implementation of parser & translation schemes, Implementation of assemblers, loaders, linkers & macro processor, Knowledge of system level APIs for implementation of IPC and system commands.

Course outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		Mapping to POs(1-12)/ PSOs (13-16)		
		Substantial Level (3)	Moderate Level (2)	Slight Level (1)
CO-1	**Prepare** the grammar for the given constructs and **Write** a program using compiler writing tools to implement lexical analyzer and parser.	13,14	1,2,3	15
CO-2	**Write** a program to **implement** a parser.	13,14	1,2,3	15
CO-3	**Write** a program to implement assembler functions.	13,14	1,2,3	15
CO-4	**Write** a program to **Implement** various UNIX commands using system calls.	13,14	1,2,3	15
CO-5	**Use** IPC concepts in implementing communication protocol.	13,14	1,2,3	15

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.0	2.0	-	-	-	-	-	-	-	-	-	3.0	3.0	1.0	-

Prerequisites: Knowledge of: Unix Operating System, Any programming language, and Finite automata and formal Languages

Contents:

1. Programs on Lex and Yacc.
2. Implementation of parser.
3. Implementation of assembler.
4. Emulation of basic commands of UNIX using system calls.
5. Application development using Inter Process Communication.

18UCSL506 Minor Project - 1 (0-0-2)1

Contact Hours: 24

Course Learning Objectives (CLOs): This course enables the student to identify the community expectations in terms of possible engineering solutions and prepare project proposal.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		Mapping to POs(1-12) / PSOs (13-16)		
		Substantial Level (3)	Moderate Level (2)	Slight Level (1)
CO-1	**Identify** the societal problems.	-	2,6,7,9,12,13	-
CO-2	**Analyze** real environment and **Formulate** the problem statement.	-	2,9,12,13	-
CO-3	**Conduct** exhaustive literature survey	-	2, 9,12,13	-
CO-4	**Propose** sustainable engineering solutions / prototypes.	-	3,5,7,12,13	-
CO-5	**Prepare** the report and communicate effectively through presentation.	-	8,9,10,12	-
CO-6	**Manage** the project in terms of various resources in a particular discipline or in a multi-disciplinary domain.	-	11	-

POs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	2.0	2.0	-	2.0	2.0	2.0	2.0	2.0	2.0	2.0	2.0	2.0	-	-	-

Guidelines for conduction:

1. Team consists of 4-5 students.
2. In consultation with guide, the team should carry out their project work.
3. Final evaluation is based on following evidence will be looked into and accordingly rubrics will be developed, such as problem statement, design, prototype/part of implementation, use of standard tools and techniques for testing and validation procedure.
4. This requires designated committee to monitor the process of conduction

18UHUL507 Soft Skills / Aptitude (0-0-2) 1

Contact Hours: 24

Course Learning Objectives (CLOs):

This is included with the objectives of improving the communication skills, proficiency in English language and aptitude ability of the student to enhance the

employability.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSO(13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the significance of communication in the profession.	-	10	-
CO-2	**Use** the English language with proficiency	-	10	12
CO-3	**Solve** Aptitude related problems	-	9	12
CO-4	**Demonstrate** the competency in the placement activities.	-	9	-

POs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	-	-	-	-	-	-	-	-	2.0	2.0	-	1.0	-	-	-	-

Contents:

Training on communication skills, proficiency in English language and aptitude ability involving the internal and external resource.

Evaluation:

Both the internal and external resource persons shall be engaged in imparting the related knowledge and shall have only CIE as the evaluation component. There shall be one test conducted at the end for 25 marks in Aptitude testing and there shall be one presentation by the student for 25 marks or any other suitable testing components. The arrangement for CIE evaluation is to be done by the department and maintain the relevant documents.

KURAKUNDI RAGHAVENDRA RAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 10:29:05 +05'30'

ELECTIVES

18UCSE508 Advanced Object Oriented Programming (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on core and advanced Java language features that are part of JDK 8 and above.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Build** graphical user interface using JavaFX for a given problem.	13,14,15	-	1, 2,3, 5, 9,16
CO-2	**Develop** applications that involve parallel programming abilities using concurrent utility feature.	13,14,15	-	1, 2,3, 5, 9,16
CO-3	**Write** programs to solve a given problem using generics and collection Frameworks.	13,14,15	-	1, 2,3, 5, 9,16
CO-4	**Use** Java networking features to **write** applications that involve client / server interactions.	13,14,15	-	1, 2,3, 5, 9,16
CO-5	**Develop** an application that use appropriate driver classes to connect databases and **perform** database operations required as per problem specification.	13,14,15	-	1, 2,3, 5, 9,16
CO-6	**Develop** web-based applications using J2EE features like Servlets and JSP.	13,14,15	-	1, 2,3, 5, 9,16
CO-7	**Write** program using lambda expressions to solve given problem scenario.	13,14,15	-	1, 2,3, 5, 9,16
CO-8	**Write** program using stream APIs to solve given problem scenario.	13,14,15	-	1, 2,3, 5, 9,16

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.0	1.0	1.0	-	1.0	-	-	-	1.0	-	-	-	3.0	3.0	3.0	1.0

Pre-requisites: Knowledge of

- Basic features of Java

- Object oriented programming paradigm, its concepts and practices

Contents:

Unit-I

GUI programming with JavaFX: Introducing JavaFX GUI programming – JavaFX basic concepts, JavaFX application skeleton, the Application thread; Exploring JavaFX controls – ToggleButton, RadioButton, CheckBox, ListView, TextField, TreeView, disabling a control; Introducing JavaFX menus – menu basics, overview of MenuBar, Menu and MenuItem, create main menu, add mnemonics and accelerators to menu items, use RadioMenuItem and CheckMenuItem, create ContextMenu and Toolbar **7 Hrs**

Unit-II

Concurrent Utilities: concurrent API packages, using synchronization objects, phaser, using an executer, TimeUnit enumeration, concurrent collections, locks, atomic operations, parallel programming via Fork/Join framework.

Generics and Collections Framework: Generics - what are generics, a simple generics example, a generic class with two type parameters, general form of general class, bounded types, using wildcard arguments, creating a generic method; Collections Framework – collections overview, the collection interfaces, the collection classes, accessing a collection via an iterator, for-each alternative to iterators, spliterators, storing user-defined classes in collections. **9 Hrs**

Unit-III

Networking and RMI: networking basics, the networking classes and interfaces, InetAddress class, TCP/IP client sockets, HttpURLConnection class, TCP/IP server sockets, Datagrams; Remote Method Invocation (RMI) – A simple client/server application using RMI

J2EE Databases: the concept of JDBC, JDBC driver types, a brief overview of JDBC process, database connection, Statement objects, ResultSet class, ResultSetMetaData class **7 Hrs**

Unit-IV

Java Servlets: Introduction,benefits of Java servlets, a simple java servlet, anatomy of a Java servlet, deployment descriptor, reading data from a client, reading/writing HTTP request/response headers; working with cookies, tracking sessions

Java ServerPages: Introduction,JSP tags, Request string, User sessions, cookies, session objects **9 Hrs**

Unit-V

Extended features of Java: Lambda Expressions – introducing lambda expressions,

block lambda expressions, passing lambda expressions as arguments, lambda expressions and exceptions, lambda expressions and variable capture, method references; The Stream API – stream basics, reduction operations, using parallel streams, mapping, collecting, iterators and streams. **7 Hrs**

Reference Books:

1) Herbert Schildt, "Java: The Complete Reference", 10th edition, McGraw-Hill, 2017
2) Jim Keogh, "J2EE: The Complete Reference", McGraw-Hill, 2011
3) Gregory Brill, "CodeNotes for J2EE: EJB, JDBC, JSP, and Servlets", Random House Publishing Group, 2002.
4) John Hunt & Chris Loftus, "Guide to J2EE: Enterprise Java", Springer Publications, 2012.

18UCSE509 System Simulation and Modeling (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- To introduce students to basic simulation methods and tools for modelling and simulation of continuous, discrete and combined systems.
- The ability to analyze a system and to make use of the information to simulate various systems to improve the performance.
- Analytical methods (Markov Models and Queuing Networks) and simulation techniques (Monte Carlo Techniques and Event Driven Simulation) applied in performance modelling of communication systems and networks.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Conceptualize** the discrete system simulation with the aid of real time examples.	-	5,13	1,2,3,15
CO-2	**Write** a program to simulate a given scenario.	-	5,13,14	1,2,3,15
CO-3	**Apply** different statistical models available in simulation and their usage in specific applications.	-	5,13,14	1,2,3,15

CO-4	**Design** the queuing systems and **evaluate** the performance.	-	5,13	1,2,3,15
CO-5	**Identify** the distribution of data to adhere to fitness test and also analyze the corresponding simulation.	-	5,13	1,2,3,15

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.0	1.0	1.0	-	2.0	-	-	-	-	-	-	-	2.0	2.0	1.0	-

Pre-requisites: Knowledge of Probability and Statistics

Contents:

Unit-I

Introduction: When simulation is the appropriate tool and when it is not appropriate; Advantages and disadvantages of Simulation; Areas of application; Systems and system environment; Components of a system; Discrete and continuous systems; Model of a system; Types of Models; Discrete-Event System Simulation; Steps in a Simulation Study. Simulation examples: Simulation of queuing systems; Simulation of inventory systems; other examples of simulation. General Principles. **7 Hrs**

Unit-II

oncepts in Discrete-Event Simulation: The Event-Scheduling, Time-Advance Algorithm, World Views, Manual simulation Using Event scheduling; List processing. **7 Hrs**

Unit-III

Statistical Models In Simulation: Review of technology and concepts; Useful statistical models; discrete distributions; Continuous distributions; Poisson process; Empirical distributions. **8 Hrs**

Unit-IV

ueuing Models: Characteristics of queuing systems; Queuing notation; Long-run measures of performance of queuing systems; Steady state behavior of M/G/1 queue; Networks of queues. **8 Hrs**

Unit-V

Input Modeling: Data Collection; Identifying the distribution with data; Parameter estimation; Goodness of Fit Tests; Fitting a non-stationary Poisson process; selecting input models without data; Multivariate and Time-Series input models.

Estimation of absolute performance: Types of simulations with respect to output analysis; stochastic nature of output data; Measures of performance and their

estimation; Output analysis for terminating simulations; Output analysis for steady-state simulations. **9 Hrs**

Reference Books:

1) Banks, John S. Carson II, Barry L. Nelson- David M. Nicol, “Discrete-Event System Simulation”, 5th Edition, Pearson Education, 2013.

2) J. A. Sokolowski, C.M. Banks, “Principles of Modeling and Simulation: A multidisciplinary Approach”, John Wiley & Sons Publications, 2011.

3) Sheldon M. Ross, “Simulation”, 4th Edition, Elsevier, 2006.

4) D.S.Hira, “System Simulation”, 2nd Edition, S.Chand Publications, 2008.

18UCSC600	Computer Networks	(4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Various network services and switching networks.
- Protocol design, implementation and performance issues.
- Various network management issues and possible remedies.
- Virtual networks for security issues in Internet Protocol (IP).

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the communication services, protocols and algorithms of packet-switching networks.	-	1,2,13	-
CO-2	**Explain** the working of TCP/IP layered model of communication and **analyze** the traffic management.	4,5	1,2,13	14
CO-3	**Explain** the issues of internet routing protocols, VPN and overlay networks.	-	1,2,13	-
CO-4	**Explain** the working of ATM layered model of communication.	-	1,2,13	-
CO-5	**Explain** the need of network management services, associated security issues and **use** modern tools to **perform** network management.	4,5	1,2,13	-

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	2.0	2.0	-	3.0	3.0	-	-	-	-	-	-	-	2.0	1.0	-	-

Pre-requisites: Knowledge of

- Data Communication
- Operating Systems

KURAKUNDI RAGHAVENDRA RAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 10:43:24 +05'30'

Contents:

Unit-I

Packet-Switching Networks: Network services and internal network operations, Packet network topology, Datagrams and virtual circuits, Routing in packet networks, Shortest-path routing, ATM networks. **10 Hrs**

Unit-II

Traffic management at the packet level: Traffic management at the flow level, Traffic management at the flow-aggregate level.

TCP / IP: TCP / IP architecture, The Internet protocol, User datagram protocol, Transmission control protocol. **12 Hrs**

Unit-III

Internet routing protocols, Multicast routing, DHCP, NAT and Mobile IP.

VPNs, Tunneling, Overlay Networks: Virtual Private Networks, Multiprotocol Label switching, Overlay networks. **10 Hrs**

Unit-IV

ATM Networks: Introduction to ATM networks, BISDN reference model, ATM layer, ATM adaptation layer, ATM signaling, PNNI routing, Classical IP over ATM**.** **10 Hrs**

Unit-V

Network Management, Security: Network management overview, SNMP, Structure of Management information, MIB, Remote network monitoring, Overview of Security and cryptographic algorithms. **10 Hrs**

Reference Books:

1. Alberto Leon – Garcia & Indra Widjaja, “Communication networks – Fundamental Concepts and Key Architecture”, 2nd Edition, Tata McGraw Hill, 2005.
2. Nader F.Mir, “Computer and Communication Networks”, 2nd Edition, Pearson Education, 2009
3. Behrouz A Forouzon, “Data Communications and Networking”, 10th Edition, Tata McGraw Hill, 2006
4. William Stallings, “Data and Computer Communication”, 8th Edition, Pearson Education, 2013.

18UCSC601 Object Oriented System Modelling and Design (4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This is a 4 credit, 52 contact hours course at undergraduate level focusing on the process of object oriented system modeling, design, patterns and tools used in the industry to enable them to construct software system using various standards and techniques.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Apply** fundamental Object Oriented concepts in solving problems.	13,14	1,2,3,15	-
CO-2	**Analyze** problem scenario and **identify** classes/ Objects, their properties and associations.	13	1,2,3,5,15	-
CO-3	**Analyze** problem scenario and **model** the system using UML diagrams.	13	1,2,3,5,15	-
CO-4	**Evaluate** the quality of Object Oriented system in-terms of Cohesion, coupling, sufficiency, completeness and primitiveness.	-	16	15
CO-5	**Implement** Object Oriented model in any Object Oriented language.	13,14	1,2,3	15
CO-6	**Identify** and **apply** the appropriate patterns in solving problems.	-	13,16	1
CO-7	**Propose** the appropriate strategies to incorporate standard quality parameters in the design of a system.	-	13,16	1

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.7	2.0	2.0	-	2.0	-	-	-	-	-	-	-	2.7	3.0	1.6	2.0

Pre-requisites: Knowledge of

- Object Oriented Programming Language (any)
- Software Engineering

KURAKUNDI RAGHAVENDRA RAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 10:45:27 +05'30'

Contents:

Unit-I

Review: Object Oriented Concepts and principles.

Introduction, modeling concepts, class modeling: Object Orientation, developments themes; Evidence for usefulness of developments; modeling history. Modeling as Design Technique: Modeling; abstraction; The three models. Class Modeling: Object and class concepts; Link and associations concepts; Generalization and inheritance; A sample class model; Navigation of class models; Practical tips.

10 Hrs

Unit-II

Advanced class modeling, state modeling: Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived data; Packages; Practical tips. State Modeling: Events, States, Transitions and Conditions; State diagrams; State diagram behavior; Practical tips.

Advanced state modeling, interaction modeling: Advance state modeling: nested state diagrams; nested states; signal generalization; concurrency; a sample state model; relation of class and state models; practical tips. Interaction modeling: use case models; sequence models; activity models. Use case relationships; procedural sequence models; special constructs for activity models.

Evaluation: OO system quality in terms of Cohesion, coupling, sufficiency, completeness and primitiveness.

Implementation : OO design in appropriate language **12 Hrs**

Unit-III

Patterns–Part 1: Introduction; layers, pipes and filters, blackboard.Distributed systems: broker; interactive systems: mvc, presentation-abstraction-control. **10 Hrs**

Unit-IV

Patterns—Part2: Adaptable systems: microkernel; reflection. Structural decomposition**:** whole - part; organiz**a**tion of work: master - slave; access control: proxy. Others: Command Processor, View Handler, Forward Receiver, Client-Dispatcher-Server and publish Subscriber. **10 Hrs**

Unit-V

Quality: Functionality and architecture; architecture and quality attributes; system quality attributes; Quality attribute scenarios in practice; Other system quality attributes; Business qualities; Architecture qualities. Achieving Quality: Introducing tactics; Availability tactics; Modifiability tactics; Performance tactics; Security tactics; Testability tactics; Usability tactics; Relationship of tactics to architectural patterns; Styles. **10 Hrs**

Reference Books:

1) Michael Blaha & James Rumbaugh, "Object-Oriented Modeling and Design with UML", 2nd Edition, Pearson Education, 2007.
2) Frank Buschmann, RegineMeunier, Hans Rohnert, Peter Sommerlad & Michael Stal, "Pattern-Oriented Software Architecture", A System of Patterns - Volume 1, John Wiley and Sons, 2006.
3) Len Bass, Paul Clements & Rick Kazman, "Software Architecture in Practice", 2nd Edition, Pearson Education, 2003.
4) Grady Booch et al, "Object-Oriented Analysis and Design with Applications", 3rd Edition, Pearson Education, 2007.
5) Ali Bahrami, "Object oriented systems development", McGrawHill, 1999.
6) Mary Shaw and David Garlan, "Software Architecture Perspectives on an Emerging Discipline", Prentice-Hall of India, 2007.

18UCSL602 Computer Networks Lab (0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Configuration of networking devices.
- Troubleshooting IPv4 and IPv6 Addressing
- DHCP and DNS Servers

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Configure** the Cisco networking devices like routers, switches, and hubs etc using commands.	13,14	1,2,3	5,15
CO-2	**Simulate** different topologies/network using Cisco Packet Tracer application.	13,14	1,2,3	5,15
CO-3	**Manage** IP addresses and troubleshooting	13,14	1,2,3	5,15
CO-4	**Manage** applications like Web, Email, DHCP, DNS and FTP.	13,14	1,2,3	5,15
CO-5	**udy** on Industry relevant tools to perform traffic management.	13,14	1,2,3	5,15
CO-6	**rite** C program using RFCs to implement standard protocol using	13,14	1,2,3	5,15

	TCP / IP.			

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.0	2.0	-	1.0	-	-	-	-	-	-	-	3.0	3.0	1.0	-

Pre-requisites: Knoweldge of Data Communication and Computer Networks (registration).

Contents:

Sl. No.	Term Work
T1	**Building a Simple Network:** Part 1: Set Up the Network Topology (Ethernet only) Part 2: Configure PC Hosts Part 3: Configure and Verify Basic Switch Settings **Learning Outcomes: [CO-1,2]**
T2	**Connecting a Wired and Wireless** Part 1: Connect to the Cloud Part 2: Connect Router0 Part 3: Connect Remaining Devices Part 4: Verify Connections Part 5: Examine the Physical Topology **Learning Outcomes: [CO-1,2]**
T3	**Troubleshooting IPv4 and IPv6 Addressing** Part 1: Troubleshoot First Issue Part 2: Troubleshoot Second Issue Part 3: Troubleshoot Third Issue **Learning Outcomes:[CO-2,3]**
T4	**Configuring IPv6 Addresses on Network Devices** Part 1: Set Up Topology and Configure Basic Router and Switch Settings Part 2: Configure IPv6 Addresses Manually Part 3: Verify End-to-End Connectivity **Learning Outcomes:[CO-2,3]**
T5	**Designing and Implementing a Subnetted IPv4 Addressing Scheme** Part 1: Design a Network Subnetting Scheme Part 2: Configure the Devices

	Part 3: Test and Troubleshoot the Network **Learning Outcomes:[CO-1,2,3]**
T6	**Web and Email** Part 1: Configure and Verify Web Services Part 2: Configure and Verify Email Services **Learning Outcomes:[CO-3,4]**
T7	**DHCP and DNS Servers** Part 1: Configure Static IPv4 Addressing Part 2: Configure and Verify DNS Records **Learning Outcomes:[CO-3,4]**
T8	**FTP Servers** Part 1: Configure FTP Services on Servers Part 2: Upload a File to the FTP Server Part 3: Download a File from the FTP Server **Learning Outcomes:[CO-3,4]**
T9	**Troubleshooting Connectivity Issues** The objective of this Packet Tracer activity is to troubleshoot and resolve connectivity issues, if possible. Otherwise, the issues should be clearly documented and so they can be escalated. **Learning Outcomes:[CO-1,2,3,4]**
T10	**udy** on Industry relevant tools to perform traffic management. **Learning Outcomes:[CO-5]**
T11	**Write** C program using RFCs to implement standard protocol using TCP / IP. **Learning Outcomes:[CO-6]**

18UCSL603 Industry Oriented Programming Practices Lab (0-0-3) 1.5

Contact Hours: 36

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- The study of the various programming practices through demonstration,
- The implementation in different programming paradigm/languages in

terms of principles and benefits it offers in the system design and development.

- The coding guidelines encompassing all aspects of code development to enable them to be a professional software developer.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the various styles, standards of different programming paradigm and **Write** simple programs.	-	1,2,3,13,14,15	5
CO-2	**Write** simple programs to explain the code quality and assess the quality of the given code.	-	1,2,3,13,14,15	5
CO-3	**Illustrate** the need for parallelization of serial parograms and its impact on performance.	-	1,2,3,13,14,15	5
CO-4	**Write** simple scripts for given system administration.	-	1,2,3,13,14,15	5
CO-5	**Generate** the technical reports and effectively communicate through presentation slides and tools.	10	-	-

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	2.0	2.0	2.0	-	1.0	-	-	-	-	3.0	-	-	2.0	2.0	2.0	-

Pre-requisites: Knowledge of any computer programming and Software Engineering is essential. Introductory level exposure to different domains like OS, Network, DBMS, and Web Technology is desirable.

Contents:

Naming convention & consistency; comments; Consistent indentations; Spaces; Structure and its impact on readability and efficiency; Organization of: Programs, Function, File and folder. Guidelines for portability and performance; Separation of code and data; Documentation; Writing reusable codes; Programming Paradigm specific style and practices (Object orientation V/S procedural etc.).;Writing code with quality in terms of: Robustness, Maintainability, Testability, Adaptability, Availability, Usability, sufficiency, completeness, Primitiveness, Cohesiveness, and Coupling; Communications, Documentations,

Proposals/ technical writing, Sharing of information in a group, Conduction of meeting/ review, versioning.

Note:

1. This course is to be conducted in the **laboratory** by demonstrating various programming/industry practices.
2. **Adjunct faculty** from industry may be used to give industry relevance to the course coverage.
3. Students are expected to **write programs / do the course work** using knowledge gained and prepare **reports** based on the course work assigned by a course teacher.
4. Students are expected to give presentation on a chosen topic approved by the course teacher. **Internal evaluation** is to be based on the **continuous evaluation** of each activity of the **course work, report** preparation and **presentation**.
5. Final evaluation (SEE) is to be based on oral exams based on implementation.

