SHRI G.S. INSTITUTE OF TECHNOLOGY & SCIENCE, INDORE

DEPARTMENT OF COMPUTER ENGINEERING

B. Tech. (Computer Science & Engineering)

Year: 2023-24

DEPARTMENT OF COMPUTER ENGINEERING
B. Tech. I YEAR (4YDC)
CO 10507: PROGRAMMING FOR PROBLEM SOLVING

*Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
2	1	-	3	-	70	30	-	-	100

PRE-REQUISITES: NIL

COURSE OBJECTIVES: This course aims to provide exposure to problem-solving through programming. It aims to train the student to the basic concepts of the C-programming language.

COURSE OUTCOMES:
After completing the course student should be able to:

1. To formulate simple algorithms for arithmetic and logical problems.
2. To translate the algorithms to programs and test and execute the programs and correct syntax and logical errors.
3. To apply programming to solve conditional and iterative statement, function, recursion, and arrays.
4. To use pointers and structures to formulate algorithms and programs. To use the concepts of OOPS.

COURSE CONTENTS:
THEORY:

Unit	Topics	CO
UNIT 1	Block Schematic of digital computer and it's working. Introduction to computer hardware and software, Flowchart and algorithm.Structure of C programs, key words and identifiers, constants, variables and memory representation, Data types, Declarations, scope and life of variables.	CO1
UNIT 2	Various types of operators and expressions. Operator precedence and associativity. Programming errors and their handling. Data type conversion and promotion, Decision making and Branching: if-else, switch-case. Use of break, continue, default.	CO1, CO2
UNIT 3	Use of looping: While-do, for, do-while etc., nesting of loops. Functions and Recursive functions.	CO2, CO3
UNIT 4	Introduction and use of array, multidimensional array, character array, strings, structure, union, pointers and Files in the programming.	CO3, CO4
UNIT 5	Introduction to Object oriented Programming paradigm, Comparison of Procedural and Object Oriented Programming paradigm. OOPS concept: Class, object, inheritance, polymorphism, Overloading vs. Overriding etc.	CO4

COURSE ASSESMENT (Th.):

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

TEXT BOOKS RECOMMENDED:

1. E. Balagurusamy, “Programming in ANSI C”, Sevenths Edition, Tata McGraw Hill, 2017.
2. ReemaThareja , “Programming in C”, Second Edition , Oxford publication, 2016
3. W. Kernighan and Dennis M. Ritchie, “The C Programming Language”, Pearson , 2015

REFERENCE BOOKS:

1. Matthias Felleisen, Robert BruceFindler , Mathew Flatt, ShriramKrishnamurthi, ”How to Design Programs: An Introduction to Programming and Computing”, Second Edition, MIT Press, 2018.
2. M.Chandwani, A. Jain and N.S. Chaudhary, ” Elements of Computer Science” , Jain Publishers 1997
3. E. Balagurusamy, ” Object Oriented Programming with C++”, Tata McGraw Hill, 2009
4. B.S. Gottfried, “Programming with C”, 3rd edition, Tata McGraw Hill, 2018
5. Problem Solving and Program Design in C, by Jeri R. Hanly, Elliot B. Koffman, Pearson Addison-Wesley, 2006.

Mapping of Course Outcomes with Program Outcomes

Statement of CO	**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**Average Target CO Attainment**
CO1	3	3	2	1	2				1		1		1.86
CO2	3	3	2	1	2				1		1		1.86
CO3	3	3	3	2	2				1		1		2.14
CO4	3	3	3	2	2				1		1		2.14
Average Target	3	3	2.5	1.5	2				1		1		2

DEPARTMENT OF COMPUTER ENGINEERING
B. Tech. I YEAR (4YDC)
CO 10654: Computer Programming Lab

*Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
-	-	2	-	1	-	-	20	30	50

PRE-REQUISITES: NIL

COURSE OBJECTIVES: This course aims to provide basic principles of C programming language. It aims to provide design & develop of C programming skills, which gives the student hands-on experience with the concepts.

COURSE OUTCOMES:

After completing the course student should be able to:

1. List linux command and editing tools and Develop C programs to solve simple mathematical and decision making problems.
2. Develop C programs to solve simple engineering problems using looping constructs.
3. Develop C programs to demonstrate the applications of derived data types such as arrays, pointers, strings and functions, recursion.
4. Develop C programs to demonstrate the use of Structures and Files

S. No.	LIST OF EXPERIMENTS	CO
Lab 1	Hands-on the Linux Commands: cd, cmp, cat, ls, man, mkdir, mv, passwd, pwd, rm, rmdir, cp Hands-on the Editing tools: vi/gedit/nano/touch	CO1
Lab 2	Develop C program using I/O statements, operators and expressions	CO1
Lab 3	Develop C program using decision-making and branching constructs: if-else, switch-case, break, default.	CO1
Lab 4	Develop C program using Loops: for, while, do-while, nesting of loops, break-continue, exit.	CO2
Lab 5	Develop C program using Arrays: 1D.	CO3
Lab 6	Develop C program using Arrays: 2D, Multi-dimensional arrays, traversal	
Lab 7	Develop C program using Strings: operations	CO3
Lab 8	Develop C program using Functions: call, return, passing parameters by (value, reference), passing arrays to function	CO3
Lab 9	Formulate program using Recursion	CO3
Lab 10	Develop C program using Pointers: Pointers to functions, Pointers to Arrays, Pointers to Strings, Pointers to Pointers, Array of Pointers	CO3
Lab 11	Develop C program using Structures: Nested Structures, Pointers to Structures, Arrays of Structures and Unions.	CO4
Lab 12	Develop C program Files: reading and writing, File pointers file operations, random access.	CO4

COURSE ASSESMENT (Pr.):

1. Internal Assessment for continuous evaluation (40%): Lab assignments, Demonstration, Quiz, Viva, file etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

TEXT BOOKS RECOMMENDED:

1. E. Balagurusamy, "Programming in ANSI C", Sevenths Edition, Tata McGraw Hill, 2017.
2. ReemaThareja , "Programming in C", Second Edition , Oxford publication, 2016

REFERENCE BOOKS:

1. Matthias Felleisen, Robert BruceFindler , Mathew Flatt, ShriramKrishnamurthi, "How to Design Programs: An Introduction to Programming and Computing", Second Edition, MIT Press, 2018.
2. E. Balagurusamy, " Object Oriented Programming with C++", Tata McGraw Hill, 2009
3. B.S. Gottfried, "Programming with C", 3rd edition, Tata McGraw Hill, 2018

Mapping of Course Outcomes with Program Outcomes

Statement of CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**Average Target CO Attainment**
CO1	3	3	2	1	2				1		1		1.86
CO2	3	3	2	1	2				1		1		1.86
CO3	3	3	3	2	2				1		1		2.14
CO4	3	3	3	2	2				1		1		2.14
Average Target	3	3	2.5	1.5	2				1		1		2

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. II YEAR (4YDC)
SEMESTER-A
CO 24057: OBJECT ORIENTED PROGRAMMING SYSTEMS

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	1	2	4	1	70	30	40	60	200

PRE-REQUISITES: CO10504: Computer Programming

COURSE OBJECTIVES: The objective of course is to develop programming skills of students, using object oriented programming concepts, learn the concept of class and object using Java and develop real world applications.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Explain various concepts of object oriented terminology.
2. Define and implement the concepts of data encapsulation, abstraction, inheritance and polymorphism.
3. Design and execute quality programs using exception handling.
4. Solve the real world business problems as per specifications.

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction to Object Oriented Thinking & Object Oriented Programming: Comparison with Procedural Programming, features of Object oriented paradigm, Merits and demerits of OO methodology; Object model; Elements of OOPS, IO processing.

UNIT 2 Encapsulation and Data Abstraction- Concept of Objects: State, Behavior & Identity of an object; Classes: identifying classes and candidates for Classes Attributes and Services, Access modifiers, Static members of a Class, Instances, Message passing, and Construction and destruction of Objects.

UNIT 3 Relationships – Inheritance: purpose and its types, ‘is a’ relationship; Association, Aggregation. Concept of interfaces and Abstract classes.

UNIT 4 Polymorphism: Introduction, Method Overriding & Overloading, static and run time Polymorphism.

UNIT 5 Strings, Exceptional handling, Introduction of Multi-threading and Data collections. Case study like: ATM, Library management system.

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Timothy Budd, “An Introduction to Object-Oriented Programming”, Addison-Wesley Publication, 3rd Edition 2002.
2. Cay S. Horstmann and Gary Cornell, “Core Java: Volume I, Fundamentals”, Prentice Hall publication, 2007.

REFERENCE BOOKS:

1. G. Booch, “Object Oriented Analysis & Design”, Addison Wesley, 2006
2. James Martin, “Principles of Object Oriented Analysis and Design”, Prentice Hall/PTR, 1992.
3. Peter Coad and Edward Yourdon, “Object Oriented Design”, Prentice Hall/PTR, 1991.
4. Herbert Schildt, “Java 2: The Complete Reference”, McGraw-Hill Osborne Media, 11th Edition, 2018.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. II YEAR (4YDC)
SEMESTER-A
CO 24009: COMPUTER ARCHITECTURE

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: NIL

COURSE OBJECTIVES:

The main objective of this course is to compare various architectures of Computers and their components like memory etc. and to develop the skills of the students to write the assembly language programs for various instructions.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Illustrate architecture of a computer, its components and their interconnection.
2. Describe execution of instruction in a computer.
3. Identify the addressing modes used in macro instruction.
4. Design programs in assembly language and justify the importance of parallel architecture.

COURSE CONTENTS:

THEORY:

UNIT 1 Introduction, Milestones in Computer Architecture, Von Neumann Model: Processor Organization- ALU, Control Unit; System Bus, Memory, I/O Devices. Multilevel model of Computer, Concept of Instruction Execution.

UNIT 2 Review of Combinational and Sequential Circuits. Memory Organization: Memory Hierarchy, Memory Properties, Main Memory, Associative Memory, Cache Memory. Machine Language Level (ISA level): Instruction Formats, Addressing Modes, Instruction Types, Flow of Control. RISC v/s CISC.

UNIT 3 Memory mapped I/O and I/O mapped I/O, I/O Techniques: Programmed I/O, Concept of Interrupts, Interrupt driven I/O and DMA, I/O Device Interfaces, I/O Processors. Serial and Parallel Communication. Computer Buses.

UNIT 4 Concept of Hardwired and Micro Programmed Control. Micro Instructions, Instruction Fetch and Queuing, Micro Instruction Control, Design of the Micro Architecture Level.

UNIT 5 Parallel Architectures: On-chip Parallelism- Instruction Level Parallelism, On-chip Multithreading, Multicore Processor Architecture. Pipelining: RISC Pipeline, Exception handling of Pipelining, Hazards of Pipelining.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. William Stallings, "Computer Organization and architecture", Ninth Edition, Pearson, 2012.
2. Tannenbaum and Austin, "Structured Computer Organization", Sixth edition, PHI, 2013.
3. Michael J. Flynn "Computer Architecture: Pipelined and Parallel Processor Design, First Edition, 1995.

REFERENCE BOOKS:

1. V. Carl Hamacher, "Computer Organization", Fifth Edition, 2011, McGraw Hill.
2. John P. Hayes, "Computer Architecture and Organization", Fourth Edition, TMH, 2003
3. Morris Mano, "Computer System Architecture", Third Edition, 2007, PHI.
4. David A. Patterson and John L. Hennessy, "Computer Organization and Design: The Hardware/Software Interface", Fourth Edition, Morgan Kauffman, 2011

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. II YEAR (4YDC)
SEMESTER-A
CO 24497: PROGRAMMING PRACTICES

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
-	1	2	-	2	-	-	40	60	100

PRE-REQUISITES: CO10504: Computer Programming

COURSE OBJECTIVES: To give hands-on experience to students on different components of good programming.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Develop programs for complex real world problems.
2. Apply good programming practices in their code like Comments, indentation etc.
3. Utilize Debugger and its tools like gdb/gnu for error handling.
4. Demonstrate configuration and usage of different software tools used in industry.

COURSE CONTENTS:

UNIT 1 Getting Familiar With Linux (Ubuntu/CentOS)

Lab 1. Doing regular activities:

(a) Changing wallpaper.
(b) Playing multimedia files (video and audio).
(c) Creating documents (docs, ppts and spreadsheets).
(d) Install softwares through repositories and different commands.
(e) Editing images using GIMP (MS Paint like software).
(f) Text Editors (Vi/Vim, Gedit, Emacs, Sublime Text etc.)

Lab 2. Bash commands and Linux utilities:

(a) Find files on hard disk using locate and find.
(b) Find within a file (text or pdf).
(c) sed and awk.
(d) View current IP address of my computer/workstation and how to change it.
(e) Rename a file or folder.
(f) Move/copy file/folder from one place to another.
(g) Change password of your user account.
(h) View hardware specification of my computer/workstation (CPU Model, RAM, HDD etc.).
(i) Documentation of commands (man, help, info).

UNIT 2 Programming

Lab 3. Introduction to file handling, structures and pointers in C.

Lab 4. Introduction to Python 3.

(a) Inbuilt data types (list, dictionary, sets etc.)
(b) Control structures (e.g. if -else, loops, functions)
(c) Introduction to Jupyter notebook.

Lab 5. Coding style.

(a) GNU Coding Standards
(b) Writing meaningful comments.
(c) Appropriate variable/function/file names.
(d) Indentation.

UNIT 3 Debugging using GDB.

Lab 6 (a) Listing source code lines.
(b) Compiling with debugging enabled.
(c) Starting a program step by step.
(d) Execute one step at a time, one function at a time, stepping out of a function.
(e) Printing values of variables, structures, arrays, strings, pointers, linked lists etc.
(f) Examining addresses (x).
(g) Stack frames.

Lab 7 (j) Running a program till the first breakpoint or run time error.
(k) Listing local, global variables. (using info, locals, all variables)
(l) Watches (list watches[info watch], write watches, read watches)
(m) Breakpoints (on line numbers, functions etc.)
(n) Continue (till next breakpoint or watch).
(o) Current line address (frame, where).
(p) Code profiling.

UNIT 4 Project management

Lab 8. (a) Subversion control using Git and GitHub/BitBucket.
(b) Using Ant.
(c) Download and modify code of an open source software.

Lab 9 Using an IDE.
(i) Create and manage projects.
(ii) Debug in CodeBlocks/IDE.
(iii) Execute/build fies/projects.
(iv) Auto indent, bulk comments.
(v) Auto complete identifier names.

UNIT 5 Scientific Documentation

Lab 10 LaTeX and Overleaf.com.
(a) Text formatting (bold, italic, size, colour etc.)
(b) Types of scientific documents (research papers, reports etc.)
(c) Setting margins
(d) itemize, enumerate.
(e) Mathematical Symbols (e.g. ; ; ac++db ; P; Q; 1; px + y + z, etc.)
(f) Equations.
(g) Sections, subsections, subsubsections ...
(h) Figures and tables.
(i) referencing sections, figures, tables, equations, references, etc.
(j) BibTex and references.

DIRECT ASSESMENT:
ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. "Learn Python 3 The Hard Way" by Zed A. Shaw, Addison-Wesley Publisher, January 2017.
2. "The Art of Readable Code" by Dustin Boswell, Trevor Foucher, O'Reilly Publication, Inc. November 2011.
3. "Art of Debugging with GDB, DDD, and Eclipse" by Norman Matloff and Peter Jay Salzman, No Starch Press, September 2008.
4. "LaTeX Cookbook" by Stefan Kottwitz, Packt Publishing, October 2015.
5. "Learning IPython for Interactive Computing and Data Visualization" by CyrilleRossant, 2nd Edition, Pakt Publication, October 2015.
6. "Mastering Git" By Jakub Narębski, Pakt Publication, April 2016.