Reference Books:

1. Brain W. Kernighan & Rob Pike, “The Practice of Programming”, Pearson education, 2008.
2. Knowledge repository created by various industries available on the internet

18UCSL604	Minor Project - 2	(0-0-4) 2

Contact Hours: 48

Course Learning Objectives (CLOs): Though the specific objectives of this course depend on the Project chosen, below are the generic objectives of this course:

Understand the domain, Analyze through Modeling and Implementation through state of the art technology available. To know Software Engineering Principles: Modeling, Estimation, Design standards and architectural issues through use of Standards etc. Also, write modular programs and handle exceptions to provide reliable solutions, to test and verify the programs for different scenarios.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Identify** and **formulate** the problem.	11,12	1,2,8	6,7
CO-2	**Analyze** the problem scenario and **Design** the solutions to complex engineering problem using software engineering principles or appropriate research methodology.	11,12, 13	2,3,5,8,16	6,7
CO-3	**Identify** and **Implement** a feasible solution using appropriate technology, tools, procedures and techniques.	11,12,14	3,4,5,8,16	-
CO-4	**Verify** and **Validate** the proposed system for correctness and to demonstrate compliance with the design and hence the stated requirements/ research gap.	11,12,15	5,8,16	-
CO-5	**Prepare** the report and **communicate** effectively through presentation.	10,11	8,9	-

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	2.0	2.0	2.0	2.0	2.0	1.0	1.0	2.0	2.0	3.0	3.0	3.0	3.0	3.0	3.0	2.0

Prerequisites: Knowledge of

- Software Engineering concepts
- Any Programming Language

Guidelines for the Conduction of Project Work:

A project may belong to any of the following categories:

1. Learning Centric: Here the output of the project activity is enhancement of the student's knowledge. Here the student chooses a work with the intension to gain the knowledge.

2. Application: Traditional Software Engineering project, with appreciable complexity. Scenario of demonstration had to be made clear and

Completeness is important.

3. **Product Base:** The final output is a novel one which may be the assembly of several existing subsystems. Completeness is important till the user manual.
4. **Research Oriented**: A research problem of student's interest. Achievement would be a publication in the IEEE/ACM international/national conferences.

General Instructions to Students:

- Students are expected to perform extensive literature survey, identify problem statements and prepare synopsis in consultation with project guide/supervisor. Students are expected to submit Synopsis- Initial (Registration Phase-1) approved by project guide, to the project coordinator as per the schedule notified. A copy is to be maintained with students and the guide. This registration/ Initial synopsis contains the description of the project concept created and acts as a base line for design and Implementation of the system.
- Notification/schedules and evaluation procedures will be sent to all students in the Google groups created in the department.
- Batch size is of maximum 4 students. Mixing of divisions is not permitted unless it is the sponsored/research project and request is initiated by project supervisor/guide.
- Students may approach the faculty members of CSE department for choosing them as project guide/supervisor.
- A teacher can guide/supervise maximum of 2 UG project groups. However in special case, a DUGC (Project Coordinating Team, **PCT**, working on the behalf of DUGC) may assign additional project groups to a given teacher.
- The department will financially support presentations of publication of paper only for reputed conference publications.
- DUGC/Project coordinating team (**PCT**) may be consulted for any other/ missing information.

Evaluation Procedures:

a) CIE Marks for the project is to be awarded by project guide/supervisor and SEE marks are to be awarded by examiners (appointed by DUGC) and PCT based on the overall guidelines and project classifications/rubrics by looking into Software Engineering aspects & usefulness w.r.t research/innovation/technology/industry trends through formal interactions and presentations. However, each team is expected to give a formal complete presentation of their work at the end of each phase (1 to 4) to project coordinating team (PCT).
b) PCT evaluates the work and suggest the corrections and observation. All project teams are expected to incorporate these changes in their work.

KURAKUNDI RAGHAVENDRA RAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 11:37:02 +05'30'

These observations will be made available to Project guides and SEE examiners, which will help them to evaluate and award marks during assessment process.

c) Project guides should keep track of all interactions they do with project team members on weekly basis.
d) All Projects are evaluated and individual students are awarded a grade based on the grading criteria set.
e) Individuals' grade/marks is decided based on both CIE and SEE marks/grade.
f) A project is considered for possible award of S grade; if and only if its 'research outputs' / 'product innovation outputs' results in to a publication of a paper. In a special case, innovative or sponsored business applications with focus on recent technological trends/ Industry trends catering for societal needs may be considered. If contents of paper/research output are not at satisfactory level, then, the assessment results in to appropriate lower grades.
g) A project is considered for possible award of A grade; if and only if it demonstrate product development skills in core system or systems level applications using all aspects of software engineering product development phases like: requirements, design, implementation, testing including standards like: use of design/architectural patterns, coding standards, use of tools for design/testing, programming practices, documentation and reporting etc..,. In a special case, innovative or sponsored business applications with focus on recent technological trends / Industry trends catering for societal needs may be considered.
h) If project work contribution/ output is not at satisfactory level, then the assessment results @ appropriate lower grades. All grades/marks are awarded based on individual contributions evaluated from software engineering perspectives **specified in the rubrics**.
i) Expected important features:
    - Report preparation using Latex.
    - Online plagiarism check report is to be enclosed in the report.
    - Use of IEEE standard. Ex: reference listing and use of PPTs for presentation etc…
    - Use of software tools. Ex: for Design, version control, UI design, Testing etc…
    - Conduction of workshop/ training on technology/domain to students and Preparation of training material/manual (.doc & .pdf).
    - Publication of paper based on outcome of the project.
    - Submission of proposal to KSCST (Govt. of Karnataka) or other agencies for funding.

- Any other features suggested by guides/coordinators from time to time.

j) Marks Weightage and Various parameters for project evaluation for both CIE and SEE level @ 6th Semester.

k)

Sl.No.	Parameter for Assessment	Marks (%)
1	Requirements Analysis (SRS): Abstract and Detailed.	20
2	Design Specification; Use of: UML diagrams, architecture diagram, ER diagram, Patterns etc… Proper cohesiveness and coupling of various components in the system design.	35
4	Use of Tools and standards.	5
5	Implementation: Code documentation, style, robustness, maintainability, Testability, Usability (User Experience) etc…	10
6	Testing: for every scenario of all use cases identified.	10
7	Final Oral Presentation (viva-voce) (IEEE Standards for slides, oral presentation techniques etc…)	10
8	Project Reports- Final and Intermediate if any: preparation using LATEX and plagiarism check	10

Note:

1. Sufficient and completeness of each parameter is to be seen while awarding marks for individual students.
2. Marks for individual students in a given project team may vary based on individuals 'learning outcomes.

- All project teams are expected to participate in the project exhibition arranged at department level. Project teams are expected to share their project experience to all their juniors and motivate them to take-up challenging work as their project work. During project exhibition, Top 2 projects from the batch will be awarded with a certificate of appreciation at the end of academic year.

KURAKUNDI RAGHAVENDRA RAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 11:42:12 +05'30'

18UHUL605	Soft Skills / Aptitude	(0-0-2) 1

Contact Hours: 24

Course Learning Objectives (CLOs):

This is included with the objectives of improving the communication skills, proficiency in English language and aptitude ability of the student to enhance the employability.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSO(13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the significance of communication in the profession.	-	10	-
CO-2	**Use** the English language with proficiency	-	10	12
CO-3	**Solve** Aptitude related problems	-	9	12
CO-4	**Demonstrate** the competency in the placement activities.	-	9	-

POs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	-	-	-	-	-	-	-	-	2.0	2.0	-	1.0	-	-	-	-

Contents:

Training on communication skills, proficiency in English language and aptitude ability involving the internal and external resource.

Evaluation:

Both the internal and external resource persons shall be engaged in imparting the related knowledge and shall have only CIE as the evaluation component. There shall be one test conducted at the end for 25 marks in Aptitude testing and there shall be one presentation by the student for 25 marks or any other suitable testing components. The arrangement for CIE evaluation is to be done by the department and maintain the relevant documents.

KURAKUNDI RAGHAVENDRA RAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 11:45:25 +05'30'

18UCSE606	UNIX Systems Programming	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course facilitates the students to get familiarity with system calls, UNIX kernel structure and use of standards like ANSI and POSIX in programming.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Describe** the ANSI and POSIX standards used in UNIX operating system and programming	-	1	13
CO-2	**Explain** the UNIX file types and **demonstrate** the use of UNIX file APIs in programming.	13,14	1,2,3	15
CO-3	**Identify** the process management activities of UNIX and **write** programs that make use of processes and their environment.	13,14	1,2,3	15
CO-4	**Describe** the use of signals in UNIX and **illustrate** the use of signals in programs.	13,14	1,2,3	15
CO-5	**Explain** the need of daemons in UNIX and **identify** the use of daemons in UNIX OS.	13,14	1,2,3	15
CO-6	**Explain** inter process communication mechanisms of UNIX and **write** programs to demonstrate IPCs for client-server interactions.	13,14	1,2,3	15
CO-7	**Describe** the ANSI and POSIX standards used in UNIX operating system and programming.	-	1	13

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	2.0	2.0	2.0	-	-	-	-	-	-	-	-	-	2.4	3.0	1.0	-

KURAKUNDI RAGHAVENDRAR AO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 11:46:24 +05'30'

Pre-requisites: Knowledge of

- UNIX operating system and its commands
- Operating Systems Fundamentals
- Basics of Networking

Contents:

Unit-I

ANSI and POSIX Standards: UNIX and ANSI Standards – ANSI C standard, POSIX standards, POSIX environment, POSIX feature test macros, limits checking at compile time and run time; UNIX and POSIX APIs – POSIX APIs, UNIX and POSIX development environment, API common characteristics

UNIX Files: file types, UNIX and POSIX file systems, UNIX and POSIX file attributes, inodes in UNIX system V, UNIX kernel support for files, directory files, hard and symbolic links. **7 Hrs**

Unit-II

UNIX File APIs: General file APIs, open, read, write, close, fcntl, lseek, link, unlink, stat, fstat, lstat, access, chmod, fchmod, chown, fchown, lchown, utime, file and record locking, directory file APIs, device file APIs, FIFO file APIs, symbolic link file APIs. **7 Hrs**

Unit-III

Environment of a UNIX Process: Introduction, main function, process termination, command line arguments, environment list, memory layout of a C program, alloca function, environment variables, setjmp and lonjmp functions, getrlimit and setrlimit functions

Process Control: Introduction, process identifiers, fork function, vfork function, exit functions, wait and waitpid functions, race conditions, exec functions, changing user IDs and group IDs, system function. **9 Hrs**

Unit-IV

Process Relationships: Introduction, terminal logins, network logins, process groups, sessions, controlling terminal, job control

Signals and Daemon Processes: Signals – UNIX kernel support for signals, signal, signal mask, sigaction, sigsetjmp and siglongjmp APIs, kill, alarm, interval timers; Daemon Processes – introduction, daemon characteristics, coding rules, error logging, client-server model. **9 Hrs**

Unit-V

Interprocess Communication: Introduction, pipes, message queues, UNIX APIs for message queues, client-server example for message queue, sockets, socket APIs, client-server example for socket. **7 Hrs**

Reference Books:

1) Terrence Chan, “UNIX System programming using C++”,Prentice Hall India, 2015

2) W. Richard Stevens, ”Advanced Programming in the UNIX environment”, Pearson Education/ PHI, 2005

3) Kay A Robbins & Steven Robbins, “Unix Systems Programming: Communication, Concurrency, and Threads”, Prentice Hall Publications, 2003.

18UCSE607 Digital Image Processing (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning objectives:

- To learn fundamental theories and techniques of digital image processing.
- To acquire the skill necessary to explore advanced topics of digital image processing.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Describe** the principles of Digital Image Processing.	-	1,2	3,5,13,14
CO-2	**Demonstrate** the image enhancement techniques that include primitives image sensing and acquisition techniques, image formation, image representation & relationship between the pixels.	13,14	1,2,3	15
CO-3	**Explain** the basic principles of mathematical morphology & **write** program to **extract** the characteristic features of image using morphological operations.	13,14	1,2,3	15
CO-4	**Apply** segmentation techniques for a given application scenario.	13,14	1,2,3	15
CO-5	**Explain** and **implement** the core princples of image representation	13,14	1,2,3	15

	techniques.			

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.0	1.8	-	1.0	-	-	-	-	-	-	-	2.6	2.6	1.0	-

Pre-requisites: Knowledge of

- Basics of Statistics
- Linear Algebra

Contents:

Unit-I

Digital Image Fundamentals - Introduction, Applications, Fundamental Steps in Digital Image Processing, Elements of visual perception, Image sensing and acquisition, Image Sampling and Quantization, Basic relationships between pixels. **8 Hrs**

Unit-II

Intensity Transformations and Spatial Filtering - Basic Intensity Transformation Functions, Histogram Processing; Fundamentals of Spatial Filtering., Smoothing and Sharpening Spatial filters. **8 Hrs**

Unit-III

Morphological Image Processing- Erosion and Dilation, Opening and Closing, Hit or Miss Transforms, Basic Morphological Algorithms, GrayScale Morphology. **8 Hrs**

Unit-IV

Image Segmentation- Point, Line, and Edge Detection, Thresholding, Region-Based Segmentation, Segmentation Using Morphological Watersheds. **8 Hrs**

Unit-V

Representation and Description- Image Representation, Boundary and Regional Descriptors **7 Hrs**

Reference Books:

1. Rafel C Gonzalez & Richard E Woods, “Digital Image Processing”, 3rd Edition, Pearson Education, 2016.
2. Milan Sonka, Vaclav Hlavac & Roger Boyle, “Image Processing, Analysis and Machine Vision”, 2nd Edition, Thomoson Learning, 2001.
3. Anil K Jain, “Fundamentals of Digital Image Processing”, Prentice-Hall of India Pvt. Ltd., 1997.
4. B.Chanda & Dutta Majumder, “Digital Image Processing and Analysis”, Prentice-Hall, India, 2002.

18UCSE608	Principles of Programming	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning objectives:

- Explain the value of declaration models, especially with respect to programming-in-the-large.
- Identify and describe the properties of a variable such as its associated address, value, scope, persistence, and size.
- Demonstrate different forms of binding, visibility, scoping, and lifetime management.
- Justify the philosophy of object-oriented design and the concepts of encapsulation, abstraction, inheritance, and polymorphism.
- Evaluate the trade offs between the different paradigms, considering such issues as space efficiency, time efficiency (of both the computer and the programmer), safety, and power of expression.
- Design, code, test, and debug programs using the functional paradigm.
- Outline the strengths and weaknesses of the logic programming paradigm.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the notations to describe syntax and semantics of programming languages.	-	-	1
CO-2	**Explain** the principles of imperative programming languages such as binding, scope, control structures, subprograms and parameter passing methods and **write** simple programs to demonstrate these for a given application.	13,14	1,2,3	15
CO-3	**Use** principles of object oriented programming features in writing the programs for the given problem scenario.	13,14	1,2,3	15
CO-4	**Write** the formal syntax for a specification of functional	13,14	1,2,3	15

KURAKUNDI RAGHAVENDRARAO GOPINATH

Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 12:51:42 +05'30'

	programming languages.			
CO-5	**Write** programs in the Prolog language for given problem scenario.	13,14	1,2,3	15

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.8	2.0	2.0	-	-	-	-	-	-	-	-	-	3.0	3.0	1.0	-

Pre-requisites: Knowledge of

- Programming language
- Mathematics

Contents:

Unit-I

Introduction: Toward higher-level languages, Programming paradigms, Language implementation: Bridging the gap, Expression notations, and Abstract syntax trees.

Types: Data Representation: Elementary data types – Data objects, Variables and Constants, Data types, Declarations, Type checking and type conversion. Numeric data types, Enumerations, Booleans, Characters. Structured data types - Structured data objects and data types, Specification of data structure types, Implementation of data structure types, Vectors and arrays, Records. **8 Hrs**

Unit-II

Imperative Programming: Basic Statements, structured sequence control, handling special cases in loops, programming with variants, proof rules for partial correctness. Procedure activations – Simple call-return subprograms, Parameter passing methods, Scope rules for names, Nested scopes in the source text, Activation records, Lexical scope: Procedures as in C. **8 Hrs**

Unit-III

Object-Oriented Programming: Object-oriented design, Encapsulation and information-hiding, Separation of behavior and implementation, Classes and subclasses, Inheritance (overriding, dynamic dispatch), Polymorphism (subtype polymorphism vs. inheritance). **7 Hrs**

Unit-IV

Functional Programming: Lamda Calculus, Elements of functional programming – A Little language of expressions, Types: values and operations, Approaches to expression evaluation, Lexical scope, Type checking. Functional programming in a typed language -Exploring a list, Function declaration by cases, Function as first-class values, ML: Implicit types, Data types, Exception handling in ML. Functional programming with lists - Scheme, a dialect of lisp, The structure of lists, List manipulation. **9 Hrs**

Unit-V

Logic Programming: Predicate Logic: FOL, Computing with relations, Introduction to prolog, Data structures in prolog, Programming techniques, Control in prolog. **7 Hrs**

Reference Books:

1. Ravi Sethi, Programming languages, concepts & constructs Addison Wesley 2/E, 2009
2. Terrence W.Pratt, Programming languages Design and Implementation Pearson Education, 4/E, 2009.
3. Robert W Sebesta, Concept of Programming language Pearson Education, 11/E, Pearson Education, 2019

18UCSE609	Data Mining	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning objectives:

- Compare and contrast different conceptions of data mining as evidenced in both research and application.
- Explain the role of finding associations in commercial market basket data.
- Characterize the kinds of patterns that can be discovered by association rule mining.
- Describe how to extend a relational system to find patterns using association rules.
- Evaluate methodological issues underlying the effective application of data mining.
- Identify and characterize sources of noise, redundancy, and outliers in presented data.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Analyze** the given scenario and **perform** data pre-processing using appropriate methods.	13,14	1,2,3	15
CO-2	**Predict** group membership for data	13,14	1,2,3	15

	instances using Hunt's algorithm, Rule based and Nearest Neighbor classification techniques**.**			
CO-3	**Analyse** the data using Apriori and Non-Apriori based algorithms in order to determine **patterns** from the large data sets.	13,14	1,2,3	15
CO-4	**Explain** and **apply** the partitional and heirarchical clustering techniques for a given data set.	13,14	1,2,3	15
CO-5	**Illustrate** the statistical, proximity based and clustering based techniques to detect the outliers in the data set.	13,14	1,2,3	15

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.8	2.0	2.0	-	-	-	-	-	-	-	-	-	3.0	3.0	1.0	-

Pre-requisites: Knowledge of Statistics at Introductory level

Contents:

Unit-I

Introduction and Basic Concepts: What is Data Mining? Motivating Challenges, The Origins of Data Mining, Data Mining Tasks.

Data: Types of Data, Data Quality, Data pre-processing, Measures of Similarity and Dissimilarity. **8 Hrs**

Unit-II

Classification: Preliminaries, General approach to solving a classification problem, Decision tree induction, Model over fitting, Evaluation of the performance of a classifier, Rule based Classification, Nearest Neighbour classifiers. **8 Hrs**

Unit-III

Association Analysis: Problem Definition, Frequent Itemset Generation, Rule Generation, Compact Representation of Frequent Itemsets, FP tree Growth algorithm, Evaluation of Association Patterns. **8 Hrs**

Unit-IV

Cluster Analysis: Overview, K-means algorithm, Agglomerative Hierarchical Clustering, DBSCAN, Prototype-based Clustering. **8 Hrs**

Unit-V

Anamoly Detection: Preliminaries, Statistical approaches – Parametric and Non Parametric models, Proximity based approaches – Distance and Density based techniques, Clustering based techniques. **7 Hrs**

Reference Books:

1. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, "Introduction to Data Mining", 3rd Edition, Pearson Education, 2014.
2. Jiawei Han &MichelineKamber, "Data Mining – Concepts and Techniques", 3rd Edition, Morgan Kaufmann Publishers, 2011.
3. Arun K Pujari, "Data Mining Techniques", 3rd Edition, Universities Press, 2013.
4. K. P. Soman, S. Diwakar & V. Ajay, "Insight into Data Mining", Prentice Hall India, 2008.

18UCSE610 Advanced Data Structures and Algorithms (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- Asymptotic and Amortized Analyses
- Linear sorting algorithms
- Advanced data structures such as Heaps, B-trees, Red-Black trees etc.
- String matching algorithms

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Analyze** the performance of the given algorithm using asymptotic notations and amortized techniques.	-	1,2	-
CO-2	**Explain** the working and assumptions of linear sorting methods and **apply** them to solve a given problem.	-	2,13	1
CO-3	**Compare** the working of string matching algorithms and **use** them appropriately in developing applications.	-	2,13	1
CO-4	**Build** and **perform** the operations	-	2,13	1

	on heap structures.			
CO-5	**Build** and **perform** the operations on search structures.	-	2,13	1
CO-6	**Use** the hash tables for the implementation of dictionary operations.	-	2,13	1
CO-7	**Choose** the appropriate data structure and **use** relevant algorithms to solve problems in different domains including project management.	-	2,13	1,11

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.1	2.0	-	-	-	-	-	-	-	-	1.0	-	2.0	-	-	-

Pre-requisites: Knowledge of

- Programming language (any)
- Data Structures
- Algorithms

Contents:

Unit-I

Introduction to Complexity Analysis: Review of Asymptotic notations and their properties, Amortized analysis – Aggregate, Accounting and Potential methods.
Linear Sorts: Counting Sort, Bucket Sort, and Radix Sorting with Analysis for all algorithms. **8 Hrs**

Unit-II

ring Matching: Naive algorithm; Rabin-Karp algorithm; String matching with Finite automata, KMP algorithm, Boyer-Moore algorithm. **8 Hrs**

Unit-III

Heap Structures: Binomial heaps, Fibonacci heaps.
Search Structures: 2-3 trees, 2-3-4 trees, B-trees, B^+ trees, Red-black trees. **8 Hrs**

Unit-IV

ashing: Direct Address Tables, Hash Tables, Collision Resolution by Chaining – Analysis, Hash Functions – Properties, Division and Multiplication methods, Universal Hashing, Open Addressing – Linear and Quadratic Probing, Double hashing. **8 Hrs**

Unit-V

KURAKUNDI RAGHAVENDRA RAO GOPINATH
Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.06 14:30:50 +05'30'

Applications: Huffman coding, Garbage collection and compaction, Min-Cut Max-Flow algorithm, Activity networks. **7 Hrs**

Reference Books:

1. Thomas H.Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein- "Introduction to Algorithms", 3rd Edition, Prentice Hall India, 2009.
2. E. Horowitz, S.Sahni and Dinesh Mehta- "Fundamentals of Data structures in C++", Galgotia, 2006.
3. Anany Levitin, "Introduction to the Design and analysis of algorithms", 3rd Edition, Pearson Education, 2011

18UCSC700 Artificial Intelligence and Machine Learning (4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- To introduce the basic concepts, theories and state-of-the-art techniques of artificial intelligence and machine learning.
- Enable student with knowledge enough to be a self-learner in exploring the application of machine learning /AI algorithms in the different fields of science, medicine, finance etc.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the world, behavior of agents and problem-solving aspects of agents.	-	3,4	1,2
CO-2	**Demonstrate** the representation and usage of knowledge using First order logic.	-	3,4	1,2
CO-3	**Explain** the decision-making process with incomplete, inconsistent and everchanging facts.	-	3,4	-
CO-4	**Explain** machine learning concepts and range of problems that can be handled by machine learning.	-	3,4	-
CO-5	**Apply** the concepts of and the machine learning to the real-world problems.	-	-	3,4,5

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.0	1.0	2.0	2.0	2.0	-	-	-	-	-	-	-	-	-	-	-

Pre-requisites: Knowledge of

1. Logic
2. Discrete Mathematics
3. Programming Fundamentals

Contents:

Unit-I

Introduction: AI Problems Underlying Assumption, AI Techniques, Criteria for Success

ate Space Search & Heuristic Search Techniques: Defining the Problems as A State Space Search, Production Systems, Production Characteristics, Issues in The Design Of Search Programs.

Generate And-Test: Hill Climbing, Best-First Search, Problem Reduction, Constraint Satisfaction, and Means-ends Analysis. **11 Hrs**

Unit-II

nowledge Representation: Issues, Representations and Mappings, Approaches to Knowledge Representation.

Using Predicate Logic: Representation Simple Facts in Logic, Representing, Computable Functions and Predicates, Resolution. Representing Knowledge Using Rules, Procedural Versus Declarative Knowledge, Logic Programming, Forward Versus Backward Reasoning. **11 Hrs**

Unit-III

atistical Reasoning, Probability and Uncertainty: Bayes' Theorem, Full joint probabilistic distribution, Bayesian Networks and Probabilistic inferences, Dumpster Shafer Theory.