REFERENCE BOOKS:

1. GNU GDB Documentation, https://www.gnu.org/software/gdb/documentation/
2. TeX User Group, https://www.tug.org/
3. Google CoLab for Jupyter, https://research.google.com/colaboratory/

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. II YEAR (4YDC)
SEMESTER-B
CO 24553: DISCRETE STRUCTURES

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	3	-	70	30	-	-	100

PRE-REQUISITES: MA10001: Mathematics - I, MA10501: Mathematics - II

COURSE OBJECTIVES: Course Objective: To enable a student to learn the basic concepts of discrete elements such as sets, combinatorics, relations, functions, graphs, trees and their applications in diverse domains.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Solve problems which involve discrete structures such as sets, relations, functions, predicate and propositional logics.
2. Explain the notion of mathematical thinking, mathematical proofs, logical thinking, and combinatorics and be able to apply them in problem solving.
3. Describe the basic terminology and properties of graphs and trees and apply them to solve practical problems.
4. Apply algebraic techniques effectively to analyse basic discrete structures and algorithms for real world applications.

COURSE CONTENTS:
THEORY:

UNIT 1 Set Theory: Operations, Types, Set Identities, Computer Representation of sets; Principle of Mathematical Induction; Principle of Inclusion-Exclusion; Logic Theory: Prepositional Logic: Well formed formula, Truth Table, Algebra of Proposition; Introduction to Predicate logic.

UNIT 2 Counting Techniques and Combinatorics; Relations and Functions: Properties and types of relations and functions, Application to RDBMS and Hashing; Pigeonhole Principle.

UNIT 3 Graphs and Trees: Basic terminologies, Types, Properties, Shortest path Algorithms; Cutsets; Hamiltonian and Eulerian paths and circuits; Tree Traversals; Spanning Trees; Applications in computer science.

UNIT 4 Recurrence Relations and Generating Functions: Introduction, Problem Solving; Introduction to Complexity of Problems and Algorithms.

UNIT 5 Algebraic Systems: Groups, Rings, Fields, Integral Domain, Lattices, Boolean Algebra.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

3. Feedback of students on attainment of cos.
4. Feedback of students on classroom learning.
5. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Kenneth H. Rosen, “Discrete Mathematics and its Applications”, 7th Edition,McGraw Hill, 2017.
2. Liu C.L., “Elements of Discrete Mathematics”, Fourth Edition, McGraw-Hill, 2012.
3. J.P. Tremblay and R. Manohar, “Discrete Mathematical Structures with Applications to Computer Science”, First Edition, McGraw-Hill, 2017.

REFERENCE BOOKS:

1. Kolman, Busby, Ross and Rehman, “Discrete Mathematical Structures”, Sixth Edition, Pearson Education, 2015.
2. NarsinghDeo, “Graph Theory with Applications to Engineering and Computer Science”, First Edition, Dover Publications, 2016.
3. Seymour Lipschutz, Marc Laras Lipson, “Discrete Mathematics”, Schaum's Outlines, Third Edition, 2017.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. II YEAR (4YDC)
SEMESTER-B
CO 24507: DATA STRUCTURES

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	1	2	4	1	70	30	40	60	200

PRE-REQUISITES: CO10504: Computer Programming, CO24497: Programming Practices

COURSE OBJECTIVES: To introduce the fundamental concept of data structures and to emphasize the importance of data structures in developing and implementing algorithms.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Explain the basic concepts of data structures and algorithms.
2. Describe basic concepts about stacks, queues, linked lists, trees, graphs and their implementation.
3. Apply data structures to efficiently organize the data for improving performance of the system.
4. Design and implement algorithms for solving problems with the help of fundamental data Structures.

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction to Data Structure: Concepts of Data and Information, Classification of Data structures, Abstract Data Types, Implementation aspects: Memory representation. Data structures operations and its cost estimation. Introduction to linear data structures- Arrays, Linked List: Representation of linked list in memory, different implementation of linked list. Circular linked list, doubly linked list, etc. Application of linked list: polynomial manipulation using linked list, etc.

UNIT 2 Stacks: Stacks as ADT, Different implementation of stack, multiple stacks. Application of Stack: Conversion of infix to postfix notation using stack, evaluation of postfix expression, Recursion. Queues: Queues as ADT, Different implementation of queue, Circular queue, Concept of Dqueue and Priority Queue, Queue simulation, Application of queues.

UNIT 3 Tree: Definitions - Height, depth, order, degree etc. Binary Search Tree – Operations, Traversal, Search. AVL Tree, Heap, Applications and comparison of various types of tree; Introduction to forest, multi-way Tree, B tree, B+ tree, B* tree and red-black tree.

UNIT 4 Graphs: Introduction, Classification of graph: Directed and Undirected graphs, etc., Representation, Graph Traversal: Depth First Search (DFS), Breadth First Search (BFS), Graph algorithm: Minimum Spanning Tree (MST)-Kruskal, Prim's algorithms. Dijkstra's shortest path algorithm; Comparison between different graph algorithms. Application of graphs.

UNIT 5 Sorting: Introduction, Sort methods like: Bubble Sort, Quick sort. Selection sort, Heap sort, Insertion sort, Shell sort, Merge sort and Radix sort; comparison of various sorting techniques. Searching: Basic Search Techniques: Sequential search, Binary search, Comparison of search methods. Hashing & Indexing. Case Study: Application of various data structures in operating system, DBMS etc.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

3. Feedback of students on attainment of cos.
4. Feedback of students on classroom learning.
5. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. AM Tanenbaum, Y Langsam& MJ Augustein, “Data structure using C”, Prentice Hall India-2007
2. Robert Kruse, Bruse Leung, “Data structures & Program Design in C”, Pearson Education, 2007.
3. Richard, Gilberg Behrouz, Forouzan ,“Data structure – A Pseudocode Approach with C”, Thomson press, 2005.

REFERENCE BOOKS:

1. Jean – Paul Trembly , Paul Sorenson, “An Introduction to Structure with application”, TMH, 2007.
2. Aho, Hopcroft, Ullman, “Data Structures and Algorithms”, Pearson Education, 2002.
3. N. Wirth, “Algorithms + Data Structure = Programs”, Prentice Hall, 1978.
4. SartajSahni, “ DataStructures,Algorithms and Applications in C++” Universities Press.
5. ReemaThareja, “Data Structures Unsing C”, Oxford Press 2012

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. II YEAR (4YDC)
SEMESTER-B
CO24508: OPERATING SYSTEMS

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: CO24009: Computer Architecture

COURSE OBJECTIVES:

To enable a student in a high level understanding of how operating systems work with respect to Process management, Memory management, File Systems and I/O management.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Describe the importance and objectives of an operating system and various services provided by the operating system.
2. Interpret the important functions of different modules of an Operating system, like process management, memory management, device management and file system, etc. and will be able to apply these concepts in given test cases.
3. Compare and contrast different policies of CPU scheduling, Inter-process Communication, Page replacement and disk scheduling algorithms etc.
4. Design and develop small modules, shell and utility programs using system calls of Linux or some educational operating system.

COURSE CONTENTS:

THEORY:

UNIT 1 **Introduction to Operating Systems**: Function, Evolution, Desirable Characteristics and features of an O/S, Operating Systems Services: Types of Services, Different ways of providing these Services – Utility Programs, System Calls.

UNIT 2 **Process Management**: Concept of a process, Process State Diagram, Process based kernel, Dual mode of process execution, CPU scheduling algorithms, deterministic modeling, and System calls for Process Management, Concept of Threads: User level & Kernel level Threads. Process Management in UNIX & Windows

Inter Process Communication: Real and Virtual Concurrency, Mutual Exclusion, Synchronization, Critical Section Problem, Solution to Critical Section Problem: Semaphores and their Operations and their implementation.Deadlocks: Deadlock Problems, Characterization, Prevention, Avoidance, Recovery. IPC in UNIX & Windows

UNIT 3 **Memory Management**: Different Memory Management Techniques – Partitioning, Swapping, Segmentation, Paging, Paged Segmentation, Comparison of these techniques, Techniques for supporting the execution of large programs: Overlay, Dynamic Linking and Loading, Virtual Memory – Concept, Implementation by Demand Paging etc. Memory management in UNIX & Windows

UNIT 4 **File Systems Management**: File Concept, User's and System Programmer's view of File System, Disk Organization, Tape Organization, Different Modules of a File System, Disk Space Allocation Methods – Contiguous, Linked, Indexed. Directory Structures, File Protection, System Calls for File Management, Disk Scheduling Algorithms. File Systems in UNIX & Windows.

UNIT 5 **Input / Output Management**: Principles and Programming, Input/Output Problems, Different I/O operations: Program Controlled, Interrupt Driven, Concurrent I/O, Asynchronous Operations, Logical structure of I/O function, I/O Buffering,Kernel I/o Subsystem.Introduction to Network, Distributed and Multiprocessor Operating Systems. I/O management in UNIX & Windows

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Silberschatz, Galvin, Gagne, "Operating System Concepts'', Wiley, 10/E, 2019.ISBN: 978-1-119-45586-8
2. William Stalling, "Operating Systems", Pearson Education, 9/E, 2019.

REFERENCE BOOKS:

1. Andrew S. Tanenbaum, "Modern Operating Systems", 4/E, Prentice Hall, 2014.
2. Maurice J. Bach, "The Design of Unix Operating System", First Edition, Pearson, 2015.
3. Bovet &Cesati, "Understanding the Linux Kernel", O'Reilly, 3/E, 2005.
4. https://www.w3schools.in/operating-system-tutorial/

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. II YEAR (4YDC)
SEMESTER-B
CO 24992: COMPUTER WORKSHOP

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
-	-	2	-	1	-	-	40	60	100

PRE-REQUISITES: CO 24009: Computer Architecture

COURSE OBJECTIVES: This course will provide students a much needed knowledge of computer hardware and networking, enabling them to identify and rectify the onboard computer hardware, software and network related problems.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Identify different components in a computer, their connection and arrangement.
2. Apply knowledge about computer peripherals to identify/rectify problems onboard.
3. Identify and install various peripheral devices, device drivers, device controllers and Interfaces.
4. Compare and contrast different kinds of memory and storage devices (HDD).

COURSE CONTENTS:
THEORY:

LAB 1 Overview of system units, Processors and Specifications of different categories of Processors.

LAB 2 Assembling and disassembling the Desktop computer Cabinet.

LAB 3 Motherboard and different categories of Motherboard– sockets and slots, expansion slots, chipsets – northbridge, south bridge.

LAB 4 CMOS, BIOS – POST, BIOS features BIOS and Boot sequences.

LAB 5 Introduction to Peripherals, Interrupts and Interfaces and types of interfaces.

LAB 6 Buses, Industry standard architecture (ISA), peripheral component Interconnect (PCI), Accelerated Graphics port (AGP).

LAB 7 SCSI concepts, VGA, USB architecture, Monitors.

LAB 8 Drives - magnetic storage, magnetic recording principles, data and disk organization, hard drive.

LAB 9 CD-ROM drive, construction, CDROM electronics, DVD-ROM, DVD media, DVD drive and decoder.

LAB 10 Mouse and its types, Memory Hierarchy.

LAB 11 Parallel port, asynchronous communication, serial port signals, video adapters, modems.

LAB 12 Keyboards and types of key Switches and its demonstration.

LAB 13 Printers and working on different types of printers.

DIRECT ASSESMENT:
ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Craig Zacker& John Rourke, “PC Hardware: The complete Reference”, McGraw Hill, 27th reprint, 2015.
2. D Balasubramanian, “Computer Installation and Servicing” Tata McGraw Hill, 2/e, 12th reprint 2012.

REFERENCE BOOKS:

1. Rajaraman V., “Fundamentals of Computers”, Publisher: PHI Learning, 2014.
2. Sanjay K. Bose, “Hardware and Software of Personal Computers”, New Age International Private Limited; 1/e edition, 2014.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. II YEAR (4YDC)
SEMESTER-B
CO 24991: DESIGN THINKING LAB-I

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
-	-	2	-	1	-	-	40	60	100

PRE-REQUISITES: NIL

COURSE OBJECTIVES:

1. To provide a social and thinking space for the recognition of innovation challenges and the design of creative solutions.
2. To immerse students into the world of innovation as a systematic process of tackling relevant business and/or social problems.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Develop a fundamental understanding of the Design Process and how it can be applied to innovation and creative problem-solving.
2. Build empathy for target audiences by applying different techniques.
3. Develop and test innovative ideas through a rapid iteration cycle.
4. Design physical prototypes/a visual representation of an idea.

COURSE CONTENTS:

LAB 1: Understanding of Design Thinking and its process model, Principles and tools.
LAB 2: How to Empathize, Role of Empathy in design thinking, Empathy Maps Design.
LAB 3: Methods for Empathetic Design, Creation of User Personas.
LAB 4: Customer Journey Mapping.
LAB 5: Problem clarification, Understanding of the problem.
LAB 6: Problem analysis and Reformulation of the problem.
LAB 7: Point-of-View Phase and Context Diagram.
LAB 8: Root Causes Analysis, Conflict of Interest, Description of customer need.
LAB 9: Cash Flow Diagram and Value Chain Analysis Diagram.
LAB 10: Design Innovation, Benefits of iteration in design thinking process.

DIRECT ASSESMENT:

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Tim Brown, Clayton M. Christensen , IndraNooyi and Vijay Govindarajan, HBR's 10 Must Reads on Design Thinking: by Harvard Business Review Press, 2020.
2. Dr. BalaRamadurai, " Karmic Design Thinking", First Edition, 2020.
3. Jeanne Liedtka, Tim Ogilvie, and Rachel Brozenske, The Designing for Growth Field Book: A Step -by - Step Project Guide (Columbia University Press, 2014).

REFERENCE BOOKS:

1. 101 Design Methods: A Structured Approach for Driving Innovation in Your Organization by Vijay Kumar.
2. Human-Centered Design Toolkit: An Open-Source Toolkit to Inspire New Solutions in the Developing World by IDEO.
3. This is Service Design Thinking: Basics, Tools,Cases by Marc Stickdorn and Jakob Schneider.
4. Ulrich, Karl T. Design: Creation of artifacts in society. Karl T. Ulrich, 2011.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-A
CO 34002: THEORY OF COMPUTATION

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	1	-	4	-	70	30	-	-	100

PRE-REQUISITES: CO24553: Discrete Structures

COURSE OBJECTIVES:

This course will help students to learn several formal mathematical models of computation along with their relationships with formal languages and grammars. Students will also learn about solvable and unsolvable problems.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Compare and analyze different theoretical computational models, languages and grammars.
2. Design and construct finite automata, pushdown automata and Turing machine for various problems.
3. Identify limitations of some computational models and possible methods of proving them.
4. Describe the concept of computable and non computable problems.

COURSE CONTENTS:

THEORY:

UNIT 1 Introduction: Review of Sets, Graphs, Trees, Mathematical formal proofs including proof by induction and by contradiction; Introduction to languages, grammars and automata: Alphabet, Representation of language and grammar, Types of Automata.

UNIT 2 Finite Automata- Deterministic Finite Automata, Nondeterministic Finite Automata, Regular Expressions, Regular Grammars and Languages, Properties of Regular Languages, Pumping Lemma for Regular Languages, Applications of Regular Expressions.

UNIT 3 Pushdown Automata- Context Free Grammar, Parsing, Ambiguity, Nondeterministic PDAs, Normal form of CFGs, CFG to NPDA, NPDA to CFGs, Deterministic PDA, Pumping Lemma for CFGs, Application of CFGs.

UNIT 4 Turing Machines – Turing Machine as acceptor, Recognizing a Language, Universal TMs, Linear Bounded Automata, Context Sensitive Languages, Recursive and Recursively Enumerable Languages, Unrestricted Grammars.

UNIT 5 Chomsky Hierarchy, Concept of Solvability and Unsolvability, Church’s Thesis, Complexity Theory – P and NP problems, Introduction to Petri Nets.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. John C. Martin, “Introduction to Languages and the Theory of Computation”, Fourth Edition, Mc Graw Hill, 2010.
2. Hopcroft, Ullman, Motwani, “Introduction to Languages, Automata and Computation”, 3rd Edition, Pearson Education, 2008.
3. Peter Linz, “An Introduction to Formal Languages and Automata”, Sixth Edition, Jones and Bartlett, 2016.

REFERENCE BOOKS:

1. Lewis and Papadimitiriou, “Elements of Theory of Computation”, Second Edition, Pearson Education, 2015.
2. Mandrioli D. and Gezzi C., “Theoretical Foundations of Computer Science”, Krieger Publishing Co., Inc., USA, 1993.
3. K.L.P. Mishra and N. Chandrasekaran, “Theory of Computer Science: Automata, Languages and Computation”, Third Edition, Prentice Hall, 2006.
4. Cohen John, “Introduction to Computer Theory”, Second Edition, Wiley and Sons, 2007.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-A
CO 34005: DATA BASE MANAGEMENT SYSTEMS

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	1	2	4	1	70	30	40	60	200

PRE-REQUISITES:

1. CO24553: Discrete Structures
2. CO24508: Operating Systems

COURSE OBJECTIVES: The objective of this course is to enable students in developing a high level understanding of the concepts of Database management systems in contrast with traditional data management systems with emphasis on skills to apply these concepts in building, maintaining and retrieving data from these DBMS.

Course Outcomes : After completing the course student should be able to:

CO1 Compare & Contrast between traditional data processing and Database Management systems, its application and can describe design of a database at various levels.

CO2 Design a database using Entity Relationship diagram and other design techniques.

CO3 Apply fundamentals of relational model to model & implement a sample Database Management System for a given domain.

CO4 Query Database Management systems using SQL ; evaluate and optimize queries and apply concepts of transaction management

COURSE CONTENTS:
THEORY:

UNIT 1 Basic Concepts of Data and DBMS, File organization and access methods; Introduction to DBMS, Difference between DBMS and traditional file storage system. Characteristics of DBMS. Data Models, Schemas and Instances, DBMS architecture, Components of DBMS. Data Independence. Study of Entity Relationship Model, Type of attributes, Entity types, Relationship and Cardinalities, Participation, Roles and constraints.

UNIT 2 Relational Data Model: Domains, Tuples, Attributes, Relations, keys and types of keys, Integrity Constraints, Relational Algebra: Queries using Select operation, project operation, renaming, joins, union, intersection, difference, division, and product etc.
SQL –basic SQL queries, functions, constraints, joins and nested queries, Triggers, assertions, views and stored procedures and PL/SQL.

UNIT 3 Normalization Theory and Database methodologies: Relation Schemas, Functional Dependencies- Definition and rules of axioms, Normal forms- 1NF, 2NF, 3NF and BCNF, Dependency preservation, properties, loss less join decomposition.

UNIT 4 Transaction Processing: States of transaction and desirable properties, Introduction to Concurrency and Recovery, Schedules, Recoverability & Serializability, types of serializability and test for serializability, Concurrency Control: Two phase locking, Timestamp Based concurrency control. Deadlocks: Avoidance, Prevention , detection &resolution, Recovery: Basic concepts, techniques based on deferred update and immediate update, Shadow paging, check points.

UNIT 5 Storage structures: Secondary Storage Devices, Hashing & Indexing structures: Single level & multilevel indices, Query Processing and Optimization: Various algorithms to implement select, project & join operation of relational algebra, complexity measures. Case Study of any contemporary DBMS.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

3. Feedback of students on attainment of cos.
4. Feedback of students on classroom learning.
5. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Korth H.F. &Silberschatz A., Sudarshan, “Database Systems”, McGraw-Hill, Sixth edition, 2013.
2. Elmasri R., Navathe S.B., “Fundamentals of Database Systems”, Seventh Edition, Pearson, 2017.

REFERENCE BOOKS & e-RESOURCES:

1. Chris J. Date, with Hugh Darwin, Addison-Wesley, “A Guide to SQL Standard”, December 2004.
2. H Garcia-Molina, JD Ullman and Widom, Database Systems: The Complete Book, 2nd Ed., Prentice-Hall, 2008.
3. R Ramakrishnan, J Gehrke, Database Management Systems, 3rd Ed., McGraw-Hill, 2014.
4. W3School SQL Tutorial@ https://www.w3schools.com/sql

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-A
CO 34007: COMPUTER NETWORKS

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	1	2	4	1	70	30	40	60	200

PRE-REQUISITES:EC24509: Digital Communication

COURSE OBJECTIVES: This course will help students to understand different network technologies, various challenges in its implementation and solutions to it.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Explain different terminologies of computer network and compare the architecture of networks.
2. Evaluate the MAC layer performance.
3. Construct and evaluate the existing protocols at the network and transport layer.
4. Design and debug the IP networks.

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction to computer networks & their uses, Different topologies, ISO-OSI model: Layered Architecture, Peer-to-Peer processes and encapsulation, Function and Services of OSI layers; The Physical layer: Digital Signals, Transmission Impairments and Maximum data rate of a channel, Shennons theorem, Nyquist theorem. Transmission media: Guided and Unguided media, Circuit, Packet and Message switching, virtual Circuit. Introduction to ISDN & its components.

UNIT 2 The data link layer: Design issues & function, Error detection & correction, Forward error correction Versus Retransmission, Hamming code & CRC codes, Framing: Fixed size and Variable size Frame, Bit stuffing and Byte stuffing. Data link layer protocols: Simplest, Stop and Wait, Sliding window protocols, PPP, SLIP, HDLC. The medium access sublayer: Static and Dynamic Channel Allocation, Protocols: ALOHA Protocol, CSMA (CSMA/CD, CSMA/CA), Collision Free Protocol- Bit Map.

UNIT 3 IEEE 802 standards for LANs (IEEE 802.3, IEEE 802.4, IEEE 802.5), LAN Devices: HUB, Switches- Learning, Cut-Through and store and forward switches, Bridges: IEEE 802.x to IEEE 802.y, Spanning Tree, Remote Bridge. Internetworking Devices: Routers & gateways. The network layer: Design issues and functions, Internal organization (Virtual Circuit & Datagrams).

UNIT 4 Routing algorithms: Shortest path routing, Flooding, LSR, Distance Vector Routing, Hierarchical Routing. Introduction to TCP/IP Protocol stack: Protocol Architecture, Classful IP addressing, ARP, RARP, IP Datagrams with options and its delivery, ICMP.

UNIT 5 Subnet, Supernet, CIDR. Transport Layer: Congestion control, Load Shedding, Jitter control, addressing and multiplexing, Connection establishment and connection release, flow control. Application layer: Introduction to DNS and Email.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Tanenbaum A. S., “Computer Networks”, Pearson Education, 4th edition, 2003.
2. Behrouz A Forouzan, “Data communication and networking”, TMH publication, 4th edition, 2006.
3. Comer, “Internetworking with TCP/ IP Vol-1”, Pearson education, 4th Edition, 2002.

REFERENCE BOOKS:

1. Peterson & Davie, “Computer Networks”, Thomson Learning.
2. W. Richard Stevens, “TCP/IP Illustrated Vol-1 ” Addison-Wesley.
3. Craig Zacker, “Networking The Complete Reference”, TMH, 2001.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-A
CO 34014: AGILE SOFTWARE METHODOLOGY

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: NIL

COURSE OBJECTIVES:

1. Understand the differences between conventional and agile approaches.
2. Learn the background and origins of various agile concepts and methodologies.
3. Learn about Scrum development.
4. Understand frameworks and practices used by agile teams.
5. Learn about agile ways of gathering requirements, estimation, release planning, performance metrics, and scaling.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Describe the fundamental principles and practices associated with software development process models.
2. Compare and contrast agile software development model with traditional development models.
3. Apply techniques and skills to build and mentor agile projects for effective software development using scrum.
4. Adapt existing agile testing techniques and knowledge to implement agile projects.

COURSE CONTENTS:

THEORY:

UNIT 1 **Software Development Methodology:** Fundamentals of Software Engineering Concepts and Process; Software Development Life Cycle: Important Steps and Effort Distribution; Overview of Software Development Models like Prototype Model, Incremental Model, Spiral Model, RAD.

UNIT 2 **Fundamentals of Agile Development Methodology**: The Genesis of Agile, Introduction and background, Agile Manifesto and principles, Agile development Lifecycle, Agile Development Methods: Adaptive Software Development (ASD), Dynamic Systems Development Methods (DSDM), Scrum, Extreme Programming (XP): XP lifecycle, Feature Driven development, Lean Software Development, Kanban, Agile project management, Test Driven Development, Key Principles, Examples, and Tools & Techniques for each Agile development methods.

UNIT 3 **Agile Requirements, Design and Development:** Impact of Agile Processes in Requirement Engineering Requirements Elicitation and Management, Agility in Design, Agile Architecture, Agile Design Practices, Role of Design Principles, Agile Product Development, and Automated build tools, Continuous Integration, Continuous

Deployment, Refactoring, Team Dynamics and Collaboration.

UNIT 4 **Agile Framework:Scrum:** Introduction to Scrum, Agile Principles - Sprints Introduction, User Stories and Product Backlog, Estimation, Velocity, Burndown chart, Sprint Zero, Roles - Team Management and Structures, Product Owner, ScrumMaster / Team Lead, Implementation Team Members, Planning in Scrum - Planning, Planning Stakeholders, Planning Types (Portfolio, Product and Sprint), Sprint phases/meeting - Sprint Planning, Sprint Review, Sprint Retrospective, Product Demo, Daily Scrum calls.

UNIT 5 **Agile Testing and Review**: Agile Testing Principles, Practice and Processes, Difference between Testing in Traditional and Agile Approaches, Agile testing methods, techniques and tools, Estimating Test Efforts, Agile Metrics and Measurements, Agile Control: the 7 control parameters; Product Quality, Agile approach to Risk, Agile Approach to Configuration Management, Agility and Quality Assurance. Case study using any one of the framework.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Robert C. Martin, Agile Software Development- Principles, Patterns and Practices, Prentice Hall, 2013.
2. Kenneth S. Rubin, Essential Scrum: A Practical Guide to the Most Popular Agile Process, Addison Wesley, 2012.
3. James Shore and Shane Warden, The Art of Agile Development, O’Reilly Media, 2007.
4. Craig Larman, ―Agile and Iterative Development: A manager’s Guide, Addison-Wesley, 2004.
5. Janet Gregory, Agile Testing: A Practical Guide for Testers and Agile Teams By Lisa Crispin, Addison Wesley.

REFERENCE BOOKS:

1. SudiptaMalakar, Agile Methodologies In-Depth: Delivering Proven Agile, SCRUM and Kanban Practices for High-Quality Business Demands, BPB Publications, 2021.
2. Maria Paasivaara and Philippe Kruchten, Agile Processes in Software Engineering and Extreme Programming, Springer; 1st edition, 2020.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-A
CO 34451: SKILL DEVELOPMENT LAB

* Periods per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
-	-	2	-	1	-	-	20	30	50

PRE-REQUISITES: CO10504: Computer Programming, CO24497: Programming Practices

COURSE OBJECTIVES: Learn various tools and programming languages prevalent in solving real world problems. Choose appropriate machine learning techniques and tools to solve problems.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Describe in-depth about programming languages, tools and packages used in machine learning.
2. Examine the nature of a problem at hand and determine which machine learning tool can solve it efficiently enough.
3. Solve and implement real world problems using different machine learning tools.
4. Develop understanding of the strengths and weaknesses of popular machine learning tools.

COURSE CONTENTS:

	Python
ACTIVITY 1	Review of python: variables, datatypes, operators, list, tuples, sets, dictionary etc. control structures and functions.
ACTIVITY 2	Introduction to anaconda: spyder, Jupyter, notebook, etc.
ACTIVITY 3	Data analysis packages: Numpy, pandas etc.
ACTIVITY 4	Machine learning frameworks-1: TensorFlow, SciKit-Learn etc.
ACTIVITY 5	Machine learning frameworks-2: Keras, PyTorch etc.
	R Language
ACTIVITY 6	R Language: Basic of R language
ACTIVITY 7	R Language: ML packages
	MaTLaB
ACTIVITY 8	MatLab/SciLab: Basic of MatLab
ACTIVITY 9	MatLab ToolKit-1: Machine learning toolkit
ACTIVITY 10	MatLab ToolKit-1: Deep learning toolkit

DIRECT ASSESMENT:
ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. "Learn Python 3 The Hard Way" by Zed A. Shaw, Addison-Wesley Publisher, January 2017.
2. "Python Data Science Handbook: Essential Tools for Working with Data" by Jake VanderPlas, O'Reilly Publication, 1st edition, 2016.
3. Hadley Wickham and Garrett Grolemund ,R for data science, O'Reilly publication, January 2017.
4. RudraPratap, Getting Started with MATLAB, OXFORD publication, 2019.

REFERENCE BOOKS:

1. S Lovelyn Rose, L Ashok Kumar, D KarthikaRenuka, "Deep Learning Using Python", Wiley, 2019.
2. "MATLAB for Machine Learning" by Giuseppe Ciaburro, Packt Publishing, August 2017.
3. "MATLAB Deep Learning: With Machine Learning, Neural Networks and Artificial Intelligence" by Phil Kim, Apress Publisher, June 2017.
4. "R for Data Science" by Garrett Grolemund, Hadley Wickham, O'Reilly Publisher, December 2016.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-A
CO 34452: DESIGN THINKING LAB-II

* Periods per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
-	-	2	-	1	-	-	20	30	50

PRE-REQUISITES: CO24991: Design thinking Lab-I

COURSE OBJECTIVES:

Design Thinking (DT) helps students to complete tasks and solve problems in structured ways. DT teaches how students can learn to self-manage tasks and projects while applying skills, strategies and dispositions to achieve reliable processes which facilitate meaningful outcomes.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Develop a fundamental understanding of the Design Process and how it can be applied to innovation and creative problem-solving.
2. Build empathy for target audiences by applying different techniques.
3. Develop and test innovative ideas through a rapid iteration cycle.
4. Create physical prototypes/a visual representation of an idea.

COURSE CONTENTS:

LAB 1: Review of various activities conducted during Empathize Phase and Define Phase.
LAB 2: Evaluation of ideas, Silent Brainstorming, Metaphors for ideation.
LAB 3: Creative principles and techniques.
LAB 4: CREATE and What-If tool for Ideation.
LAB 5: Prototyping, Lean Startup Method for Prototype Development,
LAB 6: Visualization and presentation techniques.(Best Practices of presentation.)
LAB 7: Test prototyping and Validation.
LAB 8: Kano Model, Desirability Testing.
LAB 9: Agility for Design Thinking
LAB 10: Case study

DIRECT ASSESMENT:

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.

3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Tim Brown, Clayton M. Christensen , IndraNooyi and Vijay Govindarajan, HBR's 10 Must Reads on Design Thinking: by Harvard Business Review Press, 2020.
2. Dr. BalaRamadurai, " Karmic Design Thinking", First Edition, 2020.
3. Jeanne Liedtka, Tim Ogilvie, and Rachel Brozenske, The Designing for Growth Field Book: A Step -by - Step Project Guide (Columbia University Press, 2014).

REFERENCE BOOKS:

1. 101 Design Methods: A Structured Approach for Driving Innovation in Your Organization by Vijay Kumar.
2. Human-Centered Design Toolkit: An Open-Source Toolkit to Inspire New Solutions in the Developing World by IDEO.
3. This is Service Design Thinking: Basics, Tools,Cases by Marc Stickdorn and Jakob Schneider.
4. Ulrich, Karl T. Design: Creation of artifacts in society. Karl T. Ulrich, 2011.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-B
CO34553: MACHINE LEARNING

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: MA10501: Mathematics - II, CO34451: Skill Development Lab

COURSE OBJECTIVES: The objective of this course is to impart necessary knowledge of different machine learning techniques and develop programming skills required to build machine learning based applications.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Describe in-depth about theories, methods, and algorithms in machine learning.
2. Find and analyze the optimal hyper parameters of the machine learning algorithms.
3. Examine the nature of a problem at hand and determine whether machine learning can solve it efficiently.
4. Solve and implement real world problems using machine learning.

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction to machine learning (ML): Basics of ML, History of ML, Evolution of ML, ML Models, Learning and testing models, ML Algorithm and Convergence, ML Techniques, Types of ML, supervised and unsupervised learning, classification and clustering, Applications of ML, Bias-Variance tradeoff.

UNIT 2 Neural Networks: McCulloch Pitts Neuron models, Activation Functions, Loss Functions, perceptron, Gradient Descent, Multilayer neural networks: back-propagation, backpropagation calculus, Initialization, Training rules, issues in back-propagation, Bayesian Learning, Competitive learning and self-organization map.

UNIT 3 Support Vector Machines(SVM): SVM Formulation, Interpretation & Analysis, hard and soft margin, Hinge loss, SVM dual, SVM tuning parameters, SVM Kernels, twin SVM.