Forms of learning: Issues in designing a learning system. Assumptions of Common Machine Learning Models. **10 Hrs**

Unit-IV

near Regression: Multivariate Regression, Logistic regression, Polynomial Regression.

Linear Models for Classification: Decision Trees, Regression Trees, K-nearest neighbors (KNN) algorithm. Bias Variance Trade off. **10 Hrs**

Unit-V

Perceptron: Neural Networks - Introduction, Early Models, Perceptron Learning, Backpropagation, Initialization, Training & Validation, Support Vector Machines, Introduction, Early Models, Unsupervised learning and clustering – k-means clustering, hierarchical clustering. **10 Hrs**

Reference Books:

1) Elaine Rich and Kevin Knight "Artificial Intelligence", 2nd Edition, Tata Mcgraw-Hill, 2005.
2) Stuart Russel and Peter Norvig, "Artificial Intelligence: A Modern Approach", 3rd Edition, Prentice Hall, 2009.

3) Trevor Hastie, Robert Tibshirani, and Jerome H. Friedman “The Elements of Statistical Learning”.
4) Christopher Bishop, “Pattern Recognition and Machine Learning”
5) Mitchell Tom “Machine Learning”, McGraw Hill, 1997.

18UCSL702 Artificial Intelligence and Machine Learning Laboratory (0-0-2) 1

Contact Hours: 26

Course Learning Objectives (CLOs): This course is in line with the theory course Artificial Intelligence & Machine Learning (18UCSC700). It focuses on hands on experience on creation of data models, database design, programming using appropriate technology.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course, the student will be able to:		**Mapping to POs(1-12) / PSOs(13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Represent** the Knowledge for the given scenario using appropriate tools techniques and language.	-	3,4,5	-
CO-2	**Identify** data preprocessing requirement of a given data set for the learning algorithms.	-	3,4,5	-
CO-3	**Demonstrate** of the strengths and weaknesses of regression and classification approaches in machine learning.	-	3,4,5	-
CO-4	**Demonstrate** unsupervised algorithms for clustering requirement on a data set from the real world using python.	-	3,4,5	-
CO-5	**Represent** the Knowledge for the given scenario using appropriate tools techniques and language.	-	3,4,5	-

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	-	-	2.0	2.0	2.0	-	-	-	-	-	-	-	-	-	-	-

Pre-requisites: Knowledge of

1. Logic
2. Discrete Mathematics
3. Programming Fundamentals

Course Contents

This course is in line with the theory course Artificial Intelligence & Machine Learning (18UCSC700). There will be one problem with the **Knowledge Representation** and four problems on **Machine Learning** (Linear Regression, Multilinear regression, Polynomial regression, Decision Tree, K-means clustering). The problems to be composed by the faculty are announced to the students. The student will analyze the problem, justify the requirement of AI Approach for the solution, choose the platform or technology for implementations and demonstrate all the steps involved like pre-processing the dataset, importing the dataset, Spitting the dataset into the training set and test set, training the model on the training dataset, predicting the test set results, Visualising the Training set results, Visualising the test set results, etc where ever applicable. The students will submit implementation, conduction and observation write up for each problem. An internal examination and 5 problems work will be used to grade the student's performance in this course.

Associated Lab Works (Sample)

1. Represent facts and relationships of any famous epic of your choice using first order logic, implement and demonstrate some queries.
2. Build a decision tree for the case of SDMCET students' performance based on the IA-1, IA-2, IA-3, CTA, Attendance, SEE marks (optional) and classifying them into one of the Grade S, A, B, C, D, E & F. Study of precision of classification by including the 10th, 12th and CET/COMED-K into consideration.
3. Given the features of an email like , Sender's email ID, Number of typos in the email, Occurrence of words like "offer", "prize", "free Gift", classify the email as Spam or not. Use the feature vector to train a Logistic classifier which emits a score in the range 0 to 1. If the score is more than 0.5, we label the email as spam. Otherwise, we don't label it as spam.
   (https://magoosh.com/).
4. Linear or polynomial regression to predict the salary of a person given the designation, no of years of experience, location of work, previous financial years profit etc.
5. A Hospital Care chain wants to open a series of Emergency-Care wards within a region. We assume that the hospital knows the location of all the

maximum accident-prone areas in the region. They have to decide the number of the Emergency Units to be opened and the location of these Emergency Units, so that all the accident-prone areas are covered in the vicinity of these Emergency Units. The challenge is to decide the location of these Emergency Units so that the whole region is covered. Here is when K-means Clustering comes to rescue! (https://www.edureka.co/blog/k-means-clustering/)

18UCSL703 Major Project – Phase 1 (0-0-4) 2

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:
Understand the domain through proper modeling and analysis using the state-of-art technology. Then apply relevant Software Engineering Principles to develop modular and robust applications through the use of Standards and tools. At the end build appropriate test cases, verification and validation techniques in order to make the project reliable and maintainable.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Identify** the problem and **formulate** the problem statement.	1, 8, 2,12	-	-
CO-2	**Analyze** the problem scenario and **Design** the solutions to complex engineering problem using software engineering principles or appropriate research methodology.	3,5, 8, 13,12	11, 16	6, 7
CO-3	**Identify** and **Implement** a feasible solution using appropriate technology, tools, procedures and techniques.	4,5, 8, 14,12	16	-
CO-4	**Verify** and **Validate** the proposed system for correctness and to demonstrate compliance with the design and hence the stated requirements/ research gap.	5, 8, 15,12	-	-
CO-5	**Prepare** the report and **communicate** effectively through	8, 9,10	-	-

	presentation.			

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	3.0	3.0	3.0	3.0	3.0	1.0	1.0	3.0	3.0	3.0	2.0	3.0	3.0	3.0	3.0	2.0

Prerequisites: Different programming languages / tools, Software Engineering Principles

Marks Weightage and Various parameters for project evaluation for both CIE and SEE level @ 7th and 8th Semester:

Sl.No.	Parameter for Assessment	% Weight For CIE and SEE	
		7th Sem	8th Sem
1	**Requirements Analysis (SRS):** Abstract and Detailed.	20	10
2	**Design Specification:** Use of UML diagrams, architecture diagram, ER diagram, Patterns etc. Proper cohesiveness and coupling of various components in the system design.	35	10
4	Use of Tools and standards.	5	5
5	**Implementation:** Code documentation, style, robustness, maintainability, Testability, Usability (User Experience) etc.	10	20
6	**Testing:** for every scenario of all use cases identified.	10	10
7	**Final Oral Presentation** (viva-voce) (IEEE Standards for slides, oral presentation techniques, etc.)	10	10
8	**Project Reports** - Final and Intermediate if any: preparation using LATEX and plagiarism check	10	20
9	**Preparation of Learning Materials** (.doc & .pdf) & **Videos**: Uploading through Department Channel on YouTube. **Knowledge Transfer**: through conduction of workshop and training programs	-	10
10	**Publication of paper / Equivalent Effort** (@IEEE or equivalent or higher)	-	5

References:

1. Grady Booch, “Object-Oriented Analysis and Design with Applications”, Second Edition, Addison-Wesley Publications.
2. Pankaj Jalote, “An Integrated Approach to Software Engineering”, Third Edition, Springer Publications.
3. Project Manual, prepared by the CSE Department, S D M College of Engineering and Technology, Dharwad

18UCSL704 **Internship** **2 Credits**

Contact Hours: 4 weeks

Course Learning Objectives (CLOs): Internship provides an opportunity to get industry exposure to real time scenarios that include professional skill development programs and adhere to the professional standards.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explore** the domain knowledge	1, 8, 2,12	-	-
CO-2	**Apply** the knowledge and skills in the professional career.	3,5, 8, 13,12	11, 16	6, 7
CO-3	**Prepare** a technical report	4,5, 8, 14,12	16	-
CO-4	**Demonstrate** the knowledge gained through presentation.	5, 8, 15,12	-	-

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	3.0	3.0	3.0	3.0	3.0	1.0	1.0	3.0	3.0	3.0	2.0	3.0	3.0	3.0	3.0	2.0

Prerequisites: Knowledge of both theory and practical courses learnt in all the previos semesters and relevant value-added information.

18UCSE705	Computer Graphics	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course introduces fundamental principles of computer graphics, its architecture and how transformations of objects are carried out. It facilitates students to identify good design principles to solve challenges involved in simulating real world objects/conditions. It also provides the students to learn and apply the aspects of interaction with computer and exposes them to open-source tools like OpenGL.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs(13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the principles of Computer Graphics Architecture used in industry relevant tool like OpenGL.	-	13,14	1,2,12
CO-2	**Explain** the design objectives of APIs used in OpenGL.	-	14,12	1,5
CO-3	**Apply** input interaction techniquesused in graphics environment.	14	2	-
CO-4	**Apply** affine transformations to solve problems relating to object transformations.	13	14	1
CO-5	**Discriminate** the views of objects in parallel and perspective projections under various lighting conditions.	-	13	14
CO-6	**Formulate** mathematical strategies for scan conversion algorithms to realize basic primitives, and **represent** curve and surfaces.	13,14	-	1,12

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.0	1.5	-	-	1.0	-	-	-	-	-	-	1.3	2.5	2.1	-	-

Pre-requisites: Knowledge of

- Basic Engineering Graphics
- Linear Algebra (Scalars, Vectors, Matrices)

KURAKUNDI RAGHAVENDRA RAO GOPINATH
Digitally signed by KURAKUNDI RAGHAVENDRARAO GOPINATH
Date: 2023.02.09 14:39:05 +05'30'

- Algorithms and C programming.

Contents:

Unit-I

Introduction : Applications of computer graphics; A graphics system; Images: Physical and synthetic; Imaging systems; The synthetic camera model; The programmer's interface; Graphics architectures; Programmable pipelines; Performance characteristics. Graphics Programming: The Sierpinski gasket; Programming two dimensional applications

Input and Interaction: Interaction; Input devices; Clients and servers; Display lists; Display lists and modeling; Programming event-driven input; Menus; Picking; A simple CAD program; Building interactive models; Animating interactive programs; Design of interactive programs; Logic operations. **9 Hrs**

Unit-II

Geometric Objects and Transformations: Scalars, points, and vectors; Coordinate systems and frames; Modeling a colored cube; Affine transformations; Rotation, translation and scaling; Transformations in homogeneous coordinates; Concatenation of transformations; Interfaces to three-dimensional applications. **9 Hrs**

Unit-III

Viewing and Lighting : Classical and computer viewing; Viewing with a computer; Positioning of the camera; Simple projections; Projections in OpenGL, Hidden surface removal; Parallel-projection matrices; Perspective-projection matrices.

OpenGL: Introduction to OpenGL; Programming two-dimensional Application; The OpenGL API; Primitives and Attributes; Color; Control Functions. **7 Hrs**

Unit-IV

Basic Raster Graphics Algorithms for drawing 2D primitives : Scan converting lines, circles, Filling Rectangles, Polygons; Clipping in a raster world; Clipping lines, polygons; Anti-aliasing **7 Hrs**

Unit-V

Lighting and Shading: Light and matter; Light sources; The Phong lighting model.

Representing Curves and Surfaces: Parametric Cubic Curves – Hermite Curves, Bézier Curves. **7 Hrs**

Conduction of Practical Sessions: Practical Sessions to be held with the focus of learning Open Source Tools like OpenGL and its API features. For the successful completion of the course, students are expected to undertake project to explore advanced features of Open Source Tools like OpenGL.

Reference Books:

1) Edward Angel, "Interactive Computer Graphics A Top-Down Approach with OpenGL", 5th Edition, Addison-Wesley, 2008
2) James D Foley, Andries Van Dam, Steven K Feiner & John F Hughes, "Computer Graphics Principles and Practice", 2nd Edition, Addison-Wesley, 1997.
3) Edward Angel & Dave Shreiner, "Interactive Computer Graphics A Top-Down Approach with Shader-Based OpenGL", 6th Edition, Addison-Wesley, 2012
4) F.S. Hill, Jr., "Computer Graphics Using OpenGL", 2nd Edition, Pearson Education, 2005
5) Donald Hearn and Pauline Baker, "Computer Graphics- OpenGL Version", 2nd Edition, Pearson Education, 2003

18UCSE706 Software Testing (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course discusses the notations and terminologies used in software testing with Principles of programming, Software Development Process from the perspective of quality, maintenance, testing, programming style. Further, it focuses on System quality through: requirements, design, coding, verification and validation procedures and applications, case studies, use of tools / programming techniques for design and implementation.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Map** inputs to the expected outputs of the system by understanding the system behavior represented in the form of: requirements and design specifications / Models.	1, 13	-	-
CO-2	**Design** test cases based on requirement specifications.	2, 14	1	-
CO-3	**Design** test cases based on traditional design specifications.	-	3, 13	-
CO-4	**Design** test cases based on Object specifications/Models (UML).	13	3	-
CO-5	**Write** script in any programming Language/Tools to implement test	14	15	-

	cases designed.			
CO-6	**Design** test cases based on various testing strategies to check the correctness of computer program.	-	15, 13	-

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.5	3.0	2.0	-	-	-	-	-	-	-	-	-	2.5	3.0	2.0	-

Pre-requisites: Knowledge of

- Programming language
- Software Engineering
- Object Oriented System Analysis and Design

Contents:

Unit-I

Programming Practice: A Testing Perspective: Programming Principles & Guidelines, Coding Processes, Refactoring, Verifications & Metrics, Style- Names, Expression and Statements, Consistency and Idioms, Function Macros, Debugging, Performance, Portability

Basics of Software Testing: Human Errors and Testing, Software Quality, Requirements, Behavior and Correctness, Correctness versus Reliability, Testing and Debugging, Test Metrics, Software and Hardware Testing, Testing and Verification, Defect Management, Execution History, Test generation Strategies, Static Testing. Model-Based Testing and Model Checking, Control-Flow Graph, Types of Testing, The Saturation Effect. **9 Hrs**

Unit-II

Test Generation from Requirements: Introduction; The Test-Selection Problem; Equivalence Partitioning; Boundary Value Analysis, Category-Partition Method, Cause-Effect Graphing, Test Generation from Predicates. **7 Hrs**

Unit-III

Dependence, Data Flow Models and Data Flow Testing: Definition-Use pairs; Data flow analysis; Classic analysis; From execution to conservative flow analysis; Data flow analysis with arrays and pointers; Inter-procedural analysis; Overview of data flow testing; Definition- Use associations; Data flow testing criteria; Data flow coverage with complex structures; The infeasibility problem. **7 Hrs**

Unit-IV

Structural Testing: Overview; Statement testing; Branch testing; Condition testing,

Path testing; Procedure call testing; Comparing structural testing criteria; The infeasibility problem.

Test Case Selection and Adequacy Test Execution: Overview; Test specification and cases; Adequacy criteria; Comparing criteria; Overview of test execution; From test case specification to test cases; Scaffolding; Generic versus specific scaffolding; Test oracles; Self-checks as oracles; Capture and replay **9 Hrs**

Unit-V

Testing Object Oriented Software: Issues in Testing OO Software, Intra Class Testing, Testing with State Machine Models, Inter-Class Testing, Structural Testing of Class **7 Hrs**

Reference Books:

1) Brain W. Kernighan & Rob Pike, "The Practice of Programming", Pearson education, 2008.
2) Pankaj Jalote & Narosa, "An Integrated Approach to Software Engineering", 3rd Edition, Publishing House.
3) Edward Kit, "Software Testing in the Real World", Pearson Education, 2006
4) Aditya P Mathur, "Foundations of Software Testing", Pearson Education, 2008.
5) Mauro Pezze, Michal Young, John Wiley & Sons, "Software Testing and Analysis: Process, Principles and Techniques", 2008.

18UCSE707	**Web Technology**	**(3-0-0) 3**

Contact Hours: 39

Course Learning Objectives (CLOs): This course emphasizes on practice-based learning. It enables the students to understand the web application architecture and use the state-of-the-art technology to provide web-based solutions.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs(13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Design** and **develop** a static web pages using XHTML.	5	2,3,13,14,15	12
CO-2	**Design** and **develop** dynamic web application to validate and store data using markup languages like-	5	2,3,13,14,15	12

	XML, DTD & XSD.			
CO-3	**Design** and **develop** an interactive web application using JavaScript and XHTML with CSS.	5	2,3,13,14,15	12
CO-4	**Design** and **develop** dynamic web application using server-side programming and Database connectivity.	5	2,3,13,14,15	12
CO-5	**Develop** a web service to represent the data in the standard formats for the given requirements.	5	2,3,13,14,15	12
CO-6	**Explain** the future of World Wide Web and its associated trending technologies.	-	5, 13	1,12

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.0	2.0	2.0	-	3.0	-	-	-	-	-	-	1.0	2.0	2.0	2.0	-

Pre-requisites: Knowledge of

- Programming language (any)
- Database Management Systems

Contents:

Unit-I

Introduction to Web: WWW1.0, HTML, HTML5, XHTML, XML, XSD, DTD, DOM-XML. **8 Hrs**

Unit-II

Introduction to CSS: What is CSS, CSS Syntax, Location of Styles, Selectors, The Cascade: How Styles Interact, The Box Model, CSS Text Styling.

Advanced CSS: Layout, Normal Flow, Positioning Elements, Floating Elements, Constructing Multicolumn Layouts, Approaches to CSS Layout, Responsive Design, CSS Frameworks. **8 Hrs**

Unit-III

Introduction to JavaScript: Basics, Strings, Arrays, Functions, Objects in JavaScript, building simple applications using JavaScript and HTML. **8 Hrs**

Unit-IV

Introduction to PHP: Introduction to PHP, Datatypes, Control Statements, Loops, Functions, Embedding PHP in HTML & MySQL. **8 Hrs**

Unit-V

Introduction to Web 2.0: Overview of WWW 2.0, JSON, Web Services - SOAP & WSDL, RESTful.

Introduction to AJAX: Basics of AJAX, Asynchronous and Synchronous message transformation.

Future of Web: Overview of Semantic Web, Applications of Semantic Web, Virtual Reality, Web OS. **7 Hrs**

Reference Books:

1) Robert W. Sebesta, Programming the World Wide Web, 7/E Pearson Education, 2012.
2) Luke Welling, Laura Thomson, PHP and MySQL Web Development, 5th Edition, Pearson Education, 2016.
3) Nicholas C Zakas, Professional JavaScript for Web Developers, 3rd Edition, Wrox/Wiley India, 2012.
4) Nicholas C Zakas et al, Professional AJAX, Wrox, 2007.
5) Karin K. Breitman, Marco Antonio Casanova and Walter Truszkowski, Semantic Web: Concepts, Technologies and Applications, Springer International Edition, 2007.

18UCSE708 Adhoc Networks (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course covers major aspects of wireless ad hoc networks, from design through performance issues to application requirements. It starts with characteristics features, applications of wireless ad hoc networks, different types of routing protocols and understanding the security issues and various QoS requirements.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the design issues of protocol stack.	-	3,13	-
CO-2	**Compare** the architecture and working of various MAC layer protocols.	-	16	1,14
CO-3	**Apply** the knowledge of different routing mechanisms for the better	3,13,16	-	-

	routing decisions in WANETs.			
CO-4	**Apply** the knowledge of different transport layer protocols for ensuring reliable communication in WANETs.	3,13	16	15
CO-5	**Identify** the challenges in security and QoS issues in WANETs and explain suitable solutions for the same.	-	2	1,16

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.0	2.0	2.7	-	-	-	-	-	-	-	-	-	2.7	1.0	1.0	1.7

Pre-requisites: Knowledge of

- Data Communications
- Computer Networks
- C programming

Contents:

Unit-I

Adhoc Networks: Introduction, Issues in Adhoc wireless networks, Adhoc wireless internet.

MAC–1: MAC Protocols - Introduction, Issues in designing a MAC protocol, Design goals of a MAC protocol for Adhoc wireless networks. **7 Hrs**

Unit-II

MAC–2: Classification of MAC protocols, Contention based protocols with reservation mechanisms; Contention based MAC protocols with scheduling mechanism, MAC protocols that use directional antennas, Other MAC protocols. **7 Hrs**

Unit-III

Routing: Proactive and Reactive Routing Protocols, Hybrid routing protocol, Routing protocols with effective flooding mechanisms, Hierarchical routing protocols, Power aware routing protocols. **9 Hrs**

Unit-IV

Transport Layer: Protocols - Introduction, Issues in designing a transport layer protocol, Design goals of a transport layer protocol, Classification of transport layer solutions, TCP over Ad hoc wireless Networks, Other transport layer protocols. **9 Hrs**

Unit-V

Security: Introduction, Network security requirements, Issues & challenges in security provisioning, Network security attacks, Key management, Secure routing.
QoS: Introduction, Issues and challenges in providing QoS, Classification of QoS solutions, MAC layer solutions, network layer solutions. **7 Hrs**

Reference Books:

1) C. Siva Ram Murthy & B. S. Manoj, "Adhoc Wireless Networks", 2nd Edition, Pearson Education, 2005.
2) Ozan K. Tonguz and Gianguigi Ferrari & John Wiley, "Ad hoc Wireless Networks", 2006.
3) Xiuzhen Cheng, Xiao Hung, Ding Zhu Du, & Kluwer, "Ad hoc Wireless Networking", Academic Publishers, 2004.
4) C.K. Toh, "Adhoc Mobile Wireless Networks, Protocols and Systems", Prentice Hall PTR, 2007.

18UCSE709 Operations Research (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- Apply the fundamental techniques (graphical method, simplex method, dual simplex method) of operations research to solve real world problems.
- Estimate the transportation and assignment costs.
- Design, code, test, and debug programs to solve problems in the domain of operations research.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Formulate** real-world problems in the form of mathematical model and **solve** using graphical method.	-	2	1
CO-2	**Formulate** real-world problems in the form of mathematical model and **solve** using simplex and artificial variables techniques.	-	1,2,3	-
CO-3	**Solve** LP problems using dual simplex method and **perform** sensitivity analysis.	-	1,2,3	16

CO-4	**Build** Models to **Solve** Transportation and Assignment problems.	-	1,2	5,16
CO-5	**Solve** a given problem using game theory techniques.	-	1,2	5

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.8	2.0	2.0	-	1.0	-	-	-	-	-	-	-	-	-	-	1.0

Pre-requisites: Knowledge of

- Algebra
- Probability

Contents:

Unit-I

Introduction: Definitions and various phases of operations research, Role of computers in operations research, Formulating a problem as a mathematical model in the form of linear programming (LP) problem – Minization and Maximization Types, Solving LP problems using Graphical method. **7 Hrs**

Unit-II

Simplex Method: Canonical and Standard forms of LP problem, The simplex method – examples, Artificial variables Techniques (Big M and Two Phase method) - examples. **7 Hrs**

Unit-III

Advanced Topics in Linear Programming: Duality, Dual Simplex Method, Economic interpretation of duality, Sensitivity analysis, Dynamic Programming. **9 Hrs**

Unit-IV

Transportation Model: Introduction, Formulation, Methods for Initial Basic Feasible solution, Improving the basic feasible solution using Stepping Stone and MODI methods.

Assignment Model: Introduction, Formulation, Comparison with transportation model, Hungarian method. **9 Hrs**

Unit-V

Game Theory: Introduction, Formulation, Strategies – Pure and Mixed, Methods for solving Game theory problems – Saddle point, Rule of Dominance, Arithmetic method, Graphical method. **7 Hrs**

Reference Books:

1) Er. Premakumar Gupta and Dr. D.S.Hira, “Operations Research”, S Chand Publications, 2014.