UNIT 4 Clustering: K-Means Clustering, Mean Shift Clustering, Aagglomerative clustering, Association Rule Mining, Partition Clustering, Hierarchical Clustering, Birch Algorithm, CURE Algorithm, Density-based Clustering, Gaussian Mixture Models, and Expectation Maximization. Parameters estimations – MLE,MAP.

UNIT 5 Learning Theory: Probably Approximately Correct (PAC) Model, PAC Learnability, Agnostic PAC Learning, Theoretical analysis of machine learning problems and algorithms, Generalization error bounds, VC Model, ML Tools.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

3. Feedback of students on attainment of cos.
4. Feedback of students on classroom learning.
5. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Tom Mitchell, Machine Learning, McGraw-Hill, 1997.
2. Bernhard Schölkopf and Alexander J. Smola, Learning with Kernels, MIT Press, 2002.
3. Shai Shalev-Shwartz and Shai Ben-David, Understanding Machine Learning: From Theory to Algorithms, Cambridge University Press.,2014

REFERENCE BOOKS:

1. Christopher M. Bishop, Pattern Recognition and Machine Learning, Cambridge press, 2010.
2. Leonard Kaufman and P. J. Rousseau. Finding groups in data: An introduction to cluster analysis, Wiley, 2005
3. NelloCristianini and John Shawe-Taylor, An Introduction to Support Vector Machines, Cambridge University Press, 2000.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-B
CO 34554: FOUNDATION OF INFORMATION SECURITY

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	3	-	70	30	-	-	100

PRE-REQUISITES: CO34007:Computer Network

COURSE OBJECTIVES:

COURSE OUTCOMES:
After completing the course student should be able to:

1. ***Analyze and model the Symmetric cryptographic algorithms for information security***
2. ***Model the Public Key cryptosystems***
3. ***Understand how to apply access control techniques to authenticate the data***
4. ***Apply concepts of Email, IP and web security in application development.***

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction: Needs for Security; Basic security terminologies e.g. threats, vulnerability, exploit etc.; Security principles(CIA), authentication, nonrepudiation; security attacks and their classifications; Mathematical foundation - Prime Number; Modular Arithmetic; Fermat's and Euler's Theorem; The Euclidean Algorithms; The Chinese Remainder Theorem; Discrete logarithms.

UNIT 2 Symmetric Key Cryptography: Classical cryptography – substitution, transposition and their cryptanalysis; Symmetric Cryptography Algorithm – DES, 3DES, AES etc.; Modes of operation: ECB, CBC etc.; Cryptanalysis of Symmetric Key Ciphers: Linear Cryptanalysis, Differential Cryptanalysis.

UNIT 3 Asymmetric Key Cryptography: Key Distribution and Management, Diffie-Hellman Key Exchange algorithm; Asymmetric Key Cryptography Algorithm– RSA, ECC etc.; Various types of attacks on Cryptosystems.

UNIT 4 Authentication & Integrity – MAC, Hash function, SHA, MD5, HMAC, Digital signature and authentication protocols; Authorization; Access control mechanism; X.509 Digital Certificate.

UNIT 5 E-mail, IP and Web Security: E-mail security – PGP, MIME, S/MIME; IP security protocols; Web security – TLS, SSL etc.; Secure Electronic Transaction(SET); Firewall and its types; Introduction to IDPS; Risk Management; Security Planning.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Michael E. Whitman, Herbert J. Mattord, “Principles of Information Security”, 6th Edition, Cengage Learning.
2. Stallings William, “Cryptography and Network Security - Principles and Practice”, 7th Edition, Pearson.

REFERENCE BOOKS:

1. Roberta Bragge, Mark Rhodes, Keith Straggberg, “Network Security the Complete Reference”, Tata McGraw Hill Publication,

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-B
CO 34563: DESIGN & ANALYSIS OF ALGORITHMS

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: CO24507:Data Structures

COURSE OBJECTIVES: Analyze the asymptotic performance of algorithms. Demonstrate a familiarity with major algorithms and data structures. Apply important algorithmic design paradigms and methods of analysis. Design efficient algorithms for common engineering design situations.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Describe complexity of an algorithm and various notations to represent it.
2. Apply and evaluate different algorithm design techniques for getting the effective solutions of specified problems.
3. Compare and contrast different graph algorithms with its applications.
4. Explain computability and non-computability and various complexity classes.

COURSE CONTENTS:

THEORY:

UNIT 1 Review of elementary Data Structures: Stacks, Queues, Lists, Trees, Hash, Graph. Internal representation of Data Structures, Code tuning techniques: Loop Optimization, Data Transfer Optimization, Logic Optimization, etc.

UNIT 2 Definitions of complexity, Time and Space Complexity; Time space tradeoff, various bounds on complexity, Asymptotic notation: O-notation, Ω-notation, Ɵ-notation, Recurrences and Recurrences solving techniques: Recursion-tree method and Master method, Average time analysis methods: Probabilistic methods. Amortized analysis.

UNIT 3 Design and analysis of algorithms using the brute-force, greedy, dynamic programming, divide-and-conquer and backtracking techniques.

UNIT 4 Algorithm for sorting and searching, string matching algorithm, Number-theoretic algorithms, linear programming, Matrix Manipulation algorithms, tree and Graph Algorithms.

UNIT 5 NP-hard and NP-complete problems, Approximations Algorithms, Data Stream Algorithms, Introduction to design and complexity of Parallel Algorithms.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Cormen, Leiserson, Rivest, Stein, “Introduction to Algorithms”, Second Edition, Prentice Hall of India, 2001.
2. Aho A.V., Hopcroft J.E., J. Ullman, “Design and Analysis of Computer Algorithms”, Addison Wesley, 1998.
3. Horowitz E. and Sahani, “Fundamentals of Computer Algorithms”, Galgotia Publications, 1984.

REFERENCE BOOKS:

1. Knuth D., “Fundamental algorithms: The Art of Computer programming”, Volume – I, Third Edition, Pearson Education 1998.
2. Knuth D., “Sorting and Searching: The Art of Computer programming”, Volume – III, Second Edition Pearson Education 1998.
3. John Kleinberg, Trades E., “Algorithm Design”, Pearson Education 2002.
4. A. Papoulis, S.U. Pillai, “Probability, Random Variables and Stochastic Processes”, McGraw Hill, Fourth Edition 2006.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-B
CO 34881: INTERNET OF THINGS

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
-	1	2	-	2	-	-	40	60	100

PRE-REQUISITES: CO24992: Computer Workshop, CO34007: Computer Network

COURSE OBJECTIVES:
The main objectives of this course is to design and develop the prototypes for various applications of IoT by the students.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Identify various types of sensors and actuators.
2. Develop and demonstrate programs on various types of development boards.
3. Apply concepts of communication protocols for designing programs on ESP32.
4. Solve real world problems using the concepts of IoT.

COURSE CONTENTS:

LAB 1 Study of various types of development boards like ESP32, Raspberry Pi0 etc.
LAB 2 Study and identification of various types of sensors and actuators.
LAB 3 Installation of IDE and basic programming on ESP32 like LED blinking, reading the values from the inbuilt sensors etc.
LAB 4 Interfacing of various types of sensors and actuators with ESP32, reading the values of sensors and displaying on the serial monitor and LCD etc.
LAB 5 Connecting the ESP32 with a smartphone using Bluetooth and reading the values from the sensors and controlling the actuators with the smartphone.
LAB 6 Connecting the ESP32 with a smartphone using Wi-Fi and reading the values from the sensors and controlling the actuators with smartphone and Blynk App.
LAB 7 Creating an asynchronous webserver using ESP32/RPi and sending the sensor values to the server.
LAB 8 Connecting the ESP32/RPi with any cloud platform like ThingSpeak and displaying the sensor values and controlling the actuators.
LAB 9 Installation of OS in Raspberry Pi0 and basic programming in RPi.
LAB 10 Basic application development based on the concepts of IoT and Machine Learning/DL.
LAB 11 Basic application development based on the concepts of IoT and Computer Vision/OpenCV Library.
LAB 12 Mini-Project using the kits available in the Lab.

DIRECT ASSESMENT:

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. CunoPfister, “Getting Started with Internet of Things”, O’Reilly Publication.
2. RajkumarBuyya, “Internet of Things – Principles and Paradigms”, Elesevier Publication, 2016
3. Rajkamal “Internet of Things : Architecture and Design Principles”, McGraw Hill Education, 2017

REFERENCE BOOKS:

1. Timothy Chou, “Precision: Principles, Practices and Solution for the Internet of Things, 2020
2. Hanes David “IoT Fundamentals, Networking Technologies, Protocols, and Use Cases for the Internet of Things”, Pearson Education, 2017

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-B (ELECTIVE-1)
CO 34601: DATA SCIENCE & ENGINEERING

Hoursper Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: MA10501: Mathematics-II, CO34451: Skill Development Lab

COURSE OBJECTIVES: This course focuses on enabling students to learn, understand, and use data analytics, which include the study of statistical properties of data and preparing it appropriately for applying machine learning techniques focusing on performance evaluation.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Describe in-depth about data, data types, data models and data distributions.
2. Compare and analyze different data preprocessing and data visualization techniques.
3. Examine the nature of the data at hand and determine the best suitable data engineering technique/algorithm.
4. Solve and implement real world problems using data engineering.

COURSE CONTENTS:
THEORY:

UNIT 1 Understanding Data: Types of data, Data Quality, Data Distributions, Data models, Data Wrangling and Exploratory Analysis, Introduction to contemporary tools. Data mathematics: probability, linear programming, statistics, calculus, metrics calculus: SVD, Eigen vectors, factorization.

UNIT 2 Data Preprocessing: Data Transformation & Cleaning, Aggregation, Sampling, Dimensionality reduction, Feature subset Selection, feature creation, PCA, LDA, Discretization &Binarization, variable transformation, Data Normalization, Data similarity measures, missing values, filters, Wrapper Method ,Noise reduction techniques.

UNIT 3 Exploring Data: Summary Statistics, measures of location & spread, Multivariate summary statistics, Data Visualization: boxplots, histograms, scatterplots, features map visualization, t-SNE, learning model data visualization, OLAP & Multidimensional Data Analysis.

UNIT 4 Statistical & Probabilistic analysis of Data, Multiple hypothesis testing, Parameter Estimation methods, Confidence intervals, Correlation & Regression analysis, logistic regression, Shrinkage Methods, Lasso Regression, Bayesian statistics. L1 and L2 regularizations.

UNIT 5 Data Analysis, performance metrics, ROC curve, types of errors, Overfitting & Under fitting, evaluating performance of learning model: Holdout, Random sampling, cross validation and Bootstrap method. Bagging & boosting, Gradient Boosting, Random

Forests, Committee Machines.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Trevor Hastie, Robert Tibshirani, Jerome H. Friedman , The Elements of Statistical Learning, Springer Series in Statistics, second edition, 2008
2. Brian Shive , Data Engineering: A Novel Approach to Data Design, Technics Publications, LLC, NJ, USA, 2013
3. Field Cady, “The Data Science Handbook” , 1/e,Publisher: Wiley, 2018

REFERENCE BOOKS:

1. Pang-Ning Tan, “Introduction to Data Mining”, Pearson Edu., 2007.
2. Sinan Ozdemir, “Principles of Data Science “, 1/e, 2016 Packt Publishing Limited.
3. Peter Bruce, “Practical Statistics for Data Scientists: 50 Essential Concepts”,Shroff/O'Reilly; First edition (2017)

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-B (ELECTIVE-1)
CO 3____: COMPILER CONSTRUCTION

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: CO34002: Theory of Computation

COURSE OBJECTIVES:
This course focuses on enabling students to learn, understand, and design compiler and Interpreter, which include the study of phases of compiler, code optimization, linker, loader and lex and YACC tools.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Apply fundamental concepts of system programming and different language translators.
2. Build lexical and syntax analyzers and use them in the construction of scanners and parsers
3. Apply the operations of semantic analysis, intermediate code generation and use of different code optimization schemes.
4. Design and implement a complete working compiler for a given language including linker and loader.

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction to System Programming, Evolution and Components of System Software. Compiler structure: analysis-synthesis model of compilation, various phases of a compiler, Symbol table management, Lexical analysis: Input buffering, Specification & Recognition of Tokens, LEX.

UNIT 2 Syntax Analysis: CFGs, Top down parsing, brute force approach, recursive descent parsing, transformation of the grammars, predictive parsing, bottom up parsing, operator precedence parsing, LR parsers (SLR, LALR, LR), Parser generation.

UNIT 3 Syntax directed Translation: Construction of Syntax trees, Bottom up evaluation of S attributed definition, L-attribute definition, Top down translation, Bottom Up evaluation of inherited attributes, Recursive Evaluation, Analysis of Syntax directed definition.

UNIT 4 Code Generation & optimization: Intermediate code generation, Declarations, Assignment statements, Boolean expressions, Case statements, DAG representation of basic blocks, generate code from DAG, error handing.

UNIT 5 Linkers and Loaders: Address resolution, linking of subroutines and program relocation Text editors & Word Processors, Debug monitors, IDE, Device Drivers.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. "Compilers: Principles, Techniques, and Tools" (2 nd edition), Aho, Lam, Ullman, Sethi, 2011, Pearson, ISBN: 0133002144, 9780133002140.
2. *Compiler Construction: Principles and Practice*, by Kenneth C. Louden, PWS Publishing (Thomson), Boston, MA, 1997. ISBN: 0-534-93972-4
3. Advanced compiler Design Implementation by Steven S. Muchnick
4. The Compiler Design handbook: Optimization and Machine Code Generation by Y. N. Shrikant and Priti Shankar, Second Edition.

REFERENCE BOOKS:

1. K. L. P Mishra, N. Chandrashekaran (2003), Theory of computer science- Automata Languages and computation, 2nd edition, Prentice Hall of India, New Delhi, India.
2. Andrew W. Appel (2004), Modern Compiler Implementation C, Cambridge University Press, UK.
3. Charles N. Fischer, Richard J. leBlanc, Jr.- Crafting a Compiler with C, Pearson Education, 2008.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-B (ELECTIVE-1)
CO 3____: WIRELESS & MOBILE NETWORKS

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: CO34007: Computer Network

COURSE OBJECTIVES: This course will help students to understand different wireless and mobile network technologies, various challenges in its implementation and solutions to it.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Describe utility of various network layer protocols and modify them according to the requirement of the environment in IP networks.
2. Identify and evaluate the existing protocols at the transport layer in TCP/IP protocol stack.
3. Compare and contrast the existing wireless protocols and mobility related issues.
4. Explain the underlying architecture of mobile networks and security issues.

COURSE CONTENTS:
THEORY:

UNIT 1 Basics of data communication and TCP/IP protocol stack: ARP/RARP, IP addressing, IP Datagram format and its Delivery, Routing table format, ICMP Messages, Subnetting, Supernetting and CIDR; DNS; NAT: Private addressing and NAT, SNAT, DNAT, Application of NAT in firewall. IPV6: address structure, address space and header.

UNIT 2 Routing and switching protocols like OSPF, RIP, and BGP. Transport layer protocols: TCP- Header, Synchronization, Connection Establishment and connection release, Flow control etc, UDP- format and pseudo header. VLANS: Comparison with Real LANS, Type of VLAN, Tagging, Registration protocols.

UNIT 3 Wireless and mobile communication: IEEE 802.11. Mobile IP: Agent advertisement, Registration, Tunneling, Routing Protocols for ad hoc networks.

UNIT 4 Mobile TCP: I-TCP, S-TCP, M-TCP, Introduction WAP Protocol stack, Wireless Telecomm Networks like GSM, CDMA. Other issues like Network Security, IPsec.

UNIT 5 Introduction to sensor, wireless mess, vehicular network, hybrid networks, delay tolerant networks etc...

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Comer, "Internetworking with TCP/ IP Vol-1", 5th Edition, Addison-Wesley.
2. Jochen Schiller, "Mobile communication", 2nd Edition, Pearson education,

REFERENCE BOOKS:

1. W. Richard Stevens, "TCP/IP Illustrated Vol-1", Addison-Wesley.
2. William Stallings,"Wireless Communications and Networks", 2nd Edition Pearson education

WEB SITES:

1. www.cisco.ohio-state.edu
2. www.cs.wustl.edu/~jain Research

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-B (ELECTIVE-1)
CO 34602: OBJECT ORIENTED SOFTWARE ENGINEERING

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: CO 24057: Object Oriented Programming Systems, CO 34014: Agile Software Methodology

COURSE OBJECTIVES:
The objective of this course is to expose students to formal processes for the design,implementation and management of large software systems using object oriented software development approach.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Compare and contrast Object-Oriented software development and conventional software development.
2. Apply best practices of Rational Unified Process in software development.
3. UseObject-Oriented analysis and design for moderately realistic Object-Oriented systems.
4. Construct UML diagrams (Structural and Behavioural) using the appropriate notations for various systems.

COURSE CONTENTS:
THEORY:

UNIT 1 Review of Object Oriented Concepts and Principles: The Object Oriented Paradigm, Object Oriented Principles, Basic Concepts; Review of Software Engineering Concepts: Software Development Life Cycle, Models and Model Architectures.

UNIT 2 Introduction to Rational Unified Process: Basic Concepts, Symptoms and Root Causes of Software Development Problems, Best Practices of RUP, RUP software life cycle, 4+1 view model, Various Workflows.

UNIT 3 Introduction to UML: Notations, Relationships, Stereotypes, Study of UML based tools like Rational Rose, Poseidon, etc. Object Oriented Analysis: Conventional v/s OO analysis approach, Requirement analysis, Use case diagram, Activity diagram, Analysis class diagram.

UNIT 4 Object Oriented Design: Conventional v/s OO design approach, Design of CRC cards, Class diagram; Behavioural Modelling: Interaction Diagram, State chart Diagram; Implementation Diagram: Component Diagram, Deployment Diagram; Illustrative Case Studies like ATM, Payroll, Course and Registration System.

UNIT 5 Object Oriented Testing: Correctness and consistency of OOA and OOD models, Types of Testing, Testing Strategies and test cases for OO software process; Project Management, Rational Tool Mentors, Introduction to Design Patterns.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Roger Y. Lee, “Object-Oriented Software Engineering with UML: A Hands-On Approach”, Nova Science Publishers Inc, 15 Feb 2019.
2. Ivar Jacobson, “Object Oriented Software Engineering”, Pearson Education ,2nd Edition 2009.09 Feb 2011.
3. Stephen R. Schach, “Object Oriented Classical Software Engg.” Tata McGraw Hill, 8th edition, 01 Dec 2010.
4. Grady Booch, James Rumbaugh, Ivar Jacobson, “The Unified Modelling Language User Guide”, Pearson Education 2008.
5. Gamma G.Helm, Johnson, “Design Patterns, Elements of Reusable Object Oriented Software”, 1st Edition, Pearson, 2015.
6. Phillipe Kruchten, “The Rational Unified Process - An Introduction”, 3rd Edition, Pearson Ed., 2003.

REFERENCE BOOKS:

1. Ivon Jacobson, “Object Oriented Software Engineering”, 1st Edition, Addison Wesley, 1992.
2. Ivar J, Grady B, James R., “The Unified Software Development Process”, Pearson Education, 2003.
3. Timothy C. Lethbridge, Robert Laganiere, “Object Oriented Software Engineering”, Tata McGraw Hill, 2004.
4. IBM Rational Modules.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-B (ELECTIVE-1)
CO 3____: COMPUTER GRAPHICS

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: CO10504: Computer Programming

COURSE OBJECTIVES:

1. Interpret the mathematical foundation of the concepts of computer graphics.
2. Explain the core concepts of computer graphics, including viewing, projection, perspective, modelling and transformation in two and three dimensions.
3. Apply the concepts of colour models, lighting and shading models, textures, hidden surface elimination, anti-aliasing, and rendering.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Define basics of computer graphics and its architecture.
2. Apply concepts of transformations, window and view-ports object representation on graphics objects.
3. Explain the concept of polygon filling, polygon shading and visible surface detection techniques in computer graphics applications.
4. Build interactive graphics applications using C, C++, Python, Java etc.

COURSE CONTENTS:

THEORY:

UNIT 1 Introduction to Computer Graphics; Graphics System Architecture; Line & circle drawing Algorithms; Text display; Character and Raster graphics; Graphic Input and Output devices.

UNIT 2 2-D Transformations; Windows & Viewports; Point, Line and Polygon Clipping; Segments and Display Files; Solid area Scan Conversion; Polygon and Area Filling- Scanline polygon fill algorithm, Seed-fill algorithm; Antialiasing.

UNIT 3 3-D Graphics; 3-DTransformations; Parallel and Perspective Projection Techniques; Viewing parameters; Hidden surface elimination, Z-Buffer and Area subdivision Algorithms.

UNIT 4 Colouring, Gamma Correction, Half Toning, Color Models, Color Tables, Shading; Curves and Surfaces- Cubic Splines, Beizer Spline, B-Splines; GUI design; UI Components & Containers, Events, Screen design, Interaction using Picks, Locators, Buttons etc.

UNIT 5 Overview of Graphics standards, Application and Tools; Animation: Types of Animation, Soft object modelling; Image file format.

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Zhigang Xiang, Roy Plastock, Schaum's Outlines "Computer Graphics", 2nd edition, McGraw Hill, 2017.
2. Rajesh K. Maurya, Computer Graphics with Virtual Reality Systems, the 2nd Edition, Wiley, 2018.
3. Hearn, Baker, Carithers, "Computer Graphics with OpenGL", 4th Edition, Pearson, 2013.

REFERENCE BOOKS:

1. Foley, Vandam, Feiner, Hughes, "Computer Graphics Principles & Practices", Pearson Education 2002
2. Harrington, "Computer Graphics: A Programming Approach", 2 nd Edition, McGraw-Hill, 2014.
3. Robert Eckstein, James Elliott, Brian Cole, David Wood, Marc Loy, "Java Swing", O'Reilly, 2002.
4. Roger, "Procedural Elements of Computer Graphics", 2 nd Edition, McGraw-Hill, 2001.
5. Newman and Sproull, "Principles of Interactive Computer Graphics", McGraw-Hill, 2001.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-B (ELECTIVE-1)
CO 3____: EMBEDDED SYSTEMS

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES:

1. CO24508: Operating system
2. CO24992: Computer Workshop

COURSE OBJECTIVES:

The main objective of this course is to introduce various Microcontroller architecture and their functions.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Describe working of various types of processors and microcontrollers.
2. Explain Embedded System Architecture.
3. Implement interfaces of various IO devices and peripherals with microcontrollers.
4. Use microcontrollers to simulate modules of Real Time Operating System.

COURSE CONTENTS:

THEORY:

UNIT 1 Introduction, Definition, Characteristics & Salient Features, Classification, Overview of Embedded System Architecture & Recent Trends, Application Areas.

UNIT 2 **Embedded System Architecture:** Hardware Architecture: Basic building blocks, Devices and Communication Interfaces, Software Architecture: Embedded Operating System, Applications Software & Communication Software (Firmware).

UNIT 3 **ARMCortex-M3/M4 processor:** Introduction ARM architecture and Cortex : M series, Introduction to the Tiva family viz. TM4C123x and its targeted applications, Tiva block diagram, address space, on-chip peripherals, Simplified view: block diagram, programming model: Registers, Operation modes, Exceptions and Interrupts, Reset Sequence.

UNIT 4 **ARM Cortex M4 Peripherals:** I/O pin multiplexing, GPIO control, Watchdog Timer, System Clocks and control, Hibernation Module on TM4C, Introduction to Interrupts, Interrupt vector table, interrupt programming, Timers, ADCs, UART and other serial interfaces: USB, PWM, RTC, DMA, QEI on TM4C microcontroller.

UNIT 5 **Embedded Operating System:** General Purpose Operating Systems vs Embedded Operating Systems, Overview of Embedded Operating Systems, detailed study of Real Time Operating System(RTOS) such as TI RTOS, Task Scheduling, Task Synchronization, Inter- task communication concepts on Embedded TI RTOS platform,Case Studies of Typical Embedded Systems : Digital Camera and Washing

Machine.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Introduction to Embedded Systems - A Cyber-Physical Systems Approach by Lee &Seshia, Second Edition, MIT Press.
2. Embedded Systems - SoC, IoT, AI and Real-Time Systems, By Rajkamal, McGraw Hill; Fourth edition.

REFERENCE BOOKS:

1. Embedded Systems: A Contemporary Design Tool by James K. Peckol, Wiley Publication.
2. Embedded / Real-Time Systems: Concepts, Design and Programming Black Book, by Dr. K.V. K Prasad, Dreamtech Press.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. III YEAR (4YDC)
SEMESTER-A (Open Elective-1)
CO 34298: ARTIFICIAL INTELLIGENCE

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	1	-	4	-	70	30	-	-	100

PRE-REQUISITES: MA10501: Mathematics-II

COURSE OBJECTIVES: To enable students to learn the basic concepts, theories, applications and state-of-the-art techniques of artificial intelligence.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Describe fundamentals of Artificial Intelligence (AI) and its foundations.
2. Apply formal methods of knowledge representation, logic and reasoning for problem solving.
3. Apply basic principles of AI in solutions using inference, perception, knowledge representation, and learning concept.
4. Demonstrate awareness and understanding of various applications of AI techniques in intelligent agents, expert systems and other machine learning models.

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction to Artificial Intelligence (AI): Basics, The Foundations of AI, History of AI, Intelligent Agents; Problem Solving by Searching: Problem spaces and search, Uninformed strategies-BFS, DFS, Dijkstra, Informed Strategies- A* Search, Heuristic Functions, Hill-climbing, Adversarial Search- Minimax Algorithm, Alpha-beta Pruning, Constraint Satisfaction Problems.

UNIT 2 Knowledge Representation: Knowledge-Based Agents, Representations and Mappings, Issues, Propositional Logic and Predicate Logic, First Order Logic and Inference, Ontological Engineering, Semantic Networks, Categories and Objects, Events.

UNIT 3 Reasoning: Quantifying Uncertainty, Baye’s Rule and Its Use, Naive Bayes Models, Probabilistic Reasoning: Semantics and Inference in Bayesian Networks, Probabilistic Reasoning over Time: Time and Uncertainty, Inference in Temporal Models, Hidden Markov Models and other Models.

UNIT 4 Expert Systems: Architecture of Expert Systems, Features, Roles of Expert Systems: Knowledge Acquisition, Meta knowledge, Heuristics, Knowledge Representation in Expert Systems; Typical Expert Systems: MYCIN, DART, XCON etc.; Expert System Tools and Shells.

UNIT 5 AI Applications: Case Study of Various Applications of AI in Game Playing, Natural Language Processing, Computer-Vision, Robotics etc.

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Russel,S., and Norvig,P., "Artificial Intelligence: A Modern Approach", 4th Edition, 2020, Pearson.
2. Elaine Rich, Kevin Knight,Shivashankar B. Nair, "Artificial Intelligence", 3rd Edition, 2008, McGraw-Hill International.
3. Nils J. Nilsson, Artificial Intelligence: A New Synthesis, 2015, Morgan-Kauffman, Morgan-Kaufmann.

REFERENCE BOOKS:

1. Janakiraman, K.Sarukesi, 'Foundations of Artificial Intelligence and Expert Systems', 2005, Macmillan Series in Computer Science.
2. W. Patterson, 'Introduction to Artificial Intelligence and Expert Systems', 5th Edition, 2003, Prentice Hall of India.

DEPARTMENT OF COMPUTER ENGINEERING
B. E. III YEAR (4YDC)
SEMESTER-B (OPEN CATEGORY SUBJECT -II)
CO34701: ANDROID APPLICATION DEVELOPMENT

*Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
2	1	2	3	1	70	30	40	60	200

PRE-REQUISITES: Knowledge of Core Java, Object Oriented Programming Systems (OOPS), DBMS and SQL

COURSE OBJECTIVES:
This course enables a student in designing and building android applications.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Describe the basic building blocks of Android and android architectureframework.
2. Apply concept of basic UI design, for application Views, ViewGroups and different Layouts and widgets in Android for Application Development.
3. Use styles and themes and create Android apps with databases using SQLite and Implement and use various services in Android applications.
4. Successfully create, test and deploy working android apps on Google Play Store.

COURSE CONTENTS:

THEORY:

Unit 1: Introduction to Android and its history, architecture, application life cycle, Setting up development environment: studio, Emulator, flutter; Dalvik Virtual Machine & .apk file extension, Android API levels (Versions & version names) , Detailed Application Structure, AndroidManifest.xml; Basic Building blocks in android app: Activities, events and events handlers, Layouts& Drawable Resources.

Unit 2: Resources and Uses-permission: Normal and critical, security aspects. Basic UI design : Form widgets , Text Fields , Menu , Images and media, button , Alert Dialogs , Toast and Popup etc.

Unit 3: Navigation and Inter activity communication: Intents - Explicit Intents , Implicit intents. SQLite: SQLiteOpenHelper , SQLite Database , Cursor Reading and updating. Adapters.

Unit 4: Services : Overview of services in Android, Service lifecycle. Implementing Service: Inter Process Communication and AIDL, Network connectivity. Push Notifications: Broadcast Receivers, Services and Notifications. Sensors: Android Sensor Framework. Firebase for JSON database and notifications,Testing app.

Unit 5: overview of flutter, history, architecture, advantages, basic building blocks, widgets, child and children, images, icons, media, state management, connecting to native services, navigation, internet connectivity, intent equivalent in flutter.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Android Programming for Beginners: Build in-depth, full-featured Android apps starting from zero programming experience, 3rd Edition, John Horton, Packt Publishing. 2021, ISBN 978-1-80056-343-8
2. Android App Development in Android Studio: Java + Android Edition for Beginners – First Edition, J. Paul Cardle, 2017, ISBN 978-1542885843

REFERENCE BOOKS:

USEFUL WEBSITES :

1. https://info448-s17.github.io/lecture-notes/services.html
2. https://docs.flutter.dev/reference/tutorials
3. https://developer.android.com/
4. www.freecodecamp.com
5. www.linkedinlearning.com

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-2)
CO 4____: COMPUTATIONAL INTELLIGENCE

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	-	-	70	30	40	60	200

PRE-REQUISITES: CO10504:Computer Programming, CO34451: Skill Development Lab

COURSE OBJECTIVES:

1. To introduce basic concepts, theories and techniques of computational intelligence.
2. Help students to learn the applications of computational intelligence techniques in the diverse fields of science, engineering, medicine, finance etc.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Describe in-depth about theories, methods, and algorithms in computational Intelligence.
2. Compare and contrast traditional algorithms with nature inspired algorithms.
3. Examine the nature of a problem at hand and determine whether a computational intelligent technique/algorithm can solve it efficiently enough.
4. Design and implement Computation Intelligence algorithms and approaches for solving real-life problems.

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction to Computational Intelligence (CI): Basics of CI, History of CI, Adaptation, Learning, Self Organization and Evolution, CI and Soft Computing, CI Techniques; Applications of CI; Decision Trees: Introduction, Training Decision Trees, Evaluation, Splitting Criteria, Decision Tree Induction Algorithms.

UNIT 2 Evolutionary Computation: Genetic Algorithms: Basic Genetics, Concepts, Working Principle, Creation of Offsprings, Encoding, Fitness Function, Selection Functions, Genetic Operators-Reproduction, Crossover, Mutation; Genetic Modeling, Benefits; Problem Solving; Introduction to Genetic Programming, Evolutionary Programming, and Evolutionary Strategies.

UNIT 3 Fuzzy System: Fuzzy Sets: Formal Definitions, Membership Functions, Fuzzy Operators, Fuzzy Set Characteristics, Fuzzy Relations and Composition, Fuzziness and Probability; Fuzzy Logic and Reasoning: Fuzzy Logic, Fuzzy Rules and Inferencing; Fuzzy Controllers: Components of Fuzzy Controllers, Types, Defuzzification.

UNIT 4 Rough Set Theory: Introduction, Fundamental Concepts, Knowledge Representation, Set Approximations and Accuracy, Vagueness and Uncertainty in Rough Sets, Rough Membership Function, Attributes Dependency and Reduction, Application Domain, Hidden Markov Model (HMM), Graphical Models, Variable Elimination, Belief Propagation, Markov Decision Processes.