2) Frederick S. Hillier and Gerald J. Lieberman, “Introduction to Operations Research”, 8th Edition, Tata McGraw Hill, 2005.
3) Wayne L. Winston, “Operations Research Applications and Algorithms”, 4th Edition, Thomson Course Technology, 2003.
4) Hamdy A T, “Operations Research: An Introduction”, 9th Edition, Pearson Publishers, 2014

18UCSE710 Multi Core Architecture and Programming (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- Basic concepts in multicore architecture.
- Synchronization and coordination mechanisms available on latest multicore machines.
- Effective concurrent program writing to enhance the performance for windows, C# & .net and Linux programming platforms.
- Introductory knowledge in Open MP libraries and pthreads.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the need for multicore architecture for modern day applications	1	2	13
CO-2	**Write** a program using pthread libraries and Open MP features to solve problems that needs thread models.	2	5	13
CO-3	**Write** a program to solve problems using parallel programming constructs.	-	4	5
CO-4	**Design** a solution for the problem using APIs for Win32, MFC and .NET;	-	5	-
CO-5	**Apply** standard solutions to some common parallel programming problems like data Race conditions, Dead locks,Live locks,	2	3	1

	Memory Issues etc.			

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.7	2.0	2.0	2.7	-	-	-	-	-	-	-	1.0	-	-	-

Pre-requisites: Knowledge of

- Microprocessor
- Operating Systems
- C programming

Contents:

Unit-I

Introduction to Multi-core Architecture: Motivation for Concurrency in software, Parallel Computing Platforms, Parallel Computing in Microprocessors, Differentiating Multi-core Architectures from Hyper-Threading Technology, Multi-threading on Single-Core versus Multi-Core Platforms Understanding Performance, Amdahl's Law, Growing Returns: Gustafson's Law.

System Overview of Threading: Defining Threads, System View of Threads, Threading above the Operating System, Threads inside the OS, Threads inside the Hardware, What Happens When a Thread Is Created, Application Programming Models and Threading, Virtual Environment: VMs and Platforms, Runtime Virtualization, System Virtualization. **8 Hrs**

Unit-II

Fundamental Concepts of Parallel Programming: Designing for Threads, Task Decomposition, Data Decomposition, Data Flow Decomposition, Implications of Different Decompositions, Challenges You'll Face, Parallel Programming Patterns, A Motivating Problem: Error Diffusion, Analysis of the Error Diffusion Algorithm, An Alternate Approach: Parallel Error Diffusion, Other Alternatives. **8 Hrs**

Unit-III

Threading and Parallel Programming Constructs: Synchronization, Critical Sections, Deadlock, Synchronization Primitives, Semaphores, Locks, Condition Variables, Messages, Flow Control- based Concepts, Fence, Barrier, Implementation-dependent Threading Features. Threading APIs: Threading APIs for Microsoft Windows, Win32/MFC Thread APIs, Threading APIs for Microsoft. NET Framework, Creating Threads, Managing Threads, Thread Pools, Thread Synchronization, POSIX Threads, Creating Threads, Managing Threads, Thread Synchronization, Signaling, Compilation and Linking. **8 Hrs**

Unit-IV

OpenMP: A Portable Solution for Threading: Challenges in Threading a Loop, Loop-carried Dependence, Data-race Conditions, Managing Shared and Private Data, Loop Scheduling and Portioning, Effective Use of Reductions, Minimizing Threading Overhead, Work-sharing Sections, Performance oriented Programming, Using Barrier and No wait, Interleaving Single-thread and Multi-thread Execution, Data Copy-in and Copy-out, Protecting Updates of Shared Variables, Intel Task queuing Extension to OpenMP, OpenMP Library Functions, OpenMP Environment Variables, Compilation, Debugging, performance. **8 Hrs**

Unit-V

Solutions to Common Parallel Programming Problems: Too Many Threads, Data Races, Deadlocks, and Live Locks, Deadlock, Heavily Contended Locks, Priority Inversion, Solutions for Heavily Contended Locks, Non-blocking Algorithms, ABA Problem, Cache Line Ping-ponging, Memory Reclamation Problem, Recommendations, Thread-safe Functions and Libraries, Memory Issues, Bandwidth, Working in the Cache, Memory Contention, Cache-related Issues, False Sharing, Memory Consistency. **7 Hrs**

Reference Books:

1. Shameem Akhter & Jason Roberts “Multicore Programming Increased Performance through Software Multi-threading”, Intel Press, 2006
2. The Software Optimization Cookbook – Intel Press 2007
3. M.Shyamala Devi, “Multicore Architectures and Programming”, Sahara Publications, 2016
4. Krishna Sankar P & Shangarnarayanee N P, “Multi-Core Architectures and Programming”, AR Publications, 2016.

18UCSE711 Internet of Things (2-0-2) 3

Contact Hours: 39 (26T + 13L)

Course Learning Objectives (CLOs): This course provides the basic understanding of IoT technology, communication protocols, sensor networks and its applications. It focuses on setting up IoT ecosystem to implement use cases by applying the key concepts of IoT.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		Mapping to POs(1-12)/ PSOs (13-16)		
		Substantial Level (3)	Moderate Level (2)	Slight Level (1)
CO-1	**Explain** the working of IoT and its enabling technologies.	-	5	1, 12
CO-2	**Explain** the use cases of IoT andthe useof sensors & actuators in IoT ecosystem.	-	5	1, 12
CO-3	**Compare** and **contrast** IoT & M2M and **explain** the generic design methodology for IoT system.	-	5, 13	1, 12
CO-4	**Develop** IoT applications using Arduino by making use of sensors and modules.	13	14	1, 5, 9, 15
CO-5	**Develop** IoT applications using Raspberry Pi microcontroller by making use of sensors and modules.	13	14	1, 5, 9, 15

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.0	-	-	-	1.6	-	-	-	1.0	-	-	1.0	2.7	2.0	1.0	-

Pre-requisites: Knowledge of Programming and Basics of Computer Networking

Contents:

Unit-I

Introduction to Internet of Things (IoT): Introduction, Physical and Logical Design of IoT, IoT Enabling Technologies, IoT levels and Deployment templates **7 Hrs**

Unit-II

Domain Specific IoTs: Introduction, Home Automation, Cities, Environment, Energy, Retail, Logistics, Agriculture, Industry, Health &Lifestyle

IoT Sensors and Actuators: Introduction, IoT sensors, RFID, Video Tracking, IoT Actuators **8 Hrs**

Unit-III

IoT and M2M: Introduction, M2M, Difference between IoT and M2M, SDN and NFV for

IoT

IoT Platforms Design Methodology: Introduction, IoT Design Methodology, Case Study on IoT System for Weather Monitoring **7 Hrs**

Unit-IV

Arduino Programming: Features of Arduino, Arduino UNO, Arduino IDE, Arduino sketch structure, Arduino function libraries, Blinking LED example; Operators, control statements, loops, arrays, string, interrupts, Traffic Control System example

Integration of Sensors, Actuators and modules with Arduino: Sensor Types, Sensor Interface with Arduino, Interfacing DHT, LM35, LDR, Accelerometer and HC-SR04 sensors with Arduino; Actuators, Types of Motor Actuators, Servo motor, Servo library on Arduino; HC05 Bluetooth module, Features, Interfacing HC05 with Arduino, Controlling LED using Bluetooth; ESP8266 wifi module, Features, Send LM35 sensor data to cloud using wifi module **2L + 7P Hrs**

Unit-V

Raspberry Pi: Introduction, Specifications, Basic Architecture, Pin configuration, Blinking LED example; Capture image using Raspberry Pi

Implementation of IoT with Raspberry Pi: Temperature dependent auto cooling system; interfacing DHT, LM35, LDR, Accelerometer and HC-SR04 sensors with Raspberry Pi; Send LM35 sensor data to cloud using wifi module. **2L + 6P Hrs**

Conduction of Practical Sessions:

Practical sessions shall include experiments on the following:

1) Use of Arduino board and coding to blink built-in as well as external LEDs
2) Interfacing temperature, humidity, soil moisture, light intensity, accelerometer, ultrasonic and obstacle detection sensors with Arduino and displaying the results
3) Connecting modules like Bluetooth and Wi-Fi to Arduino and sending the sensed data to cloud for storage and analytics.
4) Use of Raspberry Pi as a surveillance system

Reference Books:

1) Arshdeep Bahga, Vijay Madisetti ,“Internet of Things – A Hands-on Approach”, Universities Press, 2015

2) Ammar Rayes, Samer Salam, “Internet of Things From Hype to Reality – The Road to Digitalization”, Second Edition, , Springer Nature Switzerland AG 2017, 2019

Additional References:

3) NPTEL course on “Introduction to internet of things” by Prof. Sudip Misra

18UCSC800 Distributed Systems and Applications (4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course is at undergraduate level for 4 credits with emphasis on the principles of distributed systems and to design the solutions for message passing, clock synchronization, consistency and fault tolerance at basic level.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the characteristics and functioning of communication protocols in distributed systems.	-	1,2,13	-
CO-2	**Comprehend** the need for synchronization strategies in a distributed system and **Implement** a synchronization algorithm.	13,14	1,3,5,15	2
CO-3	**Explain** principles of consistency and replication strategies in a distributed system.	-	1,2,13	-
CO-4	**Elucidate** the principles of fault tolerance strategies in a distributed system.	-	1,2,13	-
CO-5	**Explain** the characteristics of distributed file system and **Implement** the file system operations of a distributed operating system.	13,14	1,3,5,15	2

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	2.0	1.6	2.0	-	2.0	-	-	-	-	-	-	-	2.4	3.0	2.0	-

Pre-requisites: Knowledge of

- Computer Networks
- Operating Systems

Contents:

Unit-I

Introduction: Need for distributed system and applications, process and communication model (in terms of IPCs RPCs, RMIs and software agents), architecture & design patterns. **10 Hrs**

Unit-II

Naming: Terminologies and different types of naming services.

ynchronization : Clock Synchronization, Logical Clocks, Mutual Exclusion, Global Positioning of Nodes, Election Algorithms **12 Hrs**

Unit-III

Consistency and Replication: Introduction, Data-centric Consistency Models, Client-centric Consistency Models, Replica Management. **10 Hrs**

Unit-IV

ault Tolerance & Security: Introduction, Process Resilience, Reliable Client-Server Communication, Reliable Client-Server Communication, Security and design issues in distributed system. **10 Hrs**

Unit-V

Distributed file systems: client-server architectures, cluster-based distributed file systems, symmetric architectures and processes. **10 Hrs**

Reference Books:

1. Andrew S Tanenbaum & Maarten van Steen, “Distributed Systems Principles and Paradigms”, 2nd Edition, Pearson Prentice Hall, 2007.
2. Tom White, “Hadoop: The Definitive Guide”, 4th Edition, O’Reilly, 2015,
3. George Coulouris, Jean Dollimore, Tim Kindberg & Gordon Blair, “Distributed Systems Concepts and Design”, 5th Edition, Addison-Wesley, 2012.
4. Ghosh, Sukumar. Distributed systems: an algorithmic approach. Chapman and Hall/CRC, 2006.

18UCSL801	Independent Study	(0-0-2) 1

Contact Hours: 26

Course Learning Objectives (CLOs): This course is a one credit self-learning based research / Development oriented activity leading to publication or a study of subjects outside regular UG course offerings; enabling the students to engage

in independent and life-long learning in the broadest context of trends of technological change and communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Learn** a technical report independently through self learning mode.	12	2,3,5,8	-
CO-2	**Prepare** a technical report with a given specification and standards.	10	5	-
CO-3	**Present** effectively the knowledge through verbal and written communication mode.	10	5	-

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	-	2.0	2.0	-	2.0	-	-	2.0	-	3.0	-	3.0	-	-	-	-

Course Guidelines:

Students are expected to pick a research oriented activities or study of subjects outside current Body-of-Knowledge of Computer Science & Engineering at SDMCET. Prior to registration, a detailed plan of work should be submitted by the student to the committee appointed by DUGC for approval. The various rules to be followed are listed below:

1. This course is offered at 7^{th} semester. Student from 5^{th} and 6^{th} semester can also register only if their CGPA is 9 and above. Duration may be spread across two semesters. But, physically this course will be reflected in the 8^{th} semester scheme and grade sheet.
2. Guide for this course is must and will be chosen by students by interacting with faculty.
3. In consultation with Guide, Students will prepare the courses description and its outcomes, which he/she promises to accomplish and is to be finally approved by DUGC.
4. Grading is done by the Guide based on the seminar and the report submitted by the students; which are to be in line with the policy established by DUGC's Grade Committee from time to time. Other assessment tool may include demonstration, seminar, quiz, tests, Viva-

Voce, publications etc. as the case is. If the Guide wishes, they can form their own examination body for evaluation with the approval of the committee on the behalf of DUGC.

5. Work worth of minimum of 26-30 Hrs spread across 10 to 12 weeks starting from the date of registration is to be ensured by the Guide.
6. Course content should belong to the Body-of-Knowledge (As per ACM, 2008 or its extensions).

Department of CSE will encourage the students to publish a paper in a reputed journal/conference or get certification through the use of well recognized MOOCS like NPTEL, Coursera, Swayam. Udacity etc…for their self-study and demonstrate highest level of ethics and professional practices.

18UCSL802 Major Project Phase – 2 (0-0-14) 7

Contact Hours: 78

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:
Understand the domain through proper modeling and analysis using the state-of-art technology. Then apply relevant Software Engineering Principles to develop modular and robust applications through the use of Standards and tools. At the end build appropriate test cases, verification and validation techniques in order to make the project reliable and maintainable.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12) / PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Identify** the problem and **formulate** the problem statement.	1, 8, 2,12	-	-
CO-2	**Analyze** the problem scenario and **Design** the solutions to complex engineering problem using software engineering principles or appropriate research methodology.	3,5, 8, 13,12	11, 16	6, 7
CO-3	**Identify** and **Implement** a feasible solution using appropriate technology, tools, procedures and techniques.	4,5, 8, 14,12	16	-
CO-4	**Verify** and **Validate** the proposed system for correctness and to demonstrate compliance with the design and hence the stated	5, 8, 15,12	-	-

	requirements/ research gap.			
CO-5	**Prepare** the report and **communicate** effectively through presentation.	8, 9,10	-	-

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	3.0	3.0	3.0	3.0	3.0	1.0	1.0	3.0	3.0	3.0	2.0	3.0	3.0	3.0	3.0	2.0

For details, refer the contents related to **Major Project – Phase 1** (**18UCSL703**).

18UCSE803 Cryptography and Network Security (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning perspectives:

- Principles of Cryptographic algorithms including secret key cryptography, hashing and public key algorithms.
- Use of cryptographic techniques to establish security in modern information- and communication systems.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Compare** and **contrast** different symmetric key encryption techniques to achieve data confidentiality.	-	1, 2	-
CO-2	**Apply** different security solutions for a given system using private and public key cryptography.	13	1, 2	-
CO-3	**Identify** the need for key management and message authentication; critically **compare** various key management, and authentication services and mechanisms.	-	3	-
CO-4	**Identify** and **explain** the requirement and usage of security services and mechanisms for various network security applications.	2	-	1
CO-5	Critically **compare** system threats and countermeasures.	-	-	13

CO-6	**Implement** a given cryptographic algorithm using higher level programming languages.	4, 5, 15	14	

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.7	2.3	2.0	3.0	3.0	-	-	-	-	-	-	-	2.0	2.0	3.0	-

Pre-requisites: Knowledge of

- Computer Networks
- Discrete Structures

Contents:

Unit-I

Classical Encryption Techniques: Symmetric Cipher Model, Cryptography, Cryptanalysis and Brute-Force Attack, Substitution Techniques.

Block Ciphers and the data encryption standard: Traditional block Cipher structure, stream Ciphers and block Ciphers, Motivation for the feistel Cipher structure, the feistel Cipher, The data encryption standard, DES encryption, DES decryption, A DES example, results, the avalanche effect, the strength of DES. **8 Hrs**

Unit-II

ublic-Key Cryptography and RSA: Principles of public-key cryptosystems. Public-key cryptosystems. Applications for public-key cryptosystems, requirements for public-key cryptosystems. Public-key cryptanalysis. The RSA algorithm, description of the algorithm.

ther Public-Key Cryptosystems: Diffie hellman key exchange, The algorithm, key exchange protocols, man in the middle attack, Elgamal Cryptographic systems, Elliptic curve arithmetic, abelian groups, elliptic curves over real numbers, elliptic curves over Zp **8 Hrs**

Unit-III

Key Management and Distribution: Symmetric key distribution using Symmetric encryption, A key distribution scenario, Hierarchical key control, session key lifetime, a transparent key control scheme, Decentralized key control, controlling key usage, Symmetric key distribution using asymmetric encryption, simple secret key distribution, secret key distribution with confidentiality and authentication.

User Authentication: Remote user Authentication principles, Mutual Authentication, one way Authentication,remote user Authentication using Symmetric encryption, Mutual Authentication,one-way Authentication,Kerberos, Motivation, Kerberos version 4, Kerberos version 5 **8 Hrs**

Unit-IV

Wireless network security: 802.11i pseudorandom Wireless security, Wireless network threats, Wireless network measures, mobile device security, security threats, mobile device security strategy, IEEE 802.11 Wireless LAN overview

ansport Level Security: Web Security Considerations, Web Security Threats, Web Traffic Security Approaches. Secure Sockets Layer: SSL Architecture, SSL Record Protocol, Change Cipher Spec Protocol, Alert Protocol, and shake Protocol, Transport Layer Protocol, Connection Protocol. **8 Hrs**

Unit-V

Electronic Mail Security: Pretty good privacy, notation, operational description, S/MIME, RFC5322, Multipurpose internet mail extensions, S/MIME functionality.

IP Security: IP Security overview, applications of IPsec, benefits of IPsec, Routing applications, IPsec documents, IPsec services, transport and tunnel modes, IP Security policy, Security associations, Security associations database, Security policy database. **7 Hrs**

Reference Books:

1. William Stallings, "Cryptography and Network Security", 6th Edition, Pearson Education, 2014.
2. Atul Kahate, "Cryptography and Network Security", 3rd Edition, McGraw-Hill Education (India) Pvt. Ltd., 2013
3. Bruce Schinner, "Applied Cryptography", 2nd Edition, PHI publication 2007.
4. Radia Perlman, Mike Speciner, & Charlie Kaufman, "Network Security: Private communication in a Public World", 2nd Edition, Pearson Education Asia, 2002.

18UCSE804 Cloud Computing (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on the following learning objectives:

Understand the cloud computing paradigm. Recognize its various forms and implement some cloud computing features.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** cloud computing philosophy & computing paradigms.	-	1,4	5
CO-2	**Distinguish** and **demonstrate** cloud service models and their risks.	-	2,4,5	-

CO-3	**Explain** and **use** SLA life cycle and agreement structure to build simple applications.	-	~~4~~,8	7
CO-4	**Explain** and **build** simple security models using cloud security principles.	-	1,2,4	5
CO-5	**Develop** and **deploy** cloud solution for the given application scenario.	1, 2, 3	4, 5	7

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	2.3	2.3	3.0	2.0	1.5	-	1.0	2.0	-	-	-	-	-	-	-	-

Pre-requisites: Knowledge of

- Operating Systems, Computer Architecture, Computer Networks & Web Technologies.
- Programming languages- Java, Web programming

Contents:

Unit-I

Introduction: Paradigms in Computing, Parallel Computing, Distributed Computing, Grid Computing, Service Computing; Service Oriented Architecture (SOA), Web Services. **7 Hrs**

Unit-II

ɑsics of Cloud Computing: Definition, Characteristics, Architecture, Components, Service Models, and Deployment Models.

rtualization: Server, Storage, Network, Desktop Hypervisor, Virtual Machine, Multi-tenancy, Opportunities and Risks. **7 Hrs**

Unit-III

Service Level Agreement (SLA): Definition, Types of SLA, SLA Life Cycle, Issues Related to Cloud SLA, SLA Frameworks: WS-Agreement, WSLA, WSOL, Slang, Bilateral Protocol; Translation of SLAs into Monitoring Specifications, Dynamic Creation of Monitoring Infrastructures, Penalty Management, Runtime Prediction.

7 Hrs

Unit-IV

Cloud Security: Cloud Security Fundamentals, Vulnerability Assessment, Security and Privacy in Cloud.

oud Computing Security Architecture: Identity Management and Access Control, Autonomic Security; VM Specific Security Techniques. **9 Hrs**

Unit-V

Cloud Application Programming Models: Cloud File Systems: GFS and HDFS, BigTable, HBase and Dynamo; Map Reduce Programming Model, Hadoop: Hadoop Fundamentals, Hama and other Hadoop Related Services. Cloud Application Development Platforms: Xen Hypervisor, Amazon Web Service, Windows Azure, Google App Engine, Eucalyptus, Open Stack, Open Nebula. **9 Hrs**

Experiments

1. Use of Cloud Web Services
2. Experiment on Cloud Storage using mobile
3. Explore Cloud platform like Amazon Web Service, Windows Azure, Google App Engine etc.

Reference Books:

1. A. T. Velte, “Cloud Computing - A Practical Approach”, McGraw Hills 2010
2. Tanenbaum & V. Steen, “Distributed Systems: Principles and Paradigms”, 2nd Edition, Pearson, 2015
3. David E.Y. Sarna, “Implementing and Developing Cloud Computing Applications”, CRC Press, 2011
4. R. Krutz & R. D. Vines, “Cloud Security”, Wiley-India, 2010
5. T. White, “Hadoop: The Definitive Guide”, 4th Edition, O’Reilly Media, 2015
6. Judith Hurwitz, R.Bloor, M.Kanfman, F.Halper, “Cloud Computing for Dummies”, Wiley India Edition
7. Rajkumar Buyya, James Broberg & Andrzej Goscinski, “Cloud Computing Principles and Paradigms”, Willey 2014.
8. Dan C Marinescu, “Cloud Computing Theory and Practice”, Elsevier (MK), 2013.

18UCSE805 Network Management (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course, at under graduate level for 3 credits, explains the issues for network management arising from a range of security threats, including viruses and denial-of-service attacks. Develop a strategy for ensuring appropriate levels of security in a system designed for a particular purpose.

Course Outcomes (COs):

Description of the Course Outcome:	Mapping to POs(1-12)/ PSOs (13-16)

At the end of the course the student will be able to:		Substantial Level (3)	Moderate Level (2)	Slight Level (1)
CO-1	**Explain** network communication architecture and goals	2,13	-	-
CO-2	**Explain** and **apply** Management information Tree (MIT)	13,14	3, 4	5
CO-3	**Explain** structure of management through Management Information Base	13	4	3
CO-4	**Explain** Remote Monitoring, and concept of broadband, security.	-	4,5	7
CO-5	**Design** and **develop** network management solutions for the given scenario like Inventory management, fault location and management, performance management, accounting management, report management.	-	10	-

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	3.0	2.0	2.0	1.5	-	1.0	-	-	2.0	-	-	3.0	3.0	-	-

Pre-requisites: Knowledge of Computer Networks

Contents:

Unit-I

Introduction: Some Common Network Problems; Challenges of Information Technology Managers, Network Management: Goals, Organization, and Functions Goal of Network Management, Network Provisioning, Network Operations and the NOC, Network Installation and Maintenance; Network and System Management, Network Management System platform, Current Status and Future of Network Management

Basic Foundations: Standards, Models, and Language: Network Management Standards, Network Management Model, Organization Model, Information Model – Management Information Trees, Managed Object Perspectives, Communication Model; ASN.1 Terminology, Symbols and Conventions, Objects and Data Types, Object Names, An Example of ASN.1Prere from ISO 8824; Encoding Structure; Macros, Functional Model. **8 Hrs**

Unit-II

NMPv1 Network Management : Organization and Information Models : Managed Network: The History of SNMP Management, Internet Organizations and standards, Internet Documents, The SNMP Model, The Organization Model, System Overview,

The Information Model – Introduction, The Structure of Management Information, Managed Objects, Management Information Base **8 Hrs**

Unit-III

SNMPv1 Network Management: Communication and Functional Models: The SNMP Communication Model – The SNMP Architecture, Administrative Model, SNMP Specifications, SNMP Operations, SNMP MIB Group, and Functional Model.

SNMP Management- RMON: Remote Monitoring, RMON SMI and MIB, RMONI1 RMON1 Textual Conventions, RMON1 Groups and Functions, Relationship Between Control and Data Tables, RMON1 Common and Ethernet Groups, RMON Token Ring Extension Groups, RMON2 – The RMON2 Management Information Base, RMON2.

8 Hrs

Unit-IV

roadband Network Management: ATM Networks: Broadband Networks and Services, ATM Technology – Virtual Path Virtual Circuit, TM Packet Size, Integrated Service, SONET, ATM LAN Emulation, Virtual LAN; ATM Network Management – The ATM Network Reference Model, The Integrated Local Management Interface, The ATM Management Information Base, The Role of SNMP and ILMI in ATM Management, M1 Interface: Management of ATM Network Element, M2 Interface: Management of Private Networks, M3 Interface: Customer Network Management of Public Networks, M4 Interface: Public Network Management, Management of LAN Emulation. ADSL Architecture, ADSL Channeling Schemes, ADSL Encoding Schemes; ADSL Management ADSL Network Management Elements **8 Hrs**

Unit-V

Network Management Tools: Introduction to various network management tools. Server Management: Application Servers, Web Server. Device configuration like Fire wall settings, Hot Spot, Gate Ways, Switches etc. **7 Hrs**

Experiments:

1. Analyze bandwidth throughput etc. for a given network.
2. Analyze different protocols performance like TCP and UDP for a given point-to-point network.
3. Analyze routing protocol performance.

Reference Books:

1. Mani Subramanian, “Network Management Principles and Practice”, 2nd Edition, Pearson Education Publication, 2010.
2. Jianguo Ding, “Advances in Network Management”, 1st Edition, Auerbach Publications, 2016.

3. Allan Leinwand, Karen Fang Conroy, & Karen Fang, “Network Management: A Practical Perspective”, Addison Wesley, 1996
4. Alexander Clemm, “Network Management Fundamentals”, Cisco, 1st Edition, 2016.

18UCSE807 Ontology & Semantic Web (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- Theoretical and practical aspects of ontology in semantic web.
- Concepts of RDF and web ontology language.
- Various patterns for developing and reusing ontologies.
- Models for knowledge management architecture in semantic web.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the working principles, architecture of semantic web and the importance of markup languages.	1	-	-
CO-2	**Design** and **analyze** ontologies using web ontology language.	-	3	-
CO-3	**Apply** the logic with web ontology language.	15	-	5
CO-4	**Analyze** scalable architectures and semantic web services.	-	2	-
CO-5	**Explain** the evolution of ontologies with semantic interpretation of information.	-	12	-

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	3.0	2.0	2.0	-	1.0	-	-	-	-	-	-	2.0		-	3.0	-

Pre-requisites: Knowledge of HTML and Java

Contents:

Unit-I

The semantic web vision: Today's web, Semantic web Technologies, A layered approach, Structured web documents in XML, Introduction to RDF, RDF syntax, RDF schema, Axiomatic statements for RDF and RDF schema, Direct inference system on RDF and RDF schema. **9 Hrs**

Unit-II

ntology and Web ontology Language: Introducing OWL, Discovering the Various Species of OWL, Exploring the Foundations of OWL Understanding OWL Essentials, Making Simple Assertions, Inconsistency, Examining Property Characteristics, Complex Classes, Understanding Why OWL Is Different, Developing OWL Ontologies. **8 Hrs**

Unit-III

Ontology building and inference by logic: Monotonic rules - syntax and semantics, Nonmonotonic rules - syntax, Rule markup in XML, Constructing and reusing ontologies, Semantic web knowledge management architecture, Querying using SPARQL, Jena ontology API, Reasoners. **8 Hrs**

Unit-IV

ntology scalable architectures and Discovering the semantic web services: Discovering the Roles, Creating Semantics for Enterprise Systems, Scaling Semantic Web Tools, Patterns of Architectural Usage and Applicationdevelopment using OWL API. **7 Hrs**

Unit-V

Ontology evolution and Semantic knowledge: Ontology population and enrichment, Semantic representation of multimedia content, Ontology based semantics extraction from text and Images. A survey of semantic image and video annotation tools, Overview of linked data. **7 Hrs**

Reference Books:

1. Grigoris Antoniou and Frank van Harmelen, "A Semantic Web Primer", 2nd Edition, MIT Press, Cambridge, MA 2008
2. Jeffrey T.Pollock, "Semantic Web For Dummies", Wiley Publishing, Inc 2009
3. John Hebeler, Matthew Fisher, Ryan Blace, & Andrew Perez-Lopez, "Semantic Web Programming", Wiley Publishing, Inc.
4. Georgios Paliouras, Constantine Spyropoulos, & George Tsatsaronis (Eds.) "Knowledge-Driven Multimedia Information Extraction and Ontology Evolution", Springer, 2011

18UCSE808	Data Science	(3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course is a 3 credit undergraduate course focusing on the data and types, data preprocessing. The details of R language are discussed to enable the students to perform data analytics. It also discusses the supervised and unsupervised learning.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the flow process for processing data science problems and the data types.	-	1, 2	13
CO-2	**Apply** the preprocessing methods to the raw data to make it available for analysis.	1, 2	-	-
CO-3	**Use** the R language to perform the data visualization and the analysis using fundamental statistical technques.	3	-	12
CO-4	Explain the machine learning technques.	-	1, 2	13
CO-5	Explain the unsupervised learning technques.	-	1, 2	13

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	2.3	2.3	3.0	-	-	-	-	-	-	-	-	1.0	1.0	-	-	-

Pre-requisites: Knowledge of Statistics

Contents:

Unit-I

troduction: Evolution, Roles, Stages in Data Science Project, Applications, Data Security and Privacy Issues

ata: Data types - Structured and Unstructured, Challenges with Unstructured data, Social media data, Multi modal data, Data Storage and Presentation **7 Hrs**

Unit-II

ata Preprocessing: Cleaning, Integration, Transformation, Reduction, Discretization.

echniques: Correlation, Regression. Exploratory Analysis. **7 Hrs**

Unit-III

Language: Basics, Control structures, Functions, Impoting Data.
raphics and Data Visualization: Installing gglot2, Loading the data, Plotting the Data.
Statistics and Machine Learning: Basic Statistics, Regression, Clustering **9 Hrs**

Unit-IV

achine Learning: Introduction, Regression, Classification, Gradient Descent. **8 Hrs**

Unit-V

Unsupervised Learning: Introduction, Agglomerative Clustering, Reinforcement Learning **8 Hrs**

Reference Books:

1) Chirag Shah, "A Hands on Introduction to Data Science", Cambridge University Press, 2020
2) Laura Igual and Santi Segui, "Introduction to Data Science", Springer International Publications, 2017
3) Richord Cotton "Learning R", O'Reilly Publications, 2013.

15UCSC800 Distributed Systems (4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- To know the design principles of distributed systems, architectures and
- To prepare design solutions using latest industry relevant technology at basic level.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the functional characteristics of various components of distributed system architecture.	-	13	-
CO-2	**Explain** the functioning of various communication protocols and **Write a program** using appropriate technology/protocol to build sub systems that makes a distributed system.	13,14	-	15
CO-3	**Explain** the need and principles of various synchronization strategies used in distributed system.	-	1,2,13	-
CO-4	**Explain** the need and principles of various consistency strategies and replication necessary for a distributed system.	-	1,2,13	-
CO-5	**Explain** the need and principles of various fault tolerance strategies necessary for a distributed system.	-	1,2,13	-
CO-6	**Explain** the functional characteristics of various components of an industry relevant product used in building distributed file system and **write programs** to perform storage and retrieval	-	2,3,5,13,14	15

	operations.			
CO-7	**Explain** the architectural patterns and **use** it for developing distributed system.	-	13	-

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.0	2.0	-	2.0	-	-	-	-	-	-	-	2.1	2.5	1.0	-

Pre-requisites: Knowledge of

- Computer Networks
- Operating Systems

Contents:

Unit-I

Introduction: Need for distributed system and applications, process and communication model (in terms of IPCs RPCs, RMIs and software agents), architecture & design patterns

Naming: Terminologies and different types of naming services. **9 Hrs**

Unit-II

ynchronization : Clock Synchronization, Logical Clocks, Mutual Exclusion, Global Positioning of Nodes, Election Algorithms **11 Hrs**

Unit-III

Consistency and Replication: Introduction, Data-centric Consistency Models, Client-centric Consistency Models, Replica Management, Consistency Protocols **12 Hrs**

Unit-IV

ult Tolerance & Security: Introduction, Process Resilience, Reliable Client-Server Communication, Reliable Group Communication, Distributed Commit, Recovery.

11 Hrs

Unit-V

A case study on distributed file system used in latest industry relevant technology/platform

A case study on industry relevant technology based on Proxy & Broker Architecture.

9 Hrs

Reference Books:

5. Andrew S Tanenbaum & Maarten van Steen, “Distributed Systems Principles and Paradigms”, 2nd Edition, Pearson Prentice Hall, 2007.
6. Tom White, “Hadoop: The Definitive Guide”, 4th Edition, O’Reilly, 2015,

7. George Coulouris, Jean Dollimore, Tim Kindberg & Gordon Blair, “Distributed Systems Concepts and Design”, 5th Edition, Addison-Wesley, 2012.
8. Brendan Burns, “Designing Distributed Systems: Patterns and Paradigms for Scalable , Reliable Services”, O’Reilly Publications, 2018.

15UCSC801 Independent Study (0-0-4) 2

Contact Hours: 52

Course Learning Objectives (CLOs): This course on Independent Study is a self-learning based research oriented activity leading to publication or a study of subjects outside regular course offerings under the guidance of a faculty member. This enables the students to engage in independent and life-long learning in the broadest context of technological change and communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Learn** a technical topic independently through self-learning mode.	12	2,3,8	5
CO-2	**Prepare** a technical report with a given specification and standard.	10	-	-
CO-3	**Present** effectively the knowledge through verbal and written communication mode.	10	-	-

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	-	2.0	2.0	-	1.0	-	-	2.0	-	3.0	-	3.0	-	-	-	-

Course Guidelines:

Students are expected to pick a research oriented activities or study of subjects outside current Body-of-Knowledge of Computer Science & Engineering at SDMCET. (Examples - Computational Geometry, Automatic Verification, Reconfigurable Computing etc.).Prior to registration, a detailed

plan of work should be submitted by the student to the committee appointed by DUGC for approval. The various rules to be followed are listed below.

1. This course is offered from 7th to 8th semester. Student from 5th and 6th semester can also register only if their CGPA is 9 and above. Duration may be spread across two semesters. But, physically this course will be reflected in the 8th semester scheme and grade sheet
2. Guide for this course is must and will be chosen by students by interacting with faculty.
3. In consultation with Guide, Students will prepare the courses description and its outcomes, which he/she promises to accomplish and is to be finally approved by DUGC.
4. Grading is done by the Guide based on the seminar and the report submitted by the student; which are to be in line with the policy established by DUGC's Grade Committee from time to time. Other assessment tool may include demonstration, seminar, quiz, tests, Viva-Voce, publications etc. as the case is. If the Guide wishes, they can form their own examination body for evaluation with the approval of the committee on the behalf of DUGC.
5. Work worth of minimum of 26-30 Hrs spread across 10 to 12 weeks starting from the date of registration is to be ensured by the Guide.
6. Course content should belong to the Body-of-Knowledge (As per ACM, 2008or its extensions).

Department of CSE will encourage the students to publish a paper in a reputed journal/conference or get certification through the use of well recognized MOOCS like NPTEL, Coursera, Swayam. Udacity etc…for their self-study and demonstrate highest level of ethics and professional practices.

15UCSL802	**Major Project Phase – 2**	**(0-0-20) 10**

Contact Hours: 96

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- **Understand the domain, Analyze through Modeling and Implementation through state of the art technology available.**
- **Know Software Engineering Principles: Modeling, Estimation, Design standards and architectural issues through use of Standards etc.**
- **To write modular programs and handle exceptions to provide reliable solutions.**

- **To test and verify the programs for different scenarios.**

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Identify** and **formulate** the problem.	1,2,12	8	-
CO-2	**Analyze** the problem scenario and **Design** the solutions to complex engineering problem using software engineering principles or appropriate research methodology.	3,5,12,13	8,11	6,7,16
CO-3	**Identify** and **Implement** a feasible solution using appropriate technology, tools, procedures and techniques.	4,5,12,14	8	16
CO-4	**Verify** and **Validate** the proposed system for correctness and to demonstrate compliance with the design and hence the stated requirements/ research gap.	5,12,15	8	-
CO-5	**Prepare** the report and **communicate** effectively through presentation.	9,10	8	-

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	3.0	3.0	3.0	3.0	3.0	1.0	1.0	2.0	3.0	3.0	2.0	3.0	3.0	3.0	3.0	1.0

For details, refer the contents related to Major Project – Phase I (15UCSL703).

15UCSE803 Data Warehousing and Mining (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- Compare and contrast different conceptions of data mining as evidenced in both research and application.
- Explain the role of finding associations in commercial market basket data.
- Characterize the kinds of patterns that can be discovered by association rule mining.
- Describe how to extend a relational system to find patterns using association rules.
- Evaluate methodological issues underlying the effective application of data mining.
- Identify and characterize sources of noise, redundancy, and outliers in presented data.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Analyze** the given scenario and **perform** data pre-processing using appropriate suitable methods.	1,2	-	-
CO-2	**Describe** the various architectures and main components of a data warehouse.	1,2	-	-
CO-3	**Analyse** the data using various standard algorithms in order to determine patterns and their relationships from different types of large data set.	1,13	2,5	-
CO-4	**Predict** group membership for data instances using different approaches	2	3,5	14
CO-5	**Explain** and **apply** the various clustering techniques on the given data set.	1,2	5	14

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	3.0	2.8	2.0	-	2.0	-	-	-	-	-	-	-	3.0	1.0	-	-

Pre-requisites: Knowledge of Statistics

Contents:

Unit-I

Introduction and Basic Concepts: What is Data Mining?, Definition, Comparison of DBMS and Data Mining, Motivating Challenges, The Origins of Data Mining, Data Mining Tasks, Applications of Data Mining, Data Mining Process.

Data: Types of Data: attributes, data types, Data Quality: Measurement and Data Collection Issues, Issues related to applications, Data pre-processing, Proximity measures: Basics, Examples, Issues in proximity calculations, and selecting the right proximity measure. **9 Hrs**

Unit-II

ata Warehouse and OLAP Technology for Data Mining: Data Warehouse, Multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, From Data Warehousing to Data Mining, Data Cube Computation and Data Generalization. **7 Hrs**

Unit-III

Association Analysis: What is an Association rule, Methods to discover association rules, Apriori algorithm, Partition algorithm, Dynamic Item set Counting algorithm, FP tree Growth algorithm, Discussion on different algorithms. **8 Hrs**

Unit-IV

assification: Preliminaries, General approach to solving a classification problem, Decision tree induction, Model over fitting, Evaluation of the performance of a classifier; Comparing of classifiers, Rule based Classification, Nearest Neighbour classifiers. **8 Hrs**

Unit-V

Cluster Analysis: Introduction, Types of Clustering, A Categorization of Major Clustering Methods, Partitioning Methods, Hierarchical Methods, Density-Based Methods, Grid-Based Methods, Model-Based Clustering Methods. **7 Hrs**

Reference Books:

5. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, “Introduction to Data Mining”, 3rd Edition, Pearson Education, 2014.
6. Jiawei Han &MichelineKamber, “Data Mining – Concepts and Techniques”, 3rd Edition, Morgan Kaufmann Publishers, 2011.
7. Arun K Pujari, “Data Mining Techniques”, 3rd Edition, Universities Press, 2013.

8. K. P. Soman, S. Diwakar & V. Ajay, “Insight into Data Mining”, Prentice Hall India, 2008.
9. Paulraj Ponnaiah,”Data Warehousing Fundamentals for IT Professionals”, 2nd Edition, 2010.

15UCSE804 Cryptography and Network Security (4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- Principles of Cryptographic algorithms including secret key cryptography, hashing, message digests, and public key algorithm.
- Network security issues involving standalone computers, locally networked computers and remotely networked computers.
- Use of cryptographic techniques to establish security in modern information- and communication systems.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Compare** different symmetric key encryption techniques to achieve data confidentiality	-	1, 2	-
CO-2	**Apply** different security solutions for a given system using private and public key cryptography.	13	1,2	-
CO-3	**Identify** the need for key management and message authentication; **compare** various key management, and authentication services and mechanisms.	-	3	-
CO-4	**Identify** and **explain** the requirement and usage of security services and mechanisms for various network security applications.	2	-	1
CO-5	**Explain** the vaious security threats and counter measures.	-	-	13
CO-6	**Implement** a given cryptographic	4,5,15	14	-

	algorithm using higher level programming languages			

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.7	2.3	2.0	3.0	3.0	-	-	-	-	-	-	-	2.0	-	3.0	-

Pre-requisites: Knowledge of

- Computer Networks
- Data Structures
- Discrete Mathematical Structures

Contents:

Unit-I

Classical Encryption Techniques: Symmetric Cipher Model, Cryptography, Cryptanalysis and Brute-Force Attack, Substitution Techniques, Caesar Cipher, Monoalphabetic Cipher, Playfair Cipher, Hill Cipher, Polyalphabetic Cipher, One Time Pad. Block Ciphers and the data encryption standard: Traditional block Cipher structure, stream Ciphers and block Ciphers, Motivation for the feistel Cipher structure, the feistel Cipher, The data encryption standard, DES encryption, DES decryption, A DES example, results, the avalanche effect, the strength of DES, the use of 56-Bit Keys, the nature of the DES algorithm, Block cipher design principles, number of rounds, design of function F, key schedule algorithm. **12 Hrs**

Unit-II

ıblic-Key Cryptography and RSA: Principles of public-key cryptosystems. Public-key cryptosystems. Applications for public-key cryptosystems, requirements for public-key cryptosystems. Public-key cryptanalysis. The RSA algorithm, description of the algorithm, computational aspects, the security of RSA. Other Public-Key Cryptosystems: Diffie hellman key exchange, The algorithm, key exchange protocols, man in the middle attack, Elgamal Cryptographic systems, Introduction to Elliptic curve cryptography. **10 Hrs**

Unit-III

Key Management and Distribution: Symmetric key distribution using Symmetric encryption, A key distribution scenario, Hierarchical key control, session key lifetime, a transparent key control scheme, Decentralized key control, controlling key usage, Symmetric key distribution using asymmetric encryption, simple secret key distribution, secret key distribution with confidentiality and authentication, A hybrid scheme, distribution of public keys, public announcement of public keys, publicly available directory, public key authority, public keys certificates, X-509 certificates.

Certificates, X- 509 version 3, public key infrastructure, User Authentication: Remote user Authentication principles, Mutual Authentication, one way Authentication, remote user Authentication using Symmetric encryption, Mutual Authentication, one

way Authentication, Kerberos, Motivation , Kerberos version 4, Kerberos version 5, Remote user Authentication using Asymmetric encryption, Mutual Authentication, one way Authentication. **10 Hrs**

Unit-IV

ireless network security: Wireless security, Wireless network threats, Wireless network measures, mobile device security, security threats, mobile device security strategy, IEEE 802.11 Wireless LAN overview, the Wi-Fi alliance, IEEE 802 protocol architecture. Security, IEEE 802.11i services, IEEE 802.11i phases of operation, discovery phase, Authentication phase, key management phase, protected data transfer phase, the IEEE 802.11i pseudorandom function, . Web Security Considerations: Web Security Threats, Web Traffic Security Approaches. Secure Sockets Layer: SSL Architecture, SSL Record Protocol, Change Cipher Spec Protocol, Alert Protocol, and shake Protocol, Transport Layer Security: Version Number, Message Authentication Code, Pseudorandom Functions, Alert Codes, Cipher Suites, Client Certificate Types, Certificate Verify and Finished Messages, Cryptographic Computations, Padding. HTTPS Connection Initiation, Connection Closure. Secure Shell (SSH) Transport Layer Protocol, User Authentication Protocol, Connection Protocol. **10 Hrs**

Unit-V

Electronic Mail Security: Pretty good privacy, notation, operational; description, S/MIME, RFC5322, Multipurpose internet mail extensions, S/MIME functionality, S/MIME messages, S/MIME certificate processing, enhanced security services, Domain keys identified mail, internet mail architecture, E-Mail threats, DKIM strategy, DKIM functional flow. IP Security: IP Security overview, applications of IPsec, benefits of IPsec, Routing applications, IPsec documents, IPsec services, transport and tunnel modes, IP Security policy, Security associations, Security associations database, Security policy database, IP traffic processing, Encapsulating Security payload, ESP format, encryption and authentication algorithms, Padding, Anti replay service, transport and tunnel modes, combining security associations, authentication plus confidentiality, basic combinations of security associations, internet key exchange, key determinations protocol, header and payload formats. **10 Hrs**

Reference Books:

5. William Stallings, “Cryptography and Network Security”, 6th Edition, Pearson Education, 2014.
6. Atul Kahate, “Cryptography and Network Security”, 3rd Edition, McGraw-Hill Education (India) Pvt. Ltd., 2013
7. Bruce Schinner, “Applied Cryptography”, 2nd Edition, PHI publication 2007.
8. Radia Perlman, Mike Speciner, & Charlie Kaufman, “Network Security: Private communication in a Public World”, 2nd Edition, Pearson Education Asia, 2002.

15UCSE805 Cloud Computing (4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- Undrestand the cloud computing paradigm.
- Recognize its various forms and able to implement some cloud computing features.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** cloud computing philosophy & computing paradigms.	1	4	5
CO-2	**Distinguish** and **demonstrate** cloud service models and their risks.	2,5	4	-
CO-3	**Explain** SLA life cycle and agreement structure.	4	8	7
CO-4	**Explain** basic fundamentals of cloud security principles and their vulnerability.	2,4	1	5
CO-5	**Develop** and **deploy** cloud solution for the given application scenario.	1, 2, 3	4, 5	7

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	2.7	3.0	3.0	2.4	1.8	-	1.0	2.0	-	-	-	-	-	-	-	-

Pre-requisites: Knowledge of

- Operating Systems, Computer Architecture, Computer Networks & Web Technologies.
- Programming languages- Java, Web programming

Contents:

Unit-I

Introduction: Paradigms in Computing, Parallel Computing, Distributed Computing, Grid Computing, Service Computing; Service Oriented Architecture (SOA), Web Services. **9 Hrs**

Unit-II

oud Computing: Definition, Characteristics, Architecture, Components, Service Models, and Deployment Models, Virtualization: Server, Storage, Network, Desktop Hypervisor, Virtual Machine, Multi-tenancy, Opportunities and Risks. **11 Hrs**

Unit-III

Service Level Agreement (SLA): Definition, Types of SLA, SLA Life Cycle, Issues Related to Cloud SLA, SLA Frameworks: WS-Agreement, WSLA, WSOL, Slang, Bilateral Protocol; Translation of SLAs into Monitoring Specifications, Dynamic Creation of Monitoring Infrastructures, Penalty Management, Runtime Prediction. **10 Hrs**

Unit-IV

oud Security: Cloud Security Fundamentals, Vulnerability Assessment, Security and Privacy in Cloud, Cloud Computing Security Architecture: Identity Management and Access Control, Autonomic Security; VM Specific Security Techniques. **10 Hrs**

Unit-V

Cloud Application Programming Models: Cloud File Systems: GFS and HDFS, BigTable, HBase and Dynamo; Map Reduce Programming Model, Hadoop: Hadoop Fundamentals, Hama and other Hadoop Related Services. Cloud Application Development Platforms: Xen Hypervisor, Amazon Web Service, Windows Azure, Google App Engine, Eucalyptus, Open Stack, Open Nebula. **12 Hrs**

Experiments

4. Use of Cloud Web Services
5. Experiment on Cloud Storage using mobile
6. Explore Cloud platform like Amazon Web Service, Windows Azure, Google App Engine etc.