UNIT 5 Swarm Intelligence: Introduction to Swarm Intelligence, Swarm Intelligence Techniques: Ant Colony Optimization (ACO): Overview, ACO Algorithm; Particle Swarm Optimization (PSO): Basics, Social Network Structures, PSO Parameters and Algorithm; Grey wolf optimization(GWO); Application Domain of ACO and PSO; Bee Colony Optimization etc.; Hybrid CI Techniques and applications; CI Tools.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Russell C. Eberhart and Yuhui Shi, Computational Intelligence: Concepts to Implementations, Morgan Kaufmann Publishers, 2007.
2. Andries P. Engelbrecht, Computational Intelligence: An Introduction, Wiley Publishing, 2007.
3. David E. Goldberg, Genetic Algorithm in Search Optimization and Machine Learning, Pearson Education, 2009.
4. Jagdish Chand Bansal, Pramod Kumar Singh, Nikhil R. Pal, Evolutionary and Swarm Intelligence Algorithms, Springer Publishing, 2019.
5. S. Rajasekaran, G.A. VijayalakshmiPai, “Neural Networks, Fuzzy Logic, Genetic Algorithms Synthesis and Applications”, PHI, 2003.

REFERENCE BOOKS:

1. James M. Keller; Derong Liu; David B. Fogel, Fundamentals of Computational Intelligence: Neural Networks, Fuzzy Systems, and Evolutionary Computation, Wiley-IEEE Press, 2016.
2. Russell C. Eberhart, Yuhui Shi, and James Kennedy, Swarm Intelligence, The Morgan Kaufmann Series in Artificial Intelligence, 2001.
3. Lech Polkowski, Rough Sets: Mathematical Foundations (Advances in Intelligent and Soft Computing Book), Physica, 2013.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-2)
CO 4____: ADVANCED DATA STRUCTURES

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: CO24507: Data Structures, CO34563: Design and Analysis of Algorithms

COURSE OBJECTIVES:
To provide a comprehensive skill for designing and implementation of data organization in real-world application development using advanced data structures..

COURSE OUTCOMES:
After completing the course student should be able to:

1. Explain the basic concepts of advanced data structures and algorithms.
2. Describe advanced concepts of lists, trees and graphs and their implementation.
3. Use data structures to efficiently organize data for improving performance of the system.
4. Design and implement state-of-art data structures algorithms for solving real life problems.

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction to advanced data structures: Review of data structure fundamentals, operations, algorithms, amortised analysis, Recurrences, Probabilistic analysis.

UNIT 2 Advanced Tree Data Structures: Disjoint sets / union-find, dynamic array, m-ary tree, Red-Black trees, Splay list and trees, interval trees, suffix tree, index tree, Self-adjusting lists, skip list, Fibonacci heap, Binomial heap, Tries

UNIT 3 Advanced Graph data structures: Fundamental concepts, definitions, operations, properties, proof styles; graph matchings; Connectivity, network flows; Traversibility; Coloring; Planarity , duality, Euler's formula, characterization; perfect graphs, matroids, Ramsay theory, extremal graphs, random graphs; Dynamic Graphs, heavy-light decomposition; Applications

UNIT 4 String Matching and Pattern matching: Introduction, The naïve string matching algorithm, Rabin Karp algorithm, Knuth-Morris-Pratt algorithm (KMP), Data structures for multidimensional data: maps, database tables, Hashing, Dictionaries

UNIT 5 Case studies of Advanced data structures in computer vision, Bio-informatics, Big data etc.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Peter Brass, Advanced Data Structures, Cambridge Pree, 2008
2. Douglas B. West, Introduction to Graph Theory, Prentice Hall of India.
3. NarsinghDeo, Graph Theory with Applications to Engineering and Computer Science. Prentice-Hall.

REFERENCE BOOKS:

1. Introduction to algorithms: Cormen, Leiserson, Rivest and Stein.
2. Data structures and algorithms: Aho, Hopcroft and Ullman.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-2)
CO 44242: CLOUD COMPUTING

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	-	-	70	30	40	60	200

PRE-REQUISITES: CO34007: Computer Network

COURSE OBJECTIVES: Students will be able to understand the concept of service oriented architecture and gain knowledge of various cloud service model, deployment model, types of virtualization. Get detailed knowledge about cloud resource management and security issues.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Describe the fundamentals of cloud computing and different types of service models and deployment models.
2. Explain the importance of virtualization and differentiate various types of hypervisors.
3. ***Analyze authentication, confidentiality and privacy issues in Cloud computing environment.***
4. ***Compare operation and economic models of various trending cloud platforms prevailing in the IT industry.***

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction to Service Oriented Architecture, RESTful services: Introduction to cloud computing: Definition, Characteristics, cloud services , Types; Software as a Service, Platform as a Service, Infrastructure as Service, Deployment model , Architecture of cloud computing , Application, benefits and limitations, basic technologies of cloud computing, Multi-tenancy and scalabilty

UNIT 2 Study concept of Virtualization, characteristics, Virtualization applications in enterprises, Pitfalls of virtualization, reference model of virtualization, Study about Hypervisor, types of hypervisor, case study of hypervisors.

UNIT 3 Data in the cloud: Relational databases, Cloud file systems: GFS and HDFS, Features and comparisons among GFS, HDFS etc, BigTable, HBase and Dynamo. Map-Reduce and extensions: Parallel computing, The Map-Reduce model: Parallel efficiency of Map-Reduce, Relational operations, Enterprise batch processing, Example/Application of Map-Reduce.

UNIT 4 Cloud security fundamentals, Vulnerability assessment tool for cloud, Privacy and Security in cloud: Cloud computing security architecture, General Issues, Trusted Cloud computing, Security challenges: Virtualization security management-virtual threats, VM Security Recommendations, VM-Specific Security techniques, Secure Execution Environments and Communications in cloud.

UNIT 5 Issues in cloud computing; implementing real time application; QOS Issues in Cloud, Dependability, data migration, streaming in Cloud. Cloud Middleware. Mobile Cloud Computing. Inter Cloud issues. A grid of clouds, Sky computing, load

balancing, Resource optimization, Resource dynamic reconfiguration, Monitoring in Cloud, Installing cloud platforms and performance evaluation, Features and functions of cloud computing platforms.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. George-Coulouris, " Distributed-Systems-Concepts-and-Design-5th-Edition", Addison-Wesley, 2012
2. Kai Hawang, Geoferrey C Fox, "Distributed and Cloud Computing", Elseveir publication, 2012
3. Judith Hurwitz, R.Bloor, M.Kanfman, F.Halper, "Cloud Computing for Dummies", Wiley India Edition
4. RajkumarBuyya, Christian Vecchiola, S. Thamaraselvi, Mastering Cloud Computing, McGraw Hill, 2013

REFERENCE BOOKS:

1. Scott Granneman, "Google Apps", Pearson, 2012
2. Tim Malhar, S.Kumaraswammy, S.Latif, "Cloud Security & Privacy", SPD, O'REILLY
3. Ronald Krutz and Russell Dean Vines, "Cloud Security", Wiley-India, 2011

RESEARCH JOURNALS:

1. Scott Granneman, "Google Apps", Pearson, 2012
2. Tim Malhar, S. Kumaraswammy, S.Latif, "Cloud Security & Privacy", SPD, O'REILLY
3. 3.Ronald Krutz and Russell Dean Vines, "Cloud Security", Wiley-India, 2011

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-2)
CO 4____: SOFTWARE ARCHITECTURES

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: CO34014: Agile Software Methodology

COURSE OBJECTIVES:

To provide a comprehensive skill for designing and implementation of enterprise application development using the software architectures.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Explain the quality requirements of a software system and fundamentals terminologies of software architecture.
2. Use fundamental principles, guidelines, architectural styles, patterns, techniques, and tools to design and analyze software architecture of real world problems.
3. Implement a software architecture to solve specific problems.
4. Appraise the quality of software architectures.

COURSE CONTENTS:

THEORY:

UNIT 1 Overview of Software development methodology and software quality model, different models of software development and their issues. Introduction to software architecture, software components and connectors, common software architecture frameworks, Architecture business cycle, architectural patterns, design patterns: Creational design patterns, Structural design patterns, Behavioral design patterns, Software quality attributes and models.

UNIT 2 Software Architectures styles: dataflow architecture, pipes and filters architecture, call-and return architecture, data-centered architecture, layered pattern architecture, agent based architecture, services and micro-services oriented architecture etc.

UNIT 3 Software architecture implementation technologies: Software Architecture Description Languages (ADLs), JSP, Servlets, struts, hibernate, EJBs; middleware: JDBC, JNDI, JMS and CORBA, Angular JS, Role of UML in software architecture.

UNIT 4 Software Architecture analysis and design: Architectural Tactics, requirements for architecture and the life-cycle view of architecture design and analysis methods, Cost Benefit Analysis Method (CBAM), Architecture Tradeoff Analysis Method (ATAM). Active Reviews for Intermediate Design (ARID), Attribute Driven Design method (ADD), architecture reuse, Domain –specific Software architecture.

UNIT 5 Software Architecture documentation: principles of sound documentation, refinement, context diagrams, variability, software interfaces. Documenting the behavior of software elements and software systems, documentation package using a seven-part template.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Bass, L., P. Clements, and R. Kazman, "Software Architecture in Practice", Second Edition, Prentice-Hall 2015.
2. Jim Keogh, "J2EE – Complete Reference", Tata McGraw Hill, 2008.
3. Dikel, David, D. Kane, and J. Wilson, "Software Architecture: Organizational Principles and Practices", Prentice-Hall, 2001.

REFERENCE BOOKS:

1. Bennett, Douglas, "Designing Hard Software: The Essential Tasks", Prentice-Hall, 1997.
2. Clements, Paul, R. Kazman, M. Klein, "Evaluating Software Architectures: Methods and Case Studies", Addison Wesley, 2001.
3. Albin, S. "The Art of Software Architecture", Indiana: Wiley, 2003.
4. Robert Mee, and Randy Stafford, "Patterns of Enterprise Application Architecture", Addison-Wesley, 2002.
5. Witt, B., T. Baker and E. Meritt, "Software Architecture and Design: Principles, Models and Methods", Nostrand Reinhold, 1994.

WEB SITES:

1. www.sei.cmu.edu/architecture/
2. www.softwarearchitectureportal.org/

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-2)
CO 4____: MULTIMEDIA SYSTEM

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: CO____: Computer Graphics

COURSE OBJECTIVES:

Students will be introduced to principles and current technologies of multimedia systems. Issues in effectively representing, processing, and retrieving multimedia data such as sound and music, graphics, image and video will be addressed. The students will gain hands-on experience in those areas by implementing some components of a multimedia streaming system as their term project. Latest Web technologies and some advanced topics in current multimedia research will also be discussed.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Explain relevance and underlying infrastructure of the multimedia systems.
2. Describe core multimedia technologies and standards like Digital Audio, Graphics, Video, VR, data transmission/compression.
3. Define various factors involved in multimedia systems performance, integration and evaluation.
4. Design and implement a multimedia system.

COURSE CONTENTS:

THEORY:

UNIT 1 **Introduction to Multimedia System:** Architecture and components, Multimedia distributed processing model, Synchronization, Orchestration and Quality of Service (QOS) architecture.

UNIT 2 **Audio and Speech, Images and Video:** Data acquisition, Sampling and Quantization, Human Speech production mechanism, Digital model of speech production, Analysis and synthesis, Psycho-acoustics, low bit rate speech compression, MPEG audio compression. Image acquisition and representation, Bilevel image compression standards: ITU Group III and IV standards, JPEG image compression standards, MPEG video compression standards.

UNIT 3 **Multimedia Communication:** Fundamentals of data communication and networking, Bandwidth requirements of different media, Real time constraints: Audio latency, Video data rate, multimedia over wired and wireless networks, Multimedia conferencing.

UNIT 4 **Hypermedia presentation:** Authoring and Publishing, Linear and non-linear presentation, Structuring Information, Different approaches of authoring hypermedia documents, Hyper-media data models and standards.

UNIT 5 **Multimedia Information Systems:** Operating system support for continuous media applications: limitations is usual OS, New OS support, Media stream protocol, file system support for continuous media, multimedia storage data models for multimedia and hypermedia information.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Ralf Steinmetz and Klara Nahrstedt, Multimedia Systems, Springer.
2. J. D. Gibson, Multimedia Communications: Directions and Innovations, Springer.
3. K. Sayood, Introduction to Data Compression, Morgan-Kaufmann.
4. A. Puri and T. Chen, Multimedia Systems, Standards, and Networks, Marcel Dekker.
5. BorivojeFurht, Handbook of Multimedia Computing, CRC Press.

REFERENCE BOOKS:

1. Colton McAnlis, AleksHaecky, Understanding Compression, 2016, O'Reilly Media, Inc.
2. Vic Costello, Multimedia Foundations, 2nd Edition, 2016, Routledge Publishers
3. Ling Guan, Multimedia Image and Video Processing, 2nd Edition, 2017, CRC Press.
4. Benny Bing, Next-Generation Video Coding and Streaming, 2015, Wiley Publications.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-2)
CO 4____: VLSI SYSTEM DESIGN

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES:

1. EC_____: Digital Electronics
2. EC24509: Digital Communication

COURSE OBJECTIVES:

The main objective of this course is to introduce the fundamental concepts and structures of designing VLSI systems including CMOS devices and circuits

COURSE OUTCOMES:

After completing the course student should be able to:

1. Describe basic terminologies of VLSI system.
2. Explain the concepts of Physical Design Process like partitioning, Floor planning, Placement and Routing.
3. Apply mathematical methods and circuit-analysis models in analysis of CMOS.
4. Apply CMOS technology-specific layout rules in the placement and routing of transistors.

COURSE CONTENTS:

THEORY:

UNIT 1 Introduction to VLSI Design, Different types of VLSI design styles: Full custom, standard cell based, gate array based, programmable logic, field programmable gate arrays etc., VLSI Design flow.

UNIT 2 CMOS logic: PMOS, NMOS and CMOS, Electrical characteristics, operation of MOS transistors as a switch and an amplifier, MOS inverter. Stick diagram, design rules and layout, delay analysis.

UNIT 3 Different types of MOS circuits: Dynamic logic, BiCMOS, pass transistors etc. CMOS process, Combinational logic cells, Sequential logic cells, Datapath logic cells, I/O cells. Logical effort, gate array, standard cell and datapath cell design.

UNIT 4 Memristor-based logic design: design styles, crossbar implementation. Introduction to hardware description language (HDL) Verilog/VHDL. A logic synthesis example.

UNIT 5 Floor-planning and Placement: I/O and power planning, clock planning. Routing: global and detailed. Example design technique: mapping of architecture to silicon.

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. D.A. Pucknell and K. Eshraghian, Basic VLSI Design, PHI Learning Private Limited, 2013.
2. J. Bhasker, Verilog HDL Synthesis: A Practical Primer, B.S. Publications, 1998.

REFERENCE BOOKS:

1. N.H.E. Weste, D. Harris and A. Banerjee, CMOS VLSI Design: A Circuits and Systems Perspective, Third Edition, Pearson, 2006.
2. Wayne Wolf, “Modern VLSI Design ", 2nd Edition, Prentice Hall,1998.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-3)
CO 44251: DEEP LEARNING

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: CO 34451: Skill Development Lab, CO34553: Machine Learning

COURSE OBJECTIVES: Introduce deep learning fundamentals and major algorithms, the problem settings, and their applications to solve real world problems.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Describe in-depth about theories, fundamentals, and techniques in Deep learning.
2. Identify the on-going research in computer vision and multimedia field.
3. Evaluate various deep networks using performance parameters.
4. Design and validate deep neural network as per requirements.

COURSE CONTENTS:
THEORY:

UNIT 1 History of Deep Learning, Deep Learning Success Stories, review of Neuron model, activation functions, Perceptron Learning, Multilayer Perceptrons (MLPs), Feedforward Neural Networks, Backpropagation, weight initialization methods, Batch Normalization, Representation Learning, GPU implementation, Decomposition – PCA and SVD.