Reference Books:

9. A. T. Velte, “Cloud Computing - A Practical Approach”, McGraw Hills 2010
10. Tanenbaum & V. Steen, “Distributed Systems: Principles and Paradigms”, 2nd Edition, Pearson, 2015
11. David E.Y. Sarna, “Implementing and Developing Cloud Computing Applications”, CRC Press, 2011
12. R. Krutz & R. D. Vines, “Cloud Security”, Wiley-India, 2010
13. T. White, “Hadoop: The Definitive Guide”, 4th Edition, O’Reilly Media, 2015
14. Judith Hurwitz, R.Bloor, M.Kanfman, F.Halper, “Cloud Computing for Dummies”, Wiley India Edition

15. Rajkumar Buyya, James Broberg & Andrzej Goscinski, “Cloud Computing Principles and Paradigms”, Willey 2014.
16. Dan C Marinescu, “Cloud Computing Theory and Practice”, Elsevier (MK), 2013.

15UCSE806 Mobile Computing (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- To understand the concepts of wireless communication, propagation methods, Channel models, capacity calculations multiple antennas and multiple user techniques used in the mobile communication.
- To know CDMA, GSM, Mobile IP, WImax.
- To understand various tools/techniques available to develop different mobile Internet applications.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the different mobile computing architectures like GSM, GPRS etc.	-	1	-
CO-2	**Explain** issues related to mobility.	-	4,13	-
CO-3	**Explain** different Mobile OS and issues related to user interface.	-	14	4,5
CO-4	**Design** and **develop** a mobile Internet solution for the given application scenario.	14	16	15

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	2.0	-	-	1.5	1.0	-	-	-	-	-	-	-	2.0	2.5	1.0	2.0

Pre-requisites: Knowledge of Computer Networks

Contents:

Unit-I

Mobile Computing Architecture: Architecture for Mobile Computing, 3-tier Architecture, Design Considerations for Mobile Computing. Wireless Networks: Global Systems for Mobile Communication (GSM and Short Service Messages (SMS): GSM

Architecture, Entities, Call routing in GSM, PLMN Interface, GSM Addresses and Identities, Network Aspects in GSM, Mobility Management, GSM Frequency allocation. Introduction to SMS, SMS Architecture, SM MT, SM MO, SMS as Information bearer, applications, GPRS and Packet Data Network, GPRS Network Architecture, GPRS Network Operations, Data Services in GPRS, Applications for GPRS, Billing and Charging in GPRS, Spread Spectrum technology, IS-95, CDMA versus GSM, Wireless Data, Third Generation Networks, Applications on 3G, Introduction to WiMAX. **8 Hrs**

Unit-II

obile Client: Moving beyond desktop, Mobile handset overview, Mobile phones and their features, PDA, Design Constraints in applications for handheld devices. Mobile IP: Introduction, discovery, Registration, Tunneling, Cellular IP, Mobile IP with IPv6 **7 Hrs**

Unit-III

Mobile OS and Computing Environment: Smart Client Architecture, The Client: User Interface, Data Storage, Performance, Data Synchronization, Messaging. The Server: Data Synchronization, Enterprise Data Source, Messaging. Mobile Operating Systems: WinCE, Palm OS, Symbian OS, Linux and Proprietary OS Client Development: The development process, Need analysis phase, Design phase, Implementation and Testing phase, Deployment phase, Development Tools, Device Emulators. **7 Hrs**

Unit-IV

uilding Mobile Internet Applications: Thin client: Architecture, the client, Middleware, messaging Servers, Processing a Wireless request, Wireless Applications Protocol (WAP) Overview, Wireless Languages: Markup Languages, HDML, WML, HTML, cHTML, XHTML, VoiceXM. **9 Hrs**

Unit-V

Introduction, CDC, CLDC, MIDP: Programming for CLDC, MIDlet model, Provisioning, MIDlet lifecycle, Creating new application, MIDlet event handling, GUI in MIDP, Low level GUI Components, Multimedia APIs; Communication in MIDP, Security Considerations in MIDP. **8 Hrs**

Reference Books:

1. Ashok Talukder, Roopa Yavagal & Hasan Ahmed, “Mobile Computing, Technology, Applications and Service Creation”, 2nd Edition, Tata McGraw Hill, 2010.
2. Martyn Mallik, “Mobile and Wireless Design Essentials”, Wiley India, 2003
3. Raj Kamal, “Mobile Computing”, Oxford University Press, 2007.

4. Iti Saha Misra, “Wireless Communications and Networks, 3G and Beyond”, Tata McGraw Hill, 2009.

15UCSE807 Network Management (4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- Explain the issues for network management arising from a range of security threats, including viruses, worms, Trojan horses, and denial-of-service attacks.
- Develop a strategy for ensuring appropriate levels of security in a system designed for a particular purpose.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** network communication architecture and goals	2	-	-
CO-2	**Explain** and **apply** Management information Tree (MIT)	13	4	3
CO-3	**Explain** structure of management through Management Information Base	13	4	3
CO-4	**Explain** Remote Monitoring, and concept of broadband, security.	-	5	7
CO-5	**Design** and **develop** network management solutions for the given scenario like Inventory management, fault location and management, performance management, accounting management, report management.	-	10	-

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	-	3.0	1.0	2.0	2.0	-	1.0	-	-	2.0	-	-	3.0	-	-	-

Pre-requisites: Knowledge of Computer Networks

Contents:

Unit-I

Introduction: Some Common Network Problems; Challenges of Information Technology Managers, Network Management: Goals, Organization, and Functions Goal of Network Management, Network Provisioning, Network Operations and the NOC, Network Installation and Maintenance; Network and System Management, Network Management System platform, Current Status and Future of Network Management

Basic Foundations: Standards, Models, and Language: Network Management Standards, Network Management Model, Organization Model, Information Model – Management Information Trees, Managed Object Perspectives, Communication Model; ASN.1 Terminology, Symbols, and Conventions, Objects and Data Types, Object Names, An Example of ASN.1Prere from ISO 8824; Encoding Structure; Macros, Functional Model. **12 Hrs**

Unit-II

NMPv1 Network Management : Organization and Information Models : Managed Network: The History of SNMP Management, Internet Organizations and standards, Internet Documents, The SNMP Model, The Organization Model, System Overview, The Information Model – Introduction, The Structure of Management Information, Managed Objects, Management Information Base **9 Hrs**

Unit-III

SNMPv1 Network Management: Communication and Functional Models: The SNMP Communication Model – The SNMP Architecture, Administrative Model, SNMP Specifications, SNMP Operations, SNMP MIB Group, And Functional Model.

SNMP Management- RMON: Remote Monitoring, RMON SMI and MIB, RMONI1 RMON1 Textual Conventions, RMON1 Groups and Functions, Relationship Between Control and Data Tables, RMON1 Common and Ethernet Groups, RMON Token Ring Extension Groups, RMON2 – The RMON2 Management Information Base, RMON2. **12 Hrs**

Unit-IV

roadband Network Management: ATM Networks: Broadband Networks and Services, ATM Technology – Virtual Path Virtual Circuit, TM Packet Size, Integrated Service, SONET, ATM LAN Emulation, Virtual LAN; ATM Network Management – The ATM Network Reference Model, The Integrated Local Management Interface, The ATM Management Information Base, The Role of SNMP and ILMI in ATM Management, M1 Interface: Management of ATM Network Element, M2 Interface: Management of Private Networks, M3 Interface: Customer Network Management of Public Networks, M4 Interface: Public Network Management, Management of LAN Emulation. ADSL Architecture, ADSL Channeling Schemes, ADSL Encoding Schemes; ADSL Management ADSL Network Management Elements **10 Hrs**

Unit-V

Network Management Tools: Introduction to various network management tools. Server Management: Application Servers, Web Server. Device configuration like Fire wall settings, Hot Spot, Gate Ways, Switches etc. **9 Hrs**

Experiments:

4. Analyze bandwidth throughput etc. for a given network.
5. Analyze different protocols performance like TCP and UDP for a given point-to-point network.
6. Analyze routing protocol performance.

Reference Books:

5. Mani Subramanian, "Network Management Principles and Practice", 2nd Edition, Pearson Education Publication, 2010.
6. Jianguo Ding, "Advances in Network Management", 1st Edition, Auerbach Publications, 2016.
7. Allan Leinwand, Karen Fang Conroy, & Karen Fang, "Network Management: A Practical Perspective", Addison Wesley, 1996
8. Alexander Clemm, "Network Management Fundamentals", Cisco, 1st Edition, 2016.

15UCSE808 Ontology & Semantic Web (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- Theoretical and practical aspects of ontology in semantic web.
- Concepts of RDF and web ontology language.
- Various patterns for developing and reusing ontologies.
- Models for knowledge management architecture in semantic web.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		Mapping to POs(1-12)/ PSOs (13-16)		
		Substantial Level (3)	Moderate Level (2)	Slight Level (1)
CO-1	**Explain** the working principles, architecture of semantic web and the importance of markup languages.	1	-	-
CO-2	**Design** and **analyze** ontologies using web ontology language.	-	3	-
CO-3	**Apply** the logic with web ontology language.	15	-	5
CO-4	**Analyze** scalable architectures and semantic web services.	-	2	-
CO-5	**Explain** the evolution of ontologies with semantic interpretation of information.	-	12	-

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	3.0	2.0	2.0	-	1.0	-	-	-	-	-	-	2.0		-	3.0	-

Pre-requisites: Knowledge of HTML and Java

Contents:

Unit-I

The semantic web vision: Today's web, Semantic web Technologies, A layered approach, Structured web documents in XML, Introduction to RDF, RDF syntax, RDF schema, Axiomatic statements for RDF and RDF schema, Direct inference system on RDF and RDF schema. **9 Hrs**

Unit-II

ntology and Web ontology Language: Introducing OWL, Discovering the Various Species of OWL, Exploring the Foundations of OWL Understanding OWL Essentials, Making Simple Assertions, Inconsistency, Examining Property Characteristics, Complex Classes, Understanding Why OWL Is Different, Developing OWL Ontologies. **8 Hrs**

Unit-III

Ontology building and inference by logic: Monotonic rules - syntax and semantics, Nonmonotonic rules - syntax, Rule markup in XML, Constructing and reusing ontologies, Semantic web knowledge management architecture, Querying using SPARQL, Jena ontology API, Reasoners. **8 Hrs**

Unit-IV

ntology scalable architectures and Discovering the semantic web services: Discovering the Roles, Creating Semantics for Enterprise Systems, Scaling Semantic Web Tools, Patterns of Architectural Usage and Applicationdevelopment using OWL API. **7 Hrs**

Unit-V

Ontology evolution and Semantic knowledge: Ontology population and enrichment, Semantic representation of multimedia content, Ontology based semantics extraction from text and Images. A survey of semantic image and video annotation tools, Overview of linked data. **7 Hrs**

Reference Books:

5. Grigoris Antoniou and Frank van Harmelen, "A Semantic Web Primer", 2nd Edition, MIT Press, Cambridge, MA 2008
6. Jeffrey T.Pollock, "Semantic Web For Dummies", Wiley Publishing, Inc 2009
7. John Hebeler, Matthew Fisher, Ryan Blace, & Andrew Perez-Lopez, "Semantic Web Programming", Wiley Publishing, Inc.
8. Georgios Paliouras, Constantine Spyropoulos, & George Tsatsaronis (Eds.) "Knowledge-Driven Multimedia Information Extraction and Ontology Evolution", Springer, 2011

15UCSE809 Big Data Analytics (3-0-2) 4

Contact Hours: 39T + 26L

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- To understand big data and its applications to various fileds.
- To learn business case studies for big data analytics
- To manage Big data without SQL
- To understand map-reduce analytics using Hadoop and related tools

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the various real time scenarios which demand big data	-	1	-

	and retreival techniques.			
CO-2	**Install** and **configure** Hadoop platform.	5	-	-
CO-3	**Explain** the architecture and working principles of Hadoop Distributed File System (HDFS).	-	1	-
CO-4	**Write** Map-Reduce programs on Hadoop platform.	3	4	-
CO-5	**Perform** map-reduce analytics on different datasets	4,15	-	-
CO-6	**Write** unit and integration tests for the mapper and reducer for different problems	3	4	-
CO-7	**Explain** the working principles and architectures of Apache subprojects such as HBase, Pig, and Hive for big data Analytics	-	1	-

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	2.0	-	3.0	2.3	3.0	-	-	-	-	-	-	-	-	-	3.0	-

Pre-requisites: Knowledge of Database Management Systems

Contents:

Unit-I

Understanding Big Data: Data, Data Storage and Analysis, Comparison with Other Systems – RDBMS, Grid Computing, Volunteer Computing, A Brief History of Hadoop, Applications to various fields.

Map Reduce: Weather dataset, Analyzing the Data with UNIX tools, Analyzing the Data with Hadoop, Scaling out, Hadoop Streaming, Hadoop Pipes. **9T + 4L Hrs**

Unit-II

ıdoop Distributed File System (HDFS): The Design of HDFS, HDFS Concepts – Blocks, Namenodes and Datanodes, HDFS Federation, HDFS High Availability. The Command Line Interface, Hadoop Filesystems, The Java Interface, Data Flow, Parallel Copying with distcp, Hadoop Archives. **8T + 5L Hrs**

Unit-III

Map-Reduce Application Development: The Configuration API, Configuring the Development Environment, Writing a Unit Test, Running Locally on Test Data, Running on a Cluster, Tuning a Job, MapReduce workflows. **8T + 5L Hrs**

Unit-IV

orking Of Map Reduce: Anatomy of a Map Reduce Job, Failures, Job Scheduling, Shuffle and Sort, Task Execution. **7T + 6L Hrs**

Unit-V

Hadoop Related Tools: Hbase, Pig, Hive **7T + 6L Hrs**

Lab Experiments:

1. Programs related to Hadoop operations.
2. Map – Reduce Programming – Basic
3. Map-Reduce Programming – Advanced

Reference Books:

1. Tom White, "Hadoop: The Definitive Guide", Third Edition, O'Reilley, 2012.
2. Eric Sammer, "Hadoop Operations", O'Reilley, 2012. Vignesh Prajapat

15UCSC700 Engineering Management, Enterpreunership, and Intellectual Property Rights (4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- To learn the evolution of management and related aspects like planning, organizing, decision making and staffing.
- To know the scope of entrepreneurship in small, medium and large scale industries
- To identify the issues related to the protection of intellectual property rights.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the dynamics of management in an organization.	9	2	-
CO-2	**Explain** the importance of entrepreneurship in current economy.	6	11	-

CO-3	**Analyze** the legal aspects of intellectual property rights in an organization.	-	8	3
CO-4	**Apply** the concepts of management, entrepreneurship and IPR to real time corporate issues.	-	4	-

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	-	2.0	1.0	2.0	-	3.0	-	-	3.0	-	2.0	-	-	-	-	-

Pre-requisites: Knowledge of Course on Humanities

Contents:

Unit-I

Engineering and Management: Historical Development of Engineering and Management, Management as synthesis.

Planning, Forecasting and Decision Making: Nature of Planning, foundation of planning, some planning concepts, forecasting, nature of decision making, management science, tools for decision-making. **9 Hrs**

Unit-II

Organizing and staffing: Nature of organizing, traditional organizational theory, technology and modern organization structures, staffing technical organization, authority and power; delegation, meeting & committees.
Motivation: Motivation, leadership, motivating and leading technical professionals.
Controlling: Process of control, financial and non-financial controls. **11 Hrs**

Unit-III

Foundations of Entrepreneurship: Meaning of entrepreneur, functions of entrepreneur, types of entrepreneur, concept of entrepreneurship, role of entrepreneurs in economic development& barriers of entrepreneurship.
Small Scale Industry: Definition, characteristics of SSI, role of SSI in economic development, advantages of SSI, steps to start an SSI, impact of liberalization, privatization, and globalization on SSI, definition of ancillary and tiny industry. **10 Hrs**

Unit-IV

Government and Institutional Support: Nature of support from government, objectives and functions of SSI, SIDBI, DIC, single window agency, KIADB, KSSIDC, KSFC.
Preparation of Project: Meaning of project identification, project report, contents and formulation, identification of business opportunities, feasibility studies, types and

purpose. **9 Hrs**

Unit-V

Intellectual Property Rights: Definition, Forms of intellectual property rights, competing rationale for protection, international conventions, world court.

Copyright: Meaning of copyright, content of copy right, ownership and rights, period of copyright, assignment and relinquishment of copyright, license, infringement of copy right, fair use, offenses and penalties.

Patents: Concept of patent, patentable inventions, procedure for obtaining patent, rights and obligations of patent holders, infringements and remedies, offenses and penalties.

Industrial Designs: Definition of design, procedure for registration, rights conferred by registration, infringements, Trademark and related issues. Professional Ethics and ACM code of Conduct **13 Hrs**

Reference Books:

1) N.V.R. Naidu and T. Krishna Rao "Management and Entrepreneurship" International Publishing House Pvt. Ltd, 2008

2) Morse and Bobcock "Managing Engineering and Technology", 5th Edition, PHI, 2013.

3) Thomas W. Zimmerer, "Essentials of Entrepreneurship", 2nd Edition, PHI, 2012

4) N.K.Acharya, "Intellectual Property Rights", 4th Edition, Asia Law House, 2012.

15UCSC702 Computer Graphics (3-0-2) 4

Contact Hours: 39T + 26L

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- To introduce fundamental principles of computer graphics, its architecture and to understand how transformations of objects are carried out.
- To facilitate students to identify the good design principles and to solve the challenges involved in simulating real world objects/conditions using graphics.
- To facilitate the students to learn and apply the aspects of interaction with computer and exposes them to open source tools like OpenGL.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:	**Mapping to POs(1-12)/ PSOs (13-16)**		
	Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**

CO-1	**Apply** the principles of Computer Graphics Architecture to develop graphical applications.	13	14	1,2,12
CO-2	**Use** OpenGL API features and understand the importance of Open Sources (Top-Down Approach for learning).	14	12	1,5
CO-3	**Identify** and **employ** the various implementation strategies for 2D and 3D primitives to solve the challenges involved in simulating the real world objects / conditions using graphics.	14	2	-
CO-4	**Explain** how affine transformations of objects are carried out in graphics and apply the same in simulating real world objects / conditions.	13	14	1
CO-5	**Discriminate** the views of objects in parallel and perspective projections in various lighting conditions.	13	-	-
CO-6	**Simulate** various raster graphics algorithms.	13,14	-	1,12

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.0	1.5	-	-	1.0	-	-	-	-	-	-	1.3	3.0	2.3	-	-

Pre-requisites: Knowledge of

- Basic Engineering Graphics
- Linear Algebra (Scalars, Vectors, Matrices)
- Algorithms and C programming.

Contents:

Unit-I

Introduction : Applications of computer graphics; A graphics system; Images: Physical and synthetic; Imaging systems; The synthetic camera model; The programmer's interface; Graphics architectures; Programmable pipelines; Performance characteristics. Graphics Programming: The Sierpinski gasket; Programming two dimensional applications

Input and Interaction: Interaction; Input devices; Clients and servers; Display lists;

Display lists and modeling; Programming event-driven input; Menus; Picking; A simple CAD program; Building interactive models; Animating interactive programs; Design of interactive programs; Logic operations. **9T + 5L Hrs**

Unit-II

Geometric Objects and Transformations: Scalars, points, and vectors; Coordinate systems and frames; Modeling a colored cube; Affine transformations; Rotation, translation and scaling; Transformations in homogeneous coordinates; Concatenation of transformations; Interfaces to three-dimensional applications. **9T + 6L Hrs**

Unit-III

Viewing and Lighting : Classical and computer viewing; Viewing with a computer; Positioning of the camera; Simple projections; Projections in OpenGL; Hidden surface removal; Parallel-projection matrices; Perspective-projection matrices; Lighting and Shading: Light and matter; Light sources; The Phong lighting model. **7T + 5L Hrs**

Unit-IV

Basic Raster Graphics Algorithms for drawing 2D primitives : Scan converting lines, circles, Filling Rectangles, Polygons; Clipping in a raster world; Clipping lines, polygons; Anti-aliasing **7T + 5L Hrs**

Unit-V

Representing Curves and Surfaces: Parametric Cubic Curves – Hermite Curves, Bézier Curves. **7T + 5L Hrs**

Conduction of Practical Sessions: Practical Sessions to be held with the focus of learning Open Source Tools like OpenGL and its API features. For the successful completion of the course, students are expected to undertake project to explore advanced features of Open Source Tools like OpenGL.

Reference Books:

6) Edward Angel, “Interactive Computer Graphics A Top-Down Approach with OpenGL”, 5th Edition, Addison-Wesley, 2008

7) James D Foley, Andries Van Dam, Steven K Feiner & John F Hughes, “Computer Graphics Principles and Practice”, 2nd Edition, Addison-Wesley, 1997.

8) Edward Angel & Dave Shreiner, “Interactive Computer Graphics A Top-Down Approach with Shader-Based OpenGL”, 6th Edition, Addison-Wesley, 2012

9) F.S. Hill, Jr., “Computer Graphics Using OpenGL”, 2nd Edition, Pearson Education, 2005

10) Donald Hearn and Pauline Baker, “Computer Graphics- OpenGL Version”, 2nd Edition, Pearson Education, 2003

15UCSL703 Major Project – Phase 1 (0-0-8) 4

Contact Hours: 96

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- Understand the domain, Analyze through Modeling and Implementation through state of the art technology available.
- Know Software Engineering Principles: Modeling, Estimation, Design standards and architectural issues through use of Standards etc.
- To write modular programs and handle exceptions to provide reliable solutions.
- To test and verify the programs for different scenarios.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Identify** and **formulate** the problem.	1,2,12	8	-
CO-2	**Analyze** the problem scenario and **Design** the solutions to complex engineering problem using software engineering principles or appropriate research methodology.	3,5,12,13	8,11	6,7,16
CO-3	**Identify** and **Implement** a feasible solution using appropriate technology, tools, procedures and techniques.	4,5,12,14	8	16
CO-4	**Verify** and **Validate** the proposed system for correctness and to demonstrate compliance with the design and hence the stated requirements/ research gap.	5,12,15	8	-
CO-5	**Prepare** the report and **communicate** effectively through presentation.	9,10	8	-

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	3.0	3.0	3.0	3.0	3.0	1.0	1.0	2.0	3.0	3.0	2.0	3.0	3.0	3.0	3.0	1.0

Prerequisites: Different programming languages / tools.

A project may belong to any of the following categories:

1. Learning Centric: Here the output of the project activity is enhancement of the student's knowledge. Here the student chooses a work with the intension to gain the knowledge.

2. Application: Traditional Software Engineering project, with appreciable complexity. Scenario of demonstration had to be made clear and Completeness is important.

3. Product Base: The final output is a novel one which may be the assembly of several existing subsystems. Completeness is important till the user manual.

4. Research Oriented: A research problem of student's interest. Achievement would be a publication in the IEEE/ACM international/national conferences.