UNIT 2 Deep Feedforward Neural Networks, Gradient Descent (GD), Momentum Based GD, Nesterov Accelerated GD, Stochastic GD, AdaGrad, Adam, RMSProp, Auto-encoder, Regularization in auto-encoders, Denoising auto-encoders, Sparse auto-encoders, Contractive auto-encoders,Variational auto-encoder, Auto-encoders relationship with PCA, Dataset augmentation.

UNIT 3 Introduction to Convolutional neural Networks (CNN) and its architectures, CCN terminologies: ReLu activation function, Stride, padding, pooling, convolutions operations, Convolutional kernels, types of layers: Convolutional, pooling, fully connected, Visualizing CNN, CNN examples: LeNet, AlexNet, ZF-Net, VGGNet, GoogLeNet, ResNet, RCNNetc. Deep Dream, Deep Art. Regularization: Dropout, drop Connect, unit pruning, stochastic pooling, artificial data, injecting noise in input, early stopping, Limit Number of parameters, Weight decay etc.

UNIT 4 Introduction to Deep Recurrent Neural Networks and its architectures, Backpropagation Through Time (BPTT), Vanishing and Exploding Gradients, Truncated BPTT, Gated Recurrent Units (GRUs), Long Short Term Memory (LSTM), Solving the vanishing gradient problem with LSTMs, Encoding and decoding in RNN network, Attention

Mechanism, Attention over images, Hierarchical Attention, Directed Graphical Models.

UNIT 5 Introduction to Deep Generative Models, Restricted Boltzmann Machines (RBMs), Gibbs Sampling for training RBMs, Deep belief networks, Markov Networks, Markov Chains, Autoregressive Models: NADE, MADE, PixelRNN, Generative Adversarial Networks (GANs), Applications of Deep Learning in Object detection, speech/ image recognition, video analysis, NLP, medical science etc.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Ian Goodfellow, YoshuaBengio and Aaron Courville; Deep Learning, MIT Press, 2017.
2. Charu C. Aggarwal "Neural Networks and Deep Learning: A Textbook", Springer, 2019.
3. Francois Chollet, "Deep Learning with Python", First Edition, Manning Publications, 2018.

REFERENCE BOOKS:

1. Aurelien Geon, “Hands-On Machine Learning with Scikit-Learn and Tensorflow: Concepts, Tools, and Techniques to Build Intelligent Systems”, First Edition, O'Reilly publication, 2017.
2. Andreas Muller, "Introduction to Machine Learning with Python: A Guide for Data Scientists", First edition, O'Reilly Edition, 2016.
3. Adam Gibson, Josh Patterson, "Deep Learning: A Practitioner's Approach", First Edition, O'Reilly publication, 2017.

B. E. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-III)
CO 44252 : BIG DATA

*Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PER-REQUISITES: Statistics, Java and DBMS.

COURSE OBJECTIVES: Students will be able to understand the challenges of Big data and gain knowledge about Hadoop Eco System and develop applications using data processing and analytics to solve real world problems.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Explain the concept and challenges of Big data and Demonstrate knowledge of big
   data analytics.
2. Examine Hadoop Eco System and develop Big Data Solutions using it.
3. Experiment and evaluate various large-scale analytics tools like Spark, Kafka, Hbase.
4. Apply andAnalyze machine learning algorithms on Big Data.

COURSE CONTENTS:
THEORY:

UNIT 1: Introduction to Big data, Big data characteristics, Types of big data, Traditional versus Big data, Evolution of Big data, challenges with Big Data, Technologies available for Big Data, Infrastructure for Big data, Use of Data Analytics, Desired properties of Big Data system.

UNIT 2: Introduction to Hadoop, Core Hadoop components, Hadoop Eco system, Hadoop limitations, RDBMS Versus Hadoop, Hadoop Distributed File system, Processing Data with Hadoop, MapReduce Concept, Mapreduce Programming, Managing Resources and Application with Hadoop YARN.

UNIT 3: Introduction to Apache Spark*,* Parallel programming with spark, spark built-in libraries. *Design of Key-Value Stores, Data Placement Strategies,* CAP theorem*, Consistency Solutions, CQL, Design of Zookeepaer.* Introduction to Hive, Introduction to Pig

UNIT 4: Introduction to NoSQL, NoSQL Business Drivers, NoSQL Data architectural patterns-Key-value based, Column-based, Document-based, Graph-based, Variations of NOSQL architectural patterns, using NoSQL to Manage Big Data, BASE, Introduction of Hbase, Spark Streaming *and Sliding window Analytics, Introduction to Kafka*

UNIT 5: *Big Data Machine Learning, ML algorithm K-means using* Map Reduce for Big Data Analytics*, Parallel K-Means* using Map Reduce on Big Data *cluster Analysis, Decision trees for Big Data Analytics, Big Data Predictive analysis..*

DIRECT ASSESMENT:
ASSESSMENT OF THEORY:

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc.(30%)
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva,
File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. RadhaShankarmani, M. Vijaylakshmi, "Big Data Analytics", Wiley, Second edition.
2. Seema Acharya, SubhashiniChellappan, "Big Data and Analytics", Wiley, First edition.
3. Thomas Erl, WajidKhattak, and Paul Buhler, “Big Data Fundamentals, Concepts, Drivers
& Techniques”, Pearson, First Edition.

REFERENCE BOOKS:

1. KaiHwang,Geoffrey C., Fox. Jack, J. Dongarra, “Distributed and Cloud Computing”,
Elsevier, First edition
2. Michael Minelli, Michele Chambers, AmbigaDhiraj, “Big Data Big Analytics”,Wiley

REFERENCE WEBSITES:

1. https://hadoop.apache.org/
2. https://www.cloudera.com/
3. noc19-cs33-Introduction-Big Data Computing :
https://www.youtube.com/watch?v=r5k_RLIpuA&list=PLFW6lRTa1g813IyYHLRP_bWJEKQDeEcSP

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-3)
CO 4____: HUMAN COMPUTER INTERACTION

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: CO24497: Programming Practices

COURSE OBJECTIVES:

1. Describe and apply user-centered design methods to conduct formative and summative evaluations.
2. Explain and apply core theories and models from the field of HCI.
3. Design and implement useful, usable, and engaging graphical computer interfaces.
4. Discuss and critique research in the field of HCI.
5. Describe special considerations in designing user interfaces for wellness.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Define the basics behind HCI design approaches and evaluate effective and usable graphical computer interfaces.
2. Apply an interactive design process and universal design principles for modeling HCI systems.
3. Design and Implement HCI approaches for solving real-life problems.
4. Analyze and discuss HCI issues in groupware, ubiquitous computing, virtual reality, multimedia, and Word Wide Web-related environments.

COURSE CONTENTS:
THEORY:

UNIT 1 FOUNDATIONS OF HCI
The Human: I/O channels, Memory, Reasoning and problem solving; The computer: Devices, Memory, processing and networks; Interaction: Models, frameworks, Ergonomics, styles, elements, interactivity, Paradigms; Components of HCI, Principles of HCI.

UNIT 2 DESIGN & SOFTWARE PROCESS
Interactive Design basics, process, scenarios, navigation, screen design , Iteration and prototyping. HCI in software process, software life cycle, usability engineering, Prototyping in practice, design rationale. Design rules, principles, standards, guidelines, rules. Evaluation Techniques, Universal Design.

UNIT 3 MODELS AND THEORIES
Basic idea, Introduction to different types of models, GOMS family of models (KLM and CMN-GOMS), Fitts' law and Hick Hyman's law , Model-based design case studies;

UNIT 4 MOBILE HCI

Mobile Ecosystem: Platforms, Application frameworks- Types of Mobile Applications: Widgets, Applications, Games- Mobile Information Architecture, Mobile 2.0, Mobile Design: Elements of Mobile Design, Tools.

UNIT 5 WEB INTERFACE DESIGN

Designing Web Interfaces , Drag & Drop, Direct Selection, Contextual Tools, Overlays, Inlays and Virtual Pages, Process Flow. Case Studies

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Human-Computer Interaction, Third Edition by Alan Dix et al, Prentice Hall (2004).
2. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, ―Human Computer Interaction, 3rd Edition, Pearson Education, 2004 (UNIT I, II & III)
3. Brian Fling, ―Mobile Design and Development, First Edition, O‗Reilly Media Inc., 2009 (UNIT - IV)
4. Bill Scott and Theresa Neil, ―Designing Web Interfaces, First Edition, O‗Reilly, 2009. (UNIT-V)

REFERENCE BOOKS:

1. Dix A., Finlay J.,Abowd G. D. and Beale R. Human Computer Interaction, 3 rd edition, Pearson Education, 2005.
2. Preece J., Rogers Y.,Sharp H.,Baniyon D., Holland S. and Carey T. Human Computer Interaction,Addison-Wesley, 1994.
3. B.Shneiderman; Designing the User Interface,Addison Wesley 2000 (Indian Reprint).

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-3)
CO 4____: REAL TIME SYSTEMS

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: CO24508: Operating System, CO34005: Data Base Management System

COURSE OBJECTIVES:

1. Develop an understanding of various Real Time systems Application
2. Obtain a broad understanding of the technologies and applications for the emerging and exciting domain of real-time systems
3. Get in-depth hands-on experience in designing and developing a real operational system.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Define concepts of Real-Time systems.
2. Discuss features and characteristics of Real Time Operating Systems and Databases.
3. Analyze various real time systems.
4. Develop architectural design for a real-time system.

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction
Definition, Typical Real Time Applications: Digital Control, High Level Controls, Signal Processing etc., Release Times, Dead-lines, and Timing Constraints, Hard Real Time Systems and Soft Real Time Systems, Reference Models for Real Time Systems: Processors and Resources, Temporal Parameters of Real Time Workload, Periodic Task Model, Precedence Constraints and Data Dependency.

UNIT 2 Real Time Scheduling
Common Approaches to Real Time Scheduling: Clock Driven Approach, Weighted Round Robin Approach, Priority Driven Approach, Dynamic Versus Static Systems, Optimality of Effective-Deadline-First (EDF) and Least-Slack-Time-First (LST) Algorithms, Rate Monotonic Algorithm, Offline Versus Online Scheduling, Scheduling Aperiodic and Sporadic jobs in Priority Driven and Clock Driven Systems.

UNIT 3 Resources Sharing
Effect of Resource Contention and Resource Access Control (RAC), Non-preemptive Critical Sections, Basic Priority-Inheritance and Priority-Ceiling Protocols, Stack Based Priority- Ceiling Protocol, Use of Priority-Ceiling Protocol in Dynamic Priority Systems, Preemption Ceiling Protocol, Access Control in Multiple-Module Resources, Controlling Concurrent Accesses to Data Objects.

UNIT 4 Real Time Communication
Basic Concepts in Real time Communication, Soft and Hard RT Communication systems,

Model of Real Time Communication, Priority-Based Service and Weighted Round-Robin Service Disciplines for Switched Networks, Medium Access Control Protocols for Broadcast Networks, Internet and Resource Reservation Protocols.

UNIT 5 Real Time Operating Systems and Databases

Features of RTOS, Time Services, UNIX as RTOS, POSIX Issues, Characteristic of Temporal data, Temporal Consistency, Con-currency Control, Overview of Commercial Real Time databases.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

3. Feedback of students on attainment of cos.
4. Feedback of students on classroom learning.
5. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Real Time Systems – Jane W. S. Liu, Pearson Education Publication

REFERENCE BOOKS:

1. Real-Time Systems: Scheduling, Analysis, and Verification – Albert M. K. Cheng, Wiley.
2. C.M. Krishna, Kang G. Shin, “Real Time Systems”, Tata McGraw - Hil, 2010.
3. Giorgio C. Buttazzo , “Hard real-time computing systems: predictable scheduling algorithms and applications” , Springer, 2008

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-3)
CO 4____: SOFTWARE VERIFICATION

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	2	3	1	70	30	40	60	200

PRE-REQUISITES: MA10501: Mathematics - II, CO34014: Agile Software Methodology

COURSE OBJECTIVES:
To enable a student to understand the process of software verification while performing software development.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Describe proof methods for software verification,
2. Design human-readable, machine-checkable proofs of software correctness.
3. Identify current trends and methods for software verification.
4. Design and develop reusable formal proofs for concurrent programs.

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction to software verification, Components of software verification, Proof Methods of software verification, Basic of software testing.

UNIT 2 Fundamental Proof Methods: Proving Equalities, Sentential Logic, First-Order Logic, Implication Chaining, Proofs about Fundamental Data Types, Proofs about Algorithms.

UNIT 3 Essential mathematics for software verification, Natural number orderings, Fundamental discrete structures: ordered pairs, options, sets, relations, maps operations and theorems. Tool for software verification: Athena.

UNIT 4 Model checking, abstract interpretation, symbolic testing, program analyses, Modal logics, temporal logic, knowledge logic. Programming language semantics, higher-level abstractions, correctness properties.

UNIT 5 Proof-carrying code, code-carrying proofs, certified code, compiler correctness. Dependent types, session types, information flow types, Proofs at an Abstract Level, Proofs for Concurrent Programs.

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. KonstantineArkoudas and David Musser, Fundamental Proof Methods in Computer Science by, MIT Press, 2017.
2. William Perry, "Effective Methods for Software Testing", John Wiley & Sons, New York, 1995.
3. Gordon Schulmeyer, "Zero Defect Software", McGraw-Hill, New York, 1990.
4. Tom Jenkyns and Ben Stephenson, Fundamentals of Discrete Math for Computer Science: A Problem-Solving Primer, Springer, 2018.
5. Aristides Dasso and Ana Funes, Verification, Validation and Testing in Software Engineering, IGI Global publication, USA, 2006.

REFERENCE BOOKS:

1. Marcus S. Fisher, Software Verification and Validation: An Engineering and Scientific Approach, Springer US, 2007.
2. Janusz Laski and William Stanley, Software Verification and Analysis: An Integrated, Hands-On Approach, Springer; 2009.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-4)
CO 4____: REINFORCEMENT LEARNING

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	3	-	70	30	40	60	200

PRE-REQUISITES: CO 34451: Skill Development Lab, CO34553: Machine Learning

COURSE OBJECTIVES: Explore the foundations of reinforcement learning. Explore how an agent can learn by trial and error. Reinforcement Learning (RL) deals with problems that require sequential decision making. Study different algorithms for RL.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Define the key features of reinforcement learning that distinguishes it from other machine learning techniques.
2. Describe criterias for analyzing reinforcement learning algorithms and evaluate algorithms using performance metrics.
3. Design and validate reinforcement learning models.
4. Implement real world problems using reinforcement learning.

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction to Reinforcement Learning (RL), Elements of RL, RL framework and applications, Introduction to immediate RL, RL tabular solution methods: Bandit Algorithms, Bandit optimalities, Contextual bandits, Value function based methods, Gradient Bandit algorithms, UCB, UCB Theorem, Concentration bounds, Median elimination, Thompson sampling.

UNIT 2 Policies planning, policy evaluation, Policy search, Policy iteration, value iteration, Policy Gradient Methods, Markov Decision Process (MDP), MDP Formulation, MDP modelling, Bellman Equations & Optimality Proofs, Bellman optimality equation, Cauchy sequence & Green's equation, Banach fixed point theorem, Convergence proof, Dynamic Programming, LPI convergence.

UNIT 3 Monte Carlo(MC) methods, Control in MC, Off Policy MC, MC tree search, Temporal Difference Learning, , Q-learning, Eligibility traces, Backward view of eligibility traces, Eligibility trace control, n-step bootstrapping,

UNIT 4 Function approximation, Linear parameterization, State aggregation methods, Function approximation, Afterstate value functions, Batch RL: LSTD & LSTDQ, LSPI & Fitted Q, DQN, Fitted Q & Policy Gradient Approaches, DQN & Fitted Q-iteration, Actor critic & reinforce, Policy gradient with function approximation

UNIT 5	Hierarchical reinforcement learning, Types of optimality, Semi-Markov decision processes, Options, Learning with options, Hierarchical abstract machines, MAXQ, MAXQ value function decomposition, Option discovery, POMDP introduction, Solving POMDP, applications of RL.