General Instructions to Students:

- Students are expected to perform extensive literature survey, identify problem statements and prepare synopsis in consultation with project guide/supervisor. Students are expected to submit Synopsis- Initial (Registration Phase-1) approved by project guide, to the project coordinator as per the schedule notified. A copy is to be maintained with students and the guide. This registration/ Initial synopsis contains the description of the project concept created and acts as a base line for design and Implementation of the system.
- Notification/schedules and evaluation procedures will be sent to all students in the Google groups created in the department.
- **PHASE-1** Project: Evaluation of problem statement/synopsis-Initial (registration phase-1), SRS (Requirement Analysis Phase-2) and Design Details (Design Details Phase-3) are done in the 7th semester. It is necessary to complete the implementation of few use cases of the complete design and present it during semester end examination (SEE) of project (7th semester);
- **PHASE-2** Project: During the early part of 8th semester it is expected that students complete the implementation work (Implementation and Testing-phase-4) of all planned use cases based on PHASE-1 of the project along with possible publication and knowledge transfer through conduction of training program on various technology used. Semester End Examination in 8th semester is fully based on Phase-1 to Phase-4.

- Batch size is of maximum 4 students. Mixing of divisions is not permitted unless it is the sponsored/research project and request is initiated by project supervisor/guide.
- Students may approach the faculty members of CSE department for choosing them as project guide/supervisor.
- A teacher can guide/supervise maximum of 2 UG project groups. However in special case, a DUGC (Project Coordinating Team, **PCT**, working on the behalf of DUGC) may assign additional project groups to a given teacher.
- The department will financially support presentations of publication of paper only for reputed conference publications.
- DUGC/Project coordinating team (**PCT**) may be consulted for any other/ missing information.

Evaluation Procedures:

l) CIE Marks for the project is to be awarded by project guide/supervisor and SEE marks are to be awarded by examiners (appointed by DUGC) and PCT based on the overall guidelines and project classifications/rubrics by looking into Software Engineering aspects & usefulness w.r.t research/innovation/technology/industry trends through formal interactions and presentations. However, each team is expected to give a formal complete presentation of their work at the end of each phase (1 to 4) to project coordinating team (PCT).

m) PCT evaluates the work and suggest the corrections and observation. All project teams are expected to incorporate these changes in their work. These observations will be made available to Project guides and SEE examiners, which will help them to evaluate and award marks during assessment process.

n) Project guides should keep track of all interactions they do with project team members on weekly basis.

o) All Projects are evaluated and individual students are awarded a grade based on the grading criteria set.

p) Individuals' grade/marks is decided based on both CIE and SEE marks/grade.

q) A project is considered for possible award of S grade; if and only if its 'research outputs' / 'product innovation outputs' results in to a publication of a paper. In a special case, innovative or sponsored business applications with focus on recent technological trends/ Industry trends catering for societal needs may be considered. If contents of paper/research output are not at satisfactory level, then, the assessment results in to appropriate lower grades.

r) A project is considered for possible award of A grade; if and only if it demonstrate product development skills in core system or systems level applications using all aspects of software engineering product development

phases like: requirements, design, implementation, testing including standards like: use of design/architectural patterns, coding standards, use of tools for design/testing, programming practices, documentation and reporting etc..,. In a special case, innovative or sponsored business applications with focus on recent technological trends / Industry trends catering for societal needs may be considered.

s) If project work contribution/ output is not at satisfactory level, then the assessment results @ appropriate lower grades. All grades/marks are awarded based on individual contributions evaluated from software engineering perspectives **specified in the rubrics**.

t) Expected important features:

- Report preparation using latex.
- Online plagiarism check report is to be enclosed in the report.
- Use of IEEE standard. Ex: reference listing and use of PPTs for presentation etc…
- Use of software tools. Ex: for Design, version control, UI design, Testing etc…
- Preparation of video on project, to be uploaded on YouTube.
- Conduction of workshop/ training on technology/domain to students and Preparation of training material/manual (.doc & .pdf).
- Publication of paper based on outcome of the project.
- Submission of proposal to KSCST (Govt. of Karnataka) or other agencies for funding.
- Any other features suggested by guides/coordinators from time to time.

u) Marks Weightage and Various parameters for project evaluation for both CIE and SEE level @ 7th and 8th Semester.

Sl.No.	Parameter for Assessment	% Weight For CIE and SEE	
		7th Sem	8th sem
1	Requirements Analysis (SRS): Abstract and Detailed.	20	10
2	Design Specification; Use of: UML diagrams, architecture diagram, ER diagram, Patterns etc… Proper cohesiveness and coupling of various components in the system design.	35	10
4	Use of Tools and standards.	5	5
5	Implementation: Code documentation, style, robustness, maintainability, Testability, Usability (User Experience) etc…	10	20
6	Testing: for every scenario of all use cases	10	10

	identified.		
7	Final Oral Presentation (viva-voce) (IEEE Standards for slides, oral presentation techniques etc…)	10	10
8	Project Reports- Final and Intermediate if any: preparation using LATEX and plagiarism check	10	20
9	Preparation of Learning Materials (.doc & .pdf) & Videos: Uploading through Department Channel on YouTube. Knowledge Transfer: through conduction of workshop and training programs	-	10
10	Publication of paper. (@IEEE or equivalent or higher)	-	5

Note:

3. Sufficient and completeness of each parameter is to be seen while awarding marks for individual students.
4. Marks for individual students in a given project team may vary based on individuals 'learning outcomes.

- All project teams are expected to participate in the project exhibition arranged at department level. Project teams are expected to share their project experience to all their juniors and motivate them to take-up challenging work as their project work. During project exhibition, Top 2 projects from the batch will be awarded with a certificate of appreciation at the end of academic year.

15UCSE705 Software Testing (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- Notations and terminologies used in software testing.
- Principles of programming, Software Development Process from the perspective of quality, maintenance, testing, programming style, system quality through: requirements, design, coding, verification and validation procedures.
- Various testing strategies and use of industry relevant software tools.
- Applications, case studies and use of tools / programming techniques for design and implementation of test procedure / cases for real life application.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Map** inputs to the expected outputs of the system by understanding the system behavior represented in the form of: requirements and design specifications / Models.	13	-	-
CO-2	**Design** test cases based on requirement specifications.	2	1	-
CO-3	**Design** test cases based on traditional design specifications.	3	-	-
CO-4	**Design** test cases based on Object specifications/Models (UML).	-	3	-
CO-5	**Write** script in any programming Language/Tools to implement test cases designed.	-	15	-
CO-6	**Design** test cases based on various testing strategies to check the correctness of computer program.	15	-	-

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	3.0	2.5	-	-	-	-	-	-	-	-	-	3.0	-	2.5	-

Pre-requisites: Knowledge of

- Programming language
- Software Engineering
- Object Oriented System Analysis and Design

Contents:

Unit-I

Programming Practice: A Testing Perspective: Programming Principles & Guidelines, Coding Processes, Refactoring, Verifications & Metrics, Style- Names, Expression and Statements, Consistency and Idioms, Function Macros, Debugging, Performance, Portability

Basics of Software Testing: Human Errors and Testing, Software Quality, Requirements, Behavior and Correctness, Correctness versus Reliability, Testing and Debugging, Test Metrics, Software and Hardware Testing, Testing and Verification, Defect Management, Execution History, Test generation Strategies, Static Testing. Model-Based Testing and Model Checking, Control-Flow Graph, Types of Testing, The Saturation Effect. **9 Hrs**

Unit-II

Test Generation from Requirements: Introduction; The Test-Selection Problem; Equivalence Partitioning; Boundary Value Analysis' Category-Partition Method, Cause-Effect Graphing, Test Generation from Predicates. **7 Hrs**

Unit-III

Dependence, Data Flow Models and Data Flow Testing: Definition-Use pairs; Data flow analysis; Classic analyses; From execution to conservative flow analysis; Data flow analysis with arrays and pointers; Inter-procedural analysis; Overview of data flow testing; Definition- Use associations; Data flow testing criteria; Data flow coverage with complex structures; The infeasibility problem. **7 Hrs**

Unit-IV

Structural Testing: Overview; Statement testing; Branch testing; Condition testing, Path testing; Procedure call testing; Comparing structural testing criteria; The infeasibility problem.

Test Case Selection and Adequacy Test Execution: Overview; Test specification and cases; Adequacy criteria; Comparing criteria; Overview of test execution; From test case specification to test cases; Scaffolding; Generic versus specific scaffolding; Test oracles; Self-checks as oracles; Capture and replay **9 Hrs**

Unit-V

Testing Object Oriented Software: Issues in Testing OO Software, Intra Class Testing, Testing with State Machine Models, Inter-Class Testing, Structural Testing of Class **7 Hrs**

Reference Books:

6) Brain W. Kernighan & Rob Pike, "The Practice of Programming", Pearson education, 2008.
7) Pankaj Jalote & Narosa, "An Integrated Approach to Software Engineering", 3rd Edition, Publishing House.
8) Edward Kit, "Software Testing in the Real World", Pearson Education, 2006
9) Aditya P Mathur, "Foundations of Software Testing", Pearson Education, 2008.
10) Mauro Pezze, Michal Young, John Wiley & Sons, "Software Testing and Analysis: Process, Principles and Techniques", 2008.

15UCSE706 Adhoc Networks (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- Understand the need for ad hoc networks.
- Protocol for ad hoc networks.
- Understand security issues and QoS requirements.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the design issues of protocol stack.	13	3	-
CO-2	**Compare** the architecture and working of various MAC layer protocols.	13	16	1,14
CO-3	**Apply** the knowledge of different routing mechanisms for the better routing decisions in WANETs.	3,13	16	15
CO-4	**Apply** the knowledge of different transport layer protocols for ensuring reliable communication in WANETs.	3,13	16	15
CO-5	**Explain** the challenges in security	13	2	1,16

	and QoS issues in WANETs and **suggest** suitable solutions for a given context.			

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	1.0	2.0	2.7	-	-	-	-	-	-	-	-	-	3.0	1.0	1.0	1.8

Pre-requisites: Knowledge of

- Data Communications
- Computer Networks
- C programming

Contents:

Unit-I

Ad hoc Networks: Introduction, Issues in Ad hoc wireless networks, Ad hoc wireless internet.

MAC–1: MAC Protocols - Introduction, Issues in designing a MAC protocol, Design goals of a MAC protocol. **9 Hrs**

Unit-II

MAC–2: Classification of MAC protocols, Contention based protocols with reservation mechanisms, Contention based MAC protocols with scheduling mechanism, MAC protocols that use directional antennas, Other MAC protocols. **7 Hrs**

Unit-III

Routing: Hybrid routing protocol, Routing protocols with effective flooding mechanisms, Hierarchical routing protocols, Power aware routing protocols.

Transport Layer: Transport layer - Introduction, Issues in designing a transport layer protocol, Design goals of a transport layer protocol, Classification of transport layer solutions, TCP over Ad hoc wireless Networks, Other transport layer protocols. **9 Hrs**

Unit-IV

Security: Introduction, Network security requirements, Issues & challenges in security provisioning, Network security attacks, Key management, Secure routing. **7 Hrs**

Unit-V

QoS: Introduction, Issues and challenges in providing QoS, Classification of QoS solutions, MAC layer solutions, network layer solutions. **7 Hrs**

Reference Books:

5) C. Siva Ram Murthy & B. S. Manoj, “Adhoc Wireless Networks”, 2nd Edition, Pearson Education, 2005.
6) Ozan K. Tonguz and Gianguigi Ferrari & John Wiley, “Ad hoc Wireless Networks”, 2006.
7) Xiuzhen Cheng, Xiao Hung, Ding Zhu Du, & Kluwer, “Ad hoc Wireless Networking”, Academic Publishers, 2004.
8) C.K. Toh, “Adhoc Mobile Wireless Networks, Protocols and Systems”, Prentice Hall PTR, 2007.

15UCSE707 Operations Research (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- Apply the fundamental techniques (graphical method, simplex method, dual simplex method) of operations research to solve real world problems.
- Estimate the transportation and assignment costs.
- Design, code, test, and debug programs to solve problems in the domain of operations research.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Formulate** real-world problems in the form of mathematical model and **solve** using graphical method.	-	2	1
CO-2	**Formulate** real-world problems in the form of mathematical model and **solve** using simplex and artificial variables techniques.	-	1,2,3	-
CO-3	**Solve** LP problems using dual simplex method and **perform** sensitivity analysis.	-	1,2,3	16
CO-4	**Build** and **solve** Transportation Models and Assignment Models.	-	1,2	5,16
CO-5	**Build** and **solve** game theory problems.	-	1,2	5

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping	1.8	2.0	2.0	-	1.0	-	-	-	-	-	-	-	-	-	-	1.0

Level																

Pre-requisites: Knowledge of

- Algebra
- Probability

Contents:

Unit-I

Introduction: Definitions and various phases of operations research, Role of computers in operations research, Formulating a problem as a mathematical model in the form of linear programming (LP) problem – Minization and Maximization Types, Solving LP problems using Graphical method. **7 Hrs**

Unit-II

Simplex Method: Canonical and Standard forms of LP problem, The simplex method – examples, Artificial variables Techniques (Big M and Two Phase method) - examples. **7 Hrs**

Unit-III

Advanced Topics in Linear Programming: Duality, Dual Simplex Method, Economic interpretation of duality, Sensitivity analysis, Dynamic Programming. **9 Hrs**

Unit-IV

Transportation Model: Introduction, Formulation, Methods for Initial Basic Feasible solution, Improving the basic feasible solution using Stepping Stone and MODI methods.

Assignment Model: Introduction, Formulation, Comparison with transportation model, Hungarian method. **9 Hrs**

Unit-V

Game Theory: Introduction, Formulation, Strategies – Pure and Mixed, Methods for solving Game theory problems – Saddle point, Rule of Dominance, Arithmetic method, Graphical method. **7 Hrs**

Reference Books:

5) Er. Premakumar Gupta and Dr. D.S.Hira, “Operations Research”, S Chand Publications, 2014.
6) Frederick S. Hillier and Gerald J. Lieberman, “Introduction to Operations Research”, 8th Edition, Tata McGraw Hill, 2005.
7) Wayne L. Winston, “Operations Research Applications and Algorithms”, 4th Edition, Thomson Course Technology, 2003.

8) Hamdy A T, “Operations Research: An Introduction”, 9th Edition, Pearson Publishers, 2014

15UCSE708 Internet of Things (2-0-4) 3

Contact Hours: 39 (26T + 26L)

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- Understand the basics of Internet of Things technology, its communication protocols, sensor networks, and its applications.
- Focus on setting up of IoT ecosystem to implement use cases by applying the key concepts of IoT.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the working of IoT and its enabling technologies.	-	5	1,12
CO-2	**Explain** the use cases of IoT and **understand** the use of sensors and actuators in IoT ecosystem.	-	5	1,12
CO-3	**Tabulate** the differences and similarities between IoT and M2M and **understand** the generic design methodology for IoT system.	-	5,13	1,12
CO-4	**Develop** simple IoT applications using Arduino by making use of appropriate sensors and modules.	13	14	1,5,9,15
CO-5	**Develop** simple IoT applications using Raspberry Pi microcontroller by making use of appropriate sensors and modules.	13	14	1,5,9,15

POs/PSOs	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Mapping Level	1.0	-	-	-	1.6	-	-	-	1.0	-	-	1.0	2.7	2.0	1.0	-

Pre-requisites: Knowledge of Programming and Basics of Computer Networking

Contents:

Unit-I

Introduction to Internet of Things (IoT): Introduction, Physical and Logical Design of IoT, IoT Enabling Technologies, IoT levels and Deployment templates **5L + 4P Hrs**

Unit-II

Domain Specific IoTs: Introduction, Home Automation, Cities, Environment, Energy, Retail, Logistics, Agriculture, Industry, Health &Lifestyle

IoT Sensors and Actuators: Introduction, IoT sensors, RFID, Video Tracking, IoT Actuators **4L + 4P Hrs**

Unit-III

IoT and M2M: Introduction, M2M, Difference between IoT and M2M, SDN and NFV for IoT

IoT Platforms Design Methodology: Introduction, IoT Design Methodology, Case Study on IoT System for Weather Monitoring **7L Hrs**

Unit-IV

Arduino Programming: Features of Arduino, Arduino UNO, Arduino IDE, Arduino sketch structure, Arduino function libraries, Blinking LED example; Operators, control statements, loops, arrays, string, interrupts, Traffic Control System example

Integration of Sensors, Actuators and modules with Arduino: Sensor Types, Sensor Interface with Arduino, Interfacing DHT, LM35, LDR, Accelerometer and HC-SR04 sensors with Arduino; Actuators, Types of Motor Actuators, Servo motor, Servo library on Arduino; HC05 Bluetooth module, Features, Interfacing HC05 with Arduino, Controlling LED using Bluetooth; ESP8266 wifi module, Features, Send LM35 sensor data to cloud using wifi module **5L + 9P Hrs**

Unit-V

Raspberry Pi: Introduction, Specifications, Basic Architecture, Pin configuration, Blinking LED example; Capture image using Raspberry Pi

Implementation of IoT with Raspberry Pi: Temperature dependent auto cooling system; interfacing DHT, LM35, LDR, Accelerometer and HC-SR04 sensors with Raspberry Pi; Send LM35 sensor data to cloud using wifi module **5L + 9P Hrs**

Reference Books:

1) Arshdeep Bahga, Vijay Madisetti ,“Internet of Things – A Hands-on Approach”,

Universities Press, 2015
2) Ammar Rayes, Samer Salam, “Internet of Things From Hype to Reality – The Road to Digitalization”, Second Edition, , Springer Nature Switzerland AG 2017, 2019
3) NPTEL course on “Introduction to internet of things” by Prof. Sudip Misra

15UCSE709 Multi Core Architecture and Programming (3-0-0) 3

Contact Hours: 39

Course Learning Objectives (CLOs): This course focuses on following learning perspectives:

- Basic concepts in multicore architecture.
- Synchronization and coordination mechanisms available on latest multicore machines.
- Effective concurrent program writing to enhance the performance for windows, C# & .net and Linux programming platforms.
- Introductory knowledge in Open MP libraries and pthreads.

Course Outcomes (COs):

Description of the Course Outcome: At the end of the course the student will be able to:		**Mapping to POs(1-12)/ PSOs (13-16)**		
		Substantial Level (3)	**Moderate Level (2)**	**Slight Level (1)**
CO-1	**Explain** the need for multicore architecture for modern day applications	1	2	14
CO-2	**Write** a program using pthread libraries and Open MP features to solve problems that needs thread models.	5	4	13
CO-3	**Write** a program to solve problems using parallel programming constructs.	4	3	-
CO-4	**Design** a solution for the problem using APIs for Win32, MFC and .NET;	5	-	-
CO-5	**Apply** standard solutions to some common parallel programming problems like data Race conditions, Dead locks,Live locks, Memory Issues etc.	2	3	1

POs/PSOs	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Mapping Level	2.0	2.5	2.0	2.5	3.0	-	-	-	-	-	-	-	1.0	1.0	-	-

Pre-requisites: Knowledge of

- Microprocessor
- Operating Systems
- C programming

Contents:

Unit-I

Introduction to Multi-core Architecture: Motivation for Concurrency in software, Parallel ComputingPlatforms, Parallel Computing in Microprocessors, Differentiating Multi-core Architectures from Hyper-Threading Technology, Multi-threading on Single-Core versus Multi-Core Platforms UnderstandingPerformance, Amdahl's Law, Growing Returns: Gustafson's Law.

System Overview of Threading: Defining Threads, System View of Threads, Threading above the Operating System, Threads inside theOS, Threads inside the Hardware, What Happens When a Thread Is Created, Application ProgrammingModels and Threading, Virtual Environment: VMs and Platforms, Runtime Virtualization, System Virtualization. **8 Hrs**

Unit-II

Fundamental Concepts of Parallel Programming: Designing for Threads, Task Decomposition, Data Decomposition, Data Flow Decomposition, Implications of Different Decompositions, Challenges You'll Face, Parallel Programming Patterns, A Motivating Problem: Error Diffusion, Analysis of the Error Diffusion Algorithm, An Alternate Approach: Parallel Error Diffusion, Other Alternatives. **8 Hrs**

Unit-III

Threading and Parallel Programming Constructs: Synchronization, Critical Sections, Deadlock, Synchronization Primitives, Semaphores, Locks, Condition Variables, Messages, Flow Control- based Concepts, Fence, Barrier, Implementation-dependent Threading Features. Threading APIs: Threading APIs for Microsoft Windows, Win32/MFC Thread APIs, Threading APIs for Microsoft. NET Framework, Creating Threads, Managing Threads, Thread Pools, Thread Synchronization, POSIX Threads, Creating Threads, Managing Threads, Thread Synchronization, Signaling, Compilation and Linking. **8 Hrs**

Unit-IV

OpenMP: A Portable Solution for Threading: Challenges in Threading a Loop,

Loop-carried Dependence, Data-race Conditions, Managing Shared and Private Data, Loop Scheduling and Portioning, Effective Use of Reductions, Minimizing Threading Overhead, Work-sharing Sections, Performance oriented Programming, Using Barrier and No wait, Interleaving Single-thread and Multi-thread Execution, Data Copy-in and Copy-out, Protecting Updates of Shared Variables, Intel Task queuing Extension to OpenMP, OpenMP Library Functions, OpenMP Environment Variables, Compilation, Debugging, performance. **8 Hrs**

Unit-V

Solutions to Common Parallel Programming Problems: Too Many Threads, Data Races, Deadlocks , and Live Locks, Deadlock, Heavily Contended Locks, Priority Inversion, Solutions for Heavily Contended Locks, Non-blocking Algorithms, ABA Problem, Cache Line Ping-ponging, Memory Reclamation Problem, Recommendations, Thread-safe Functions and Libraries, Memory Issues , Bandwidth, Working in the Cache, Memory Contention, Cache-related Issues, False Sharing, Memory Consistency. **7 Hrs**

Reference Books:

5. Shameem Akhter & Jason Roberts “Multicore Programming Increased Performance through Software Multi-threading”, Intel Press, 2006
6. The Software Optimization Cookbook – Intel Press 2007
7. M.Shyamala Devi, “Multicore Architectures and Programming”, Sahara Publications, 2016
8. Krishna Sankar P & Shangarnarayanee N P, “Multi-Core Architectures and Programming”, AR Publications, 2016.

PG:

Name of the Course	Course Code	Activities/Content with direct bearing on Employability/ Enterpereneurship /Skill development	Year of Introduction (during the last five years)	Link to the relevent document
Artificial Intelligence and Machine Learning	20PCSEC102	**Employability/ Skill development/**	2020	
Block Chain Technology	20PCSEE126	**Employability/ Skill development**	2020	
Internet of Things	20PCSEE225	**Employability/ Skill development**	2020	
Machine Learning	18PCSEE227	**Employability/ Skill development/**	2018	
Internet of Things	18PCSEE231	**Employability/ Skill development**	2018	
Machine Learning	16PCSEE232	**Employability/ Skill development**	2016	
Internet of Things	16PCSEE235	**Employability/ Skill development/**	2016	

	Year 1 21-22	Year 2 20-21	Year 3 19-20	Year 4 18-19	Year 5 17-18
New courses	3	3	2	2	2
Total courses	27	20	24	20	28
Percentage	11.11%	15%	8.3%	10%	7.1%

20PCSEC102 Artificial Intelligence and Machine Learning (4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs):

Objective of this course is to make student knowledge-full enough to determine when an AI approach is appropriate for a given problem, identify the appropriate representation, reasoning mechanism, models, algorithms, implement and evaluate it.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs(1 to 6)		
		Substantial Level (3)	Moderate Level (2)	Slight Level (1)
CO-1	**Understand** the world, behavior of agents and problem solving aspects of agents.	-	3,4	-
CO-2	**Demonstrate** the representation and usage of knowledge using First order logic.	-	3,4	-
CO-3	**Understand** the decision making process even with incomplete, inconsistent and ever changing facts.	-	3,4	-
CO-4	**Understand** the strengths and weaknesses of many popular machine learning approaches with awareness to SVM and Neural Networks in machine learning.	-	3,4	-
CO-5	**Understand** and **apply** unsupervised algorithms for clustering, able to interpret appropriateness of among the three learning styles and performance of a simple learning system on a real-world dataset.	-	3,4,5	-

Mapping Level:

POs	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
Mapping Level	-	-	2	2	2	-

Prerequisites: Knowledge of Logic, Discrete Mathematic, Programming fundamentals.

Contents:

1. **What is AI (Artificial Intelligence)?** : The AI Problems, The Underlying Assumption, What are AI Techniques, The Level Of The Model, Criteria For Success, Some General References, One Final Word Problems, State Space Search & Heuristic Search Techniques: Defining The Problems As A State Space Search, Production Systems, Production Characteristics, Production System Characteristics, And Issues In The Design Of Search Programs, Additional Problems. Generate And-Test, Hill Climbing, Best-First Search, Problem Reduction, Constraint Satisfaction, Means Ends Analysis.

   10 Hrs
2. **Knowledge Representation Issues:** Representations And Mappings, Approaches To Knowledge Representation. Using Predicate Logic: Representation Simple Facts In Logic, Representing Instance And Isa Relationships, Computable Functions And Predicates, Resolution. Representing Knowledge Using Rules: Procedural Versus Declarative Knowledge, Logic Programming, Forward Versus Backward Reasoning.
   10 Hrs
3. **Symbolic Reasoning Under Uncertainty**: Introduction To No monotonic Reasoning, Logics For Non-monotonic Reasoning. Statistical Reasoning: Probability and Bays' Theorem, Certainty Factors And Rule-Base Systems, Bayesian Networks, Dempster Shafer Theory **10 Hrs**
4. **Introduction to Statistical Decision Theory** - Regression, Classification, Bias Variance Linear Regression, Multivariate Regression, Subset Selection, logistic regression Linear Models for Classification, Decision Trees, Regression Trees
   11 Hrs
5. **Perceptron, Support Vector Machines, Neural Networks** - Introduction, Early Models, Perceptron Learning, Backpropagation, Initialization, Training & Validation, Parameter Estimation - MLE, MAP, Bayesian Estimation
   Unsupervised learning and clustering – k-means clustering, hierarchical clustering
   11 Hrs

Reference Books:

1. Elaine Rich and Kevin Knight "Artificial Intelligence", 2nd Edition, Tata Mcgraw-Hill, 2005.
2. Stuart Russel and Peter Norvig, "Artificial Intelligence: A Modern Approach", 3rd Edition, Prentice Hall, 2009.
3. The Elements of Statistical Learning, by Trevor Hastie, Robert Tibshirani, Jerome H. Friedman (freely available online)
4. Pattern Recognition and Machine Learning, by Christopher Bishop (optional)
5. Mitchell Tom, Machine Learning. McGraw Hill, 1997

20PCSEE126 Block Chain Technology (4-0-0) 4

Contact Hours: 52

Course Learning Objectives (CLOs):

This course focuses on understanding emerging abstract models for Blockchain Technology and familiarizes the functional/operational aspects of crypto currency ecosystem. Identify major research challenges and technical gaps existing between theory and practice in crypto currency domain.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs (1 to 6)		
		Substantial level(3)	Moderate level (2)	Slight level (1)
CO-1	**Demonstrate** the basics of Block chain concepts using modern tools/technologies.	-	3, 4, 5	1
CO-2	**Analyze** the role of block chain applications in different domains including cyber security.	-	3, 5	-
CO-3	**Evaluate** the usage of Block chain implementation/features for the given scenario.	-	1, 3, 4	-
CO-4	**Exemplify** the usage of bitcoins and its impact on the economy.	3	-	-
CO-5	**Analyze** the application of specific block chain architecture for a given problem	3	2	6

Mapping Level:

POs	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
Mapping Level	1.5	2	2.4	2	2	1

Prerequisites: Programming and Data Structures

Contents:

1. Introduction to Blockchain, How Blockchain works, Blockchain vs. Bitcoin, Practical applications, public and private key basics, pros and cons of Blockchain, Myths about Bitcoin **10 Hrs**
2. Blockchain: Architecture, versions, variants, use cases, Life use cases of Blockchain, Blockchain vs. shared Database, Introduction to crypto currencies, Types, Applications. **11 Hrs**
3. Concept of Double Spending, Hashing, Mining, Proof of work. Introduction to Merkel tree, Privacy , payment verification , Resolving Conflicts , Creation of Blocks **10 Hrs**
4. Introduction to Bitcoin, key concepts of Bitcoin, Merits and De Merits Fork and Segwits, Sending and Receiving bitcoins, choosing bitcoin wallet, Converting Bitcoins to Fiat Currency **10 Hrs**
5. Introduction to Ethereum, Advantages and Disadvantages, Ethereum vs. Bitcoin, Introduction to Smart contracts, usage, application, working principle, Law and Regulations. Case Study **11 Hrs**

Reference Books:

1. Beginning Blockchain: A Beginner's Guide to Building Blockchain Solutions by Arshdeep Bikramaditya Signal, Gautam Dhameja (Priyansu Sekhar Panda., APress.)
2. Blockchain Applications: A Hands-On Approach by Bahga, Vijay Madisetti
3. Blockchain by Melanie Swan, OReilly.
4. Bitcoin and Cryptocurrency Technologies by Aravind Narayan. Joseph Bonneau, princton
5. Bitcoin and Blockchain Basics: A non-technical introduction for beginners by Arthu.T Books

20PCSEE225	Internet of Things	(3-0-2) 4

Contact Hours: 39+13

Course Learning Objectives (CLOs):

This course provides insights on IoT architecture, communication protocols, sensor networks and applications of IoT. It addresses security and privacy challenges faced by IoT. It also focuses on setting up of an IoT ecosystem to implement use cases by applying the key concepts of IoT.

Course Outcomes (COs):

CO	Description of the Course Outcome: At the end of the course, the student will be able to:	Mapping to POs (1 to 6)		
		Substantial level(3)	Moderate level(2)	Slight level(1)
CO-1	**Describe** IoT service-oriented architecture and its components.	-	-	3, 6
CO-2	**Demonstrate** the use of IoT protocols for developing applications related to smart spaces.	-	1	3, 6
CO-3	**Explain** WSN and UAV (Unmanned Aerial Vehicles) network architecture for interconnectivity and communication among heterogeneous IoT devices.	-	-	3, 6
CO-4	**Explore** the challenges faced in IoT with respect to privacy and security and **provide** theoretical/practical solutions.	2	1, 5	3, 6
CO-5	**Set up** an IoT ecosystem and **implement** various use cases by applying the key concepts of IoT.	1	3	6

Mapping Level

PO	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
Mapping Level	2.34	3.0	1.2	-	2.0	1.0

Prerequisites: Knowledge of block chain technology.

Contents:

1. **Introduction to IoT and networking:** What is IoT; IoT overview; Applications of IoT; Sensing - what is a sensor and transducer, sensor types; Actuation - what are actuators, actuator types, IoT networking: Basics of IoT networking; components of IoT; service-oriented architecture of IoT. **7 L+ 4 P**
2. **IoT Protocols:** IoT Protocols- MQTT, SMQTT, CoAP, XMPP, AMQP; Communication Protocols- IEEE 802.15.4, ZigBee, 6LoWPAN, HART, Bluetooth, Zwave, ISA 100.11 A **7 L+ 4 P**
3. **WSNs and UAV (Unmanned Aerial Vehicles) Networks:** WSNs - what is wireless sensor network, concepts and challenges, node behavior; Applications of WSNs - Target/Object Tracking, Agriculture; UAV Networks - features, constraints, advantages, topologies.
   M2M Communication and Interoperability in IoT: M2M communication - concepts, features, applications, node types, M2M area management; Interoperability in IoT - challenges, types of interoperability - syntactic and device interoperability. **7 L+6 P**
4. **Internet of Things Security & Privacy:** Introduction, IoT Security Challenges, IoT Security Requirements, IoT Three-Domain Architecture, Cloud Domain Attacks and Countermeasures, Fog Domain Attacks and Countermeasures, Sensing Domain Attacks and Countermeasures, Future Directions.
   The Blockchain in IoT: Consensus Algorithms in IoT, Blockchain Applications in IoT, M2M Transactions, Energy Management, Supply Chain Management, Healthcare, Retail, Automotive and Transportation, Smart City, Identity, Authentication, and Access Management, Other Blockchain IoT Applications, Blockchain Security in IoT, Trust Between Nodes, Malicious Activity and Cryptographic Principles, IoT Security and Blockchain Advantages **9 L+ 6 P**
5. **Implementation of IoT with Raspberry Pi:** Raspberry Pi- architecture, components, blinking LED, image processing with Raspberry Pi; Interfacing various sensors with Raspberry Pi.
   Demonstration of blockchain technology: Ethereum IDE, Blockchain demo Anders, writing smart contracts using solidity language, Building DApp for blockchain, Use cases of IoT using blockchain. **9 L+ 6 P**

List of Assignments:

1. Write python program to collect and display sensor data (temperature, humidity, LDR, ultrasonic) from Raspberry Pi microcontroller on the console.
2. Write python program to send the sensor data to the cloud using freely available cloud frameworks and visualize the collected data on the cloud.
3. Write programs to perform basic data analytics on the collected sensor data.
4. Write python program to control the sensors deployed through cloud, using one of the protocols – MQTT, SMQTT, CoAP, XMPP, AMQP.

5. Write python program to make use of one of the communication protocols – IEEE 802.15.4, ZigBee, 6LoWPAN, HART, Bluetooth, Zwave, ISA 100.11 A, to establish communication link between two or more Rasberry Pi units/clusters.
6. Realization of IoT communication protocols – IEEE 802.15.4, ZigBee, 6LoWPAN, HART, Bluetooth, Zwave, ISA 100.11 A, using Centiki OS and Cooja/NS3 Network Simulator.
7. Use solidity language to write and deploy smart contracts using the cloud IDEs – Ethereum Remix, Ethereum Studio.

Reference Books:

1. Pethuru Raj, Anupama C. Raman, "The Internet of Things: Enabling Technologies, Platforms and Use Cases", CRC Press,2017.

2. Arshdeep Bahga, Vijay Madisetti ,"Internet of Things – A Hands-on Approach", Universities Press, 2015.

3. Ammar Rayes, Samer Salam, "Internet of Things From Hype to Reality – The Road to Digitalization", Second Edition, ISBN 978-3-319-99515-1, Springer Nature Switzerland AG 2017, 2019.

4. Research Papers on recent trends in IoT.

18PCSEE227	Machine Learning	(3-0-2) 4 Credits

Contact Hours: 39T+26L

Course Learning Objectives

This course focuses on the basic principles of machine learning and design machine learning system with proper exposure to appropriate programming language, mathematical modelling, and Architectural strategy of ML, knowledge representation and Industry relevant tools / Library for developing intelligent systems.

Course Outcomes:

Description of the Course Outcome: At the end of the course the student will be able to:		Mapping to POs/ PSOs		
		Substantial Level (3)	Moderate Level (2)	Low Level (1)
CO 1	**Design** an appropriate knowledge model for a given problem scenario using concept learning and/or decision tree	1,2,3,4	-	-
CO 2	**Design** an appropriate knowledge model for a given problem scenario using neural networks and/or genetic algorithms.	1,2,3,4	-	-
CO 3	**Design** an appropriate knowledge model for a given problem scenario using Bayesian technique and K-Nearest Neighbour	1,2,3,4	-	-
CO 4	**Write** a program in any appropriate language for a given	1,2,3,4	-	-
CO 5	**Evaluate** the effectiveness of a given learning strategy under	1,2,3,4	-	-
CO 6	Explore the standard dataset available in public domain	1,2,3,4	-	-
CO 7	Conduct experiments and develop a proper dataset capturing real time features in a given domain	1,2,3,4	-	-

Mapping Level

POs	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
Mapping Level	3	3	3	3	-	-

Pre-requisites: The students meet the following prerequisites:

- Basic programming skills
- Algorithm design
- Basics of probability & statistics

Course Contents:

1	**Introduction, Concept Learning And Decision Trees:** Learning Problems– Designing Learning systems, Perspectives and Issues–Concept Learning– Version Spaces and Candidate Elimination Algorithm–Inductive bias–Decision Tree learning–Representation–Algorithm–Heuristic Space Search.	**09 Hrs**
2	**Neural Networks And Genetic Algorithms:** Neural Network Representation– Problems–Perceptrons–Multilayer Networks and Back Propagation Algorithms– Advanced Topics–Genetic Algorithms–Hypothesis Space Search–Genetic Programming – Models of Evolution and Learning.	**10 Hrs**
3	**Bayesian And Computational Learning:** Bayes Theorem–Concept Learning– Maximum Likelihood–Minimum Description Length Principle–Bayes Optimal Classifier–Gibbs Algorithm–Naïve Bayes Classifier–Bayesian Belief Network– EM Algorithm–Probably Learning–Sample Complexity for Finite and Infinite Hypothesis Spaces–Mistake Bound Model.	**10 Hrs**
4	**Instant Based Learning And Learning Set Of Rules:** K- Nearest Neighbor Learning–Locally Weighted Regression–Radial Basis Functions–Case-Based Reasoning–Sequential Covering Algorithms–Learning Rule Sets–Learning First Order Rules–Learning Sets of First Order Rules–Induction as Inverted Deduction–Inverting Resolution.	**10 Hrs**

Additional contents beyond the syllabi:

- Exposure to research avenues in the field of Machine Learning.
- Seminars on the research work done using Machine Learning algorithms.
- Certification course

LABORATORY WORK

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

1. Implement the **CANDIDATE–ELIMINATION** algorithm. Show how it is used to learn from training examples and hypothesize new instances in Version Space.
2. Implement the **FIND–S** algorithm. Show how it can be used to classify new instances of target concepts. Run the experiments to deduce instances and hypothesis consistently.
3. Implement the **ID3** algorithm for learning Boolean–valued functions for classifying the training examples by searching through the space of a Decision Tree.
4. Implement the **Genetic algorithm**.
5. Design and implement **Naïve Bayes** Algorithm for learning and classifying **TEXT DOCUMENTS**.

Reference Books:

1. Tom M. Mitchell, “Machine Learning”, McGraw-Hill Education (INDIAN EDITION), 2013.
2. EthemAlpaydin, “Introduction to Machine Learning”, 2nd Edition, PHI Learning Pvt. Ltd., 2013.
3. T. Hastie, R. Tibshirani, J. H. Friedman, “The Elements of Statistical Learning”, Springer; 1st edition, 2001.
4. Pattern Recognition and Machine Learning by Christopher M Bishop, Springer, 2016

18PCSEE231	Internet of Things	(3-0-2) 4

Contact Hours: 39T+26L

Course Learning Objectives:

This course provides basic insights on architecture, communication protocols, sensor networks and applications of IoT. It also focuses on implementation of various use cases of IoT with Arduino and Raspberry Pi.

Course Outcomes:

Description of the Course Outcome: At the end of the course the student will be able to:		Mapping to POs/ PSOs		
		Substantial Level (3)	Moderate Level (2)	Low Level (1)
CO 1	Explain the need for IoT and its application in various domains	-	3	6
CO 2	Explain various networking and communication protocols of IoT	-	3	1,6
CO 3	Explain the architecture and applications of WSNs and UAV	3	-	1,6
CO 4	Explain the concepts of M2M communication and challenges of	-	3	6
CO 5	Demonstrate use cases of IoT using Arduino and Raspberry Pi	-	3,4	6

Mapping Level:

POs	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6
Mapping Level	1.0	-	2.2	2	-	1.0

Pre-requisites: Computer Networks basics and knowledge of microcontrollers and microprocessors.

Course Contents:

1. **Introduction to IoT:** What is IoT; IoT overview; Applications of IoT; Sensing- what is a sensor and transducer, sensor types; Actuation- what are actuators, actuator types — **06 Hrs**
2. **IoT Networking:** Basics of IoT networking; components of IoT; service oriented architecture of IoT — **02 Hrs**
3. **IoT Protocols:** IoT Protocols-MQTT, SMQTT, CoAP, XMPP, AMQP; Communication Protocols- IEEE 802.15.4, ZigBee, 6LoWPAN, HART, Bluetooth, Zwave, ISA 100.11 A — **06 Hrs**
4. **WSNs and UAV Networks :** WSNs- what is wireless sensor network, concepts and challenges, node behavior; Applications of WSNs- Target/Object Tracking, Agriculture; UAV Networks- features, constraints, advantages, topologies — **06 Hrs**

5. **M2M Communication and Interoperability in IoT:** M2M communication-concepts, features, applications, node types, M2M area management; Interoperability in IoT- challenges, types of interoperability (ex: syntactic and device interoperability) **03 Hrs**
6. **Arduino Microcontroller and Sensor Integration:** Arduino Microcontroller-what is Arduino, arduino programming, features, arduino boards, arduino IDE, blinking LED; Sensor Integration- integration of various sensors with arduino **05 Hrs**
7. **Implementation of IoT with Raspberry Pi:** Raspberry Pi- architecture, components, blinking LED, image processing with Raspberry Pi; Interfacing various sensors with arduino **05 Hrs**
8. **IoT with Cloud and Data Processing:** Sending collected sensor data to cloud using recent cloud services like ThingSpeak; IBM bluemix and processing the data using IBM Watson APIs **06 Hrs**

Reference Books:

1. Pethuru Raj, Anupama C. Raman, "The Internet of Things: Enabling Technologies, Platforms and Use Cases", CRC Press, 2017
2. ArshdeepBahga, Vijay Madisetti,"Internet of Things – A Hands-on Approach", Universities Press, 2015
3. Research Papers.

Laboratory Assignments:

1. Programs to blink LED (internal/external) using Arduino / Raspberry Pi.
   CO-6,7
2. Programs to interface sensors (LM35, DHT, LDR, Ultrasonic) using Arduino / Raspberry Pi. CO-1, 6,7
3. Programs interface Bluetooth/ Wi-fi modules to Arduino / Raspberry Pi and send sensor data to cloud using cloud APIs.
   CO-3,6,7
4. Programs to explore protocols like MQTT, CoAP, and ZigBee using Arduino / Raspberry Pi. CO-3,6,7
5. Programs to send sensor data to cloud and perform data analytics using platforms like ThingSpeak / IBM Watson APIs.
   CO-6,7,8
6. Programs to realize IoT case studies like Home Automation, Smart Agriculture, Smart Cities, etc. CO-6,7

16PCSEE232 Machine Learning (4-0-0) 4

Contact Hours: 52

Course Learning Objectives: This course is at Postgraduate level for 52 contact hours with focus on following learning objectives:

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

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

CO #	Description of Course Outcomes	Substantial	Moderate	Low
CO 1	Understand the design of learning system, searching through hypothesis space and purpose of decision trees.	1	2,3	-
CO 2	Understand the architecture, design and algorithms of neural network and genetic algorithm.	1	2,3	5
CO 3	Understand the probabilistic learning approaches.	1,2	3	-
CO 4	Understand the supervised learning approaches.	2	1,3	-
CO 5	Understand the critics based learning approaches.	1,2	3	-

PO	1	2	3	4	5	6	7	8	9	10	11
Mapping Level	2.8	2.6	2	-	1	-	-	-	-	-	-

Prerequisites: Nil

Course Contents

1. **INTRODUCTION, CONCEPT LEARNING AND DECISION TREES** 10 Hrs
   Learning Problems – Designing Learning systems, Perspectives and Issues – Concept Learning – Version Spaces and Candidate Elimination Algorithm –

Inductive bias – Decision Tree learning– Representation – Algorithm – Heuristic Space Search.

2. **NEURAL NETWORKS AND GENETIC ALGORITHMS** 10 Hrs

Neural Network Representation – Problems – Perceptrons – Multilayer Networks and Back Propagation Algorithms – Advanced Topics – Genetic Algorithms – Hypothesis Space Search – Genetic Programming – Models of Evolution and Learning.

3. **BAYESIAN AND COMPUTATIONAL LEARNING** 10 Hrs

Bayes Theorem – Concept Learning – Maximum Likelihood – Minimum Description Length Principle – Bayes Optimal Classifier – Gibbs Algorithm – Naïve Bayes Classifier – Bayesian Belief Network – EM Algorithm – Probably Learning – Sample Complexity for Finite and Infinite Hypothesis Spaces – Mistake Bound Model.

4. **INSTANT BASED LEARNING AND LEARNING SET OF RULES** 10 Hrs

K- Nearest Neighbor Learning – Locally Weighted Regression – Radial Basis Functions – Case-Based Reasoning – Sequential Covering Algorithms – Learning Rule Sets – Learning First Order Rules – Learning Sets of First Order Rules – Induction as Inverted Deduction – Inverting Resolution.

5. **ANALYTICAL LEARNING AND REINFORCED LEARNING** 12 Hrs

Perfect Domain Theories – Explanation Based Learning – Inductive-Analytical Approaches - FOCL Algorithm – Reinforcement Learning – Task – Q-Learning – Temporal Difference Learning.

Reference Books:

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

16PCSEE235	Internet of Things	(2-0-4) 4

Course Learning Objectives:

At the end of this course, the student should understand the basics of **Internet of Things** and **Cloud Computing** technology and should be in a position to develop his own product deployed on cloud in Java / Python / Android programing languages for Arduino & RaspberryPi with Wireless Sensor Networks.

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

CO #	Description of Course Outcomes	Substantial	Moderate	Low
CO1	**Design** and **Develop** IoT with Cloud applications using Microcontroller, Modules and	1,2,3	4,5	10

	Sensors			
CO2	**Design** and **Develop** IoT with Cloud applications using Microprocessor, Modules and Sensors	1,2,3	4,5	10
CO3	**Analyze** streaming (sensor) data using Cloud Data Analytical tools	1,2,3	4,5	10
CO4	**Assemble** and **Control** : Robots and Drones	1,2,3	4,5	10
CO5	**Design** & **Develop** Cloud/Mobile Applications using Cloud IDEs, Google, Facebook, Twitter, LinkedIn & Aadhaar APIs	1,2,3	4,5	10

PO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
Mapping Level	3	3	3	2	2	-	-	-	-	1	-

Prerequisites: No prior knowledge is required (open elective for IT branches), but programming knowledge will be useful.

Course Contents:

1. **Introduction to Internet of Things (IoT) and Sensors:** IoT overview, IoT Applications, Sensors - overview, applications, usage. Modules - Bluetooth, Wifi, Ethernet, GSM, GPS, Camera - overview, applications, usage. DC, Stepper motors — 6Hrs
2. **Arduino Micro-controller:** Introduction, Arduino Pinout, Types, Programming Arduino using cloud and offline IDEs, Introduction to MIT App Inventor — 5Hrs
3. **IoT Protocol and Data Processing:** Introduction**,** Protocols - MQTT, ZigBee; Cloud for IoT ; IBM BlueMix and Watson Analytics - RTI — 4Hrs
4. **Raspberry Pi Micro-processor:** Introduction, Raspberry Pi Pinout, Types, Programming RaspberryPi using Python, Image Processing using OpenCV and IBM Watson APIs — 5Hrs
5. **Introduction to APIs:**Google, Facebook, Twitter, LinkedIn & Aadhaar APIs — 3 Hrs
6. **IoT Case Study:** Design and development of Robots, Drones with Solar Panels using 3D printers — 3 Hrs

Additional contents beyond the syllabi: Students need to develop useful products and conduct a workshop cum exhibition to juniors.

Online Courses Certification for CTA Marks

Reference Books:

1. Arshdeep Bahga, Vijay Madisetti - Internet of Things, A Hands-on Approach, Universities Press, 2015.