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. "Reinforcement Learning: An Introduction" by Andrew Barto and Richard S. Sutton, Second Edition, MIT Press, 2018
2. Marco Wiering and Martijn van Otterlo, “Reinforcement Learning: State-of-the-Art (Adaptation, Learning, and Optimization)” Springer publication, 2012

REFERENCE BOOKS:

1. "Deep Reinforcement Learning Hands-On: Apply Modern RL Methods, with Deep Q-networks, Value Iteration, Policy Gradients, TRPO, AlphaGo Zero and More" by Maxim Lapan, Third Edition, Packt Publishing, 2020.
2. Alexander Zai, Brandon Brown, "Deep Reinforcement Learning in Action", Manning Publications, 1st edition, 2020.
3. Laura Graesser, Wah Loon Keng, "Foundations of Deep Reinforcement Learning: Theory and Practice in Python", Addison-Wesley Publishers, 1st edition, 2019.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-4)
CO 4____: ADVANCED DATABASES

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	3	-	70	30	40	60	200

PRE-REQUISITES: CO34005: Data Base Management Systems, CO____: Advanced Data Structures

COURSE OBJECTIVES: To enable a student in high level understanding of design of advanced databases and their applications with emphasis on Big Data.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Explain design issues of advanced databases and compare different types of data models.
2. Analyze design aspects for databases and query evaluation.
3. Compare and contrast different architectures used in distributed and parallel databases
4. Design and evaluate transactions using concurrency control protocols

COURSE CONTENTS:
THEORY:

UNIT 1 Brief overview of Relational Data Model, Database storage and indexing: Hash based indices; Tree based Indices, Indexes and Performance Tuning. Comparison of File Organizations: cost model, heap files, sorted models, clustered files.

UNIT 2 Introduction to query evaluation: External merge-sort, nested loops join, Sort merge join, Hash joins, Query optimization using selectivity and cost Estimates, Index selection.

UNIT 3 Distributed and Parallel Databases: Design Issues, Distributed query processing, Distributed concurrency control, Reliability, Data integration and Data replication, Distributed recovery, Introduction to Hadoop: Distributed file system, Architecture for Parallel Databases, Parallel Query Evaluation, Parallel Query Optimization.

UNIT 4 Advanced Transaction Management: Granularity of Locks: Granularity of Locks and Degrees of Consistency in a Shared Data Base. Multiversion concurrency control, Optimistic Concurrency control: B-tree locking techniques Isolation Levels, The Aries Recovery Algorithm.

UNIT 5 Emerging Database Technologies: Introduction to XML Databases, No SQL Databases, Spatial and Temporal databases etc.

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Raghu Ramakrishnan and Johannes Gehrke, Database Management Systems, 4/e, 2016,McGraw Hill
2. A SilberschatzHenry.F. Korth&S.Sudarshan, “Database System Concepts” McGraw Hill, 7/e or later.

REFERENCE BOOKS:

1. T. Ozsu and P. Valduriez, Principles of Distributed Database Systems, Springer; 3rd Edition. edition (March 2011), ISBN-10: 1441988335
2. Hector Garcia-Molina , Jeffrey ULLman and Jennifer Widom, “Database Systems:The complete Book”, Prentice-Hall, 3/e, 2017
3. Advanced Databases, Morgan Kauffman (International Handbooks on Information Systems:Book Series)

RESEARCH REFERENCES:

1. ACM Transactions on Database Systems (TODS)
2. IEEE Transactions on Knowledge and Data Engineering (TKDE)
3. T. Neumann, Efficiently Compiling Efficient Query Plans for Modern Hardware, in VLDB, 2011
4. K. Krikellas, et al., Generating Code for Holistic Query Evaluation, in ICDE, 2010
5. H. Pirk, et al., CPU and Cache Efficient Management of Memory-Resident Databases, in ICDE, 2013
6. X. Yu, et al., TicToc: Time-Traveling Optimistic Concurrency Control, in SIGMOD, 2016
7. X. Yu, et al., Staring into the Abyss: An Evaluation of Concurrency Control with One Thousand Cores, in VLDB, 2014

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-4)
CO 44307: CYBER SECURITY AND FORENSICS

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	3	-	70	30	40	60	200

PRE-REQUISITES: CO34007: Computer Networks, CO34554: Foundation of Information Security

COURSE OBJECTIVES:
To understand computer crimes and methods to investigate them.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Define and explain computer crimes.
2. Assess threats and vulnerabilities in operating systems and computer networks and their security requirements
3. Plan and implement preliminary cyber forensic investigations.
4. Design a secure cyber system and comprehensive cyber security policy.

COURSE CONTENTS:
THEORY:

UNIT 1 Review of cryptographic security mechanisms, Definition and types of computer crimes, Distinction between computer crimes and conventional crimes, Types of computer crimes, Need for a Comprehensive Cyber Security Policy, Man-in-the-Middle Attacks

UNIT 2 Review of Network Security: Threats on World Wide Web: phishing, Format string Attack, URL manipulation attack. Cross-site Scripting Attack, SQL Injection Attack etc.; Scrutinizing E-mail, Validating E-mail header information, Tracing Internet access, Browser Helper Objects

UNIT 3 Security in Operating Systems, Memory and Address protection, File system protection, Cases- Windows, Linux O/s. DLL Injection, Virtual Machine Detection

UNIT 4 Introduction to Cyber Forensics, Handling Preliminary Investigations, Controlling an Investigation; Introduction to Database security methods, Attacks against Privileged User Accounts and Escalation of Privileges

UNIT 5 Memory Forensics, Conducting disk-based analysis, Investigating Information-hiding, Tracing memory in real-time, Memory Analysis Frameworks, Dumping Physical Memory, Installing and Using Volatility, Finding Hidden Processes, Volatility Analysis

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. SunitBelapure, Nina Godbole, "Cyber Security", Latest edition, Wiley.
2. Bill Nelson, Amelia Phillips, Christopher Steuart, "Guide to Computer Forensics and Investigations", Latest edition, Cengage Learning.
3. James Graham, Richard Howard, Ryan Olson, "Cyber Security Essentials", CRC Press, Tayor And Francis Group.

REFERENCE BOOKS:

1. Michael Hale Ligh, Steven Adair Blake Hartstein, Matthew Richard, Malware Analyst's Cookbook and DVD, Tools and Techniques for Fighting Malicious Code, Wiley Publishing Inc.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-4)
CO 44308: WEB TECHNOLOGY

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	3	-	70	30	40	60	200

PRE-REQUISITES: NIL

COURSE OBJECTIVES:

1. Understanding the structure of HTTP communication and the format of request- response messages.
2. Design and develop basic web pages using HTML and CSS.
3. Appreciate the functionality of web browsers and web servers.
4. Understand HTML DOM object hierarchy and to learn how to access HTML elements using DOM. The objective of this unit is to understand how to use JavaScript code in an HTML document and to learn about syntax and semantics of JavaScript programming language.
5. The objective of this unit is to understand the importance of XML and to learn the basic building blocks of XML documents.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Create and design web pages using HTML and CSS.
2. Explain working of browser, server engines and XML scripts.
3. Apply the concept of Client-side and server-side programming to create responsive websites.
4. Define working of DOM model to represent web pages.

COURSE CONTENTS:

THEORY:

UNIT 1 HTTP: Headres, working, methods, MIME type and content encoding, Session tracking and Cookies; Web Sockets; HTML–SGML, Basic HTML Elements, Tags and usages, HTML Standards, Issues in HTML, DHTML: Introduction, Cascading Style Sheets.

UNIT 2 Browser: Working of a Browser, Plug-ins; Study of one browser such as Internet Explorer, Mozilla Firefox etc.; Search Engines- Evolution of search engines, Working of Search Engines; Search strategies, indexing.

UNIT 3 Client Side Programming: Java Script, JavaScript Regular expressions, Java script and HTML DOM, Advanced JavaScript and HTML forms, AJAX. Web Servers; Server-Side technologies- plug-in; Server Side Programming: CGI, Servlet, JSP,

UNIT 4 Host Objects: Browsers and the DOM, Introduction to the Document Object Model, Intrinsic Event Handling, Modifying Element Style, The Document Tree, DOM Event

Handling, Accommodating Noncompliant Browsers, Additional Properties of window.

UNIT 5 XML – Basic Standards, Namespaces, DTDS, XML Schema, Linking & Presentation Standards, Parsing XML, XPath, XML Transformation; Web site planning and design; Application lifecycle; Android based web application.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Uttam K. Roy, “Web Technologies”, Oxford University Press, 2012
2. AtulKahate, “Web Technologies”,Tata-McGraw Hill, 2013
3. Moller,”An Introduction to XML and Web Technologies”, Pearson Education, 2012

REFERENCE BOOKS:

1. Jackson, “Web Technologies”, Pearson Education, 2012
2. Deitel, “Internet & World Wide Web”, 5/e, Pearson Education, 2013

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-4)
CO 4____: VIRTUAL REALITY

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	3	-	70	30	40	60	200

PRE-REQUISITES: MA10001: Mathematics – I

COURSE OBJECTIVES:
This course provides students with an opportunity to explore the research issues in Augmented Reality and Virtual Reality (AR & VR). It also makes the students know the basic concept and framework of virtual reality.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Define the fundamentals of virtual reality systems and geometric modeling.
2. Compare and Contrast different types of transformation techniques.
3. Analyserendering problems and their solutions in VR.
4. Design and implement VR experiences.

COURSE CONTENTS:
THEORY:

UNIT 1 **I**ntroduction to VR : Course mechanics, Goals and VR definitions, Historical perspective; Birds-eye view - Hardware, sensors, displays, software, virtual world generator, game engines, human senses, perceptual psychology, psychophysics.

UNIT 2 The Geometry of Virtual Worlds: Geometric modeling, transforming rigid bodies, yaw, pitch, roll, axis-angle representation, quaternions, 3D rotation inverses and conversions, homogeneous transforms, transforms to displays, look-at and eye transforms, canonical view and perspective transforms, viewport transforms.

UNIT 3 Light and Optics, Physiology of Human Vision, Visual Perception**-** Depth perception, motion perception, vection, stroboscopic apparent motion, color perception, combining information from multiple cues and senses, implications of perception on VR; Visual Rendering **-** Graphical rendering, ray tracing, shading, BRDFs, rasterization, barycentric coordinates, VR rendering problems, anti-aliasing, distortion shading, image warping (time warp), panoramic rendering.

UNIT 4 Motion in Real and Virtual Worlds- vestibular system, virtual world physics, simulation, collision detection, avatar motion, vection; Tracking- Tracking systems, estimating rotation, IMU integration, drift errors, tilt and yaw correction, estimating position, camera-feature detection model, perspective n-point problem, sensor fusion, lighthouse approach, attached bodies, eye tracking, inverse kinematics, map building, SLAM; Interaction- Remapping, locomotion, manipulation, social interaction, specialized interaction mechanisms

UNIT 5 **Audio** Sound propagation, ear physiology, auditory perception, auditory localization; Fourier analysis; acoustic modeling, HRTFs, rendering, auralization. Evaluating VR Systems and Experiences:Perceptual training, recommendations for developers, best practices, VR sickness, experimental methods that involve human subjects.

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Steve LaValle'srecorded VR lectures from NPTEL at IIT Madras, July 2015.
2. Doug A. Bowman,ErnstKruijff, Joseph J. LaViola, and Ivan Poupyrev, 3D User Interfaces, AddisonWesley, 2005
3. K.S. Hale and K. M.Stanney, Handbook on Virtual Environments, 2nd edition, CRC Press, 2015.
4. Burdea, G. C. and P. Coffet. Virtual Reality Technology, Second Edition. Wiley-IEEE Press, 2003/2006.

REFERENCE BOOKS:

1. Peter Shirley, MichaelAshikhmin, and Steve Marschner, Fundamentals of Computer Graphics,A K Peters/CRC Press; 3 edition, 2009
2. George Mather, Foundations of Sensation and Perception:Psychology Press; 2 edition, 2009.
3. Alan Craig, William Sherman and Jeffrey Will, Developing Virtual Reality Applications, Foundations of Effective Design, Morgan Kaufmann, 2009.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-4)
CO 4____: ROBOTICS

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	3	-	70	30	40	60	200

PRE-REQUISITES: Basics of Applied Mechanics

COURSE OBJECTIVES:
To provide the basic understanding to the students related to the robots, various types of robots and their working.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Define and identify the basic components of robotic systems.
2. Describe different types of kinematics and dynamics of robots.
3. Design and execute the basic program and functions for robotics.
4. Design and develop the basic robots for specific tasks.

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction - History of robots, Classification of robots, Present status and future trends. Basic components of robotic systems. Basic terminology- Accuracy, Repeatability, Resolution, Degree of freedom. Mechanisms and transmission, End effectors, Grippers-different methods of gripping, Mechanical grippers-Slider crank mechanism, Screw type, Rotary actuators, Cam type gripper, Magnetic grippers, Vacuum grippers, Air operated grippers; Specifications of robot.

UNIT 2 Drive systems and Sensors- Drive system- hydraulic, pneumatic and electric systems Sensors in robot – Touch sensors, Tactile sensor, Proximity and range sensors, Robotic vision sensor, Force sensor, Light sensors, Pressure sensors.

UNIT 3 Kinematics and Dynamics of Robots 2D, 3D Transformation, Scaling, Rotation, Translation, Homogeneous coordinates, multiple transformations, Simple problems. Matrix representation, Forward and Reverse Kinematics Of Three Degree of Freedom, Homogeneous Transformations, Inverse kinematics of Robot, Robot Arm dynamics, D-H representation of robots, Basics of Trajectory Planning.

UNIT 4 Robot Control, Programming and Applications Robot controls-Point to point control, Continuous path control, Intelligent robot, Control system for robot joint, Control actions, Feedback devices, Encoder, Resolver, LVDT, Motion Interpolations, Adaptive control.

UNIT 5 Introduction to Robotic Programming, On-line and off-line programming, programming examples. Robot applications and case studies.

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Mikell P Groover, Nicholas G Odrey, Mitchel Weiss, Roger N Nagel, Ashish Dutta, “Industrial Robotics, Technology programming and Applications", McGraw Hill, 2012.
2. Craig. J. J. “Introduction to Robotics- mechanics and control”, Addison- Wesley, 1999.

REFERENCE BOOKS:

1. Richard D. Klafter, Thomas .A, ChriElewski, Michael Negin, "Robotics Engineering an Integrated Approach", PHI Learning., 2009.
2. Francis N. Nagy, AndrasSiegler, "Engineering foundation of Robotics", Prentice Hall Inc., 1987.
3. Carl D. Crane and Joseph Duffy, "Kinematic Analysis of Robot Manipulators", Cambridge University press, 2008.
4. Fu. K. S., Gonzalez. R. C. & Lee C.S.G., “Robotics control, sensing, vision and intelligence”, McGraw Hill Book co, 1987.

B.Tech. IV YEAR (4YDC)

Semester -A (ELECTIVE-IV)

CO__________: NATURAL LANGUAGE PROCESSING

(Passed in B.o.S on 06/06/23 to be added in list of Electives w.e.f 2020 Enterants)

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	3	-	70	30	40	60	200

Course Objective: To enable a student in understanding and applying basics of NLP, covering standard frameworks for dealing with natural language as well as algorithms and techniques to solve various NLP problems in field of machine learning.

Pre Requisites:

1. Knowledge of probability, linear algebra, multivariate calculus.
2. Proficiency in Python: Python, Numpy and PyTorch libraries.

Course Outcomes: After Completing this course, student will be able to:

CO1: Identify the basic models of text processing and will be able to perform PoS tagging

CO2: Select and implement sequential tagging techniques that are suitable for the applications under consideration.

CO3 :Solve problems associated withDistributional Semantics, Lexical Semantics

CO4: Apply and implement text mining algorithms and comprehend sentiment Analysis.

Syllabus

UNIT1. Introduction and Basic Text Processing, Spelling Correction, Language Modeling, Advanced smoothing for language modeling, POS tagging

UNIT2. Models for Sequential tagging – MaxEnt, CRF , Syntax – Constituency Parsing , Dependency Parsing

UNIT3. Distributional Semantics, Lexical Semantics, Topic Models

UNIT4. Entity Linking, Information Extraction , Text Summarization, Text Classification

UNIT5. Sentiment Analysis and Opinion Mining

Text Books:

1. Dan Jurafsky and James Martin. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition. Prentice Hall, Second Edition, 2009. Some draft chapters of the third edition are available online: https://web.stanford.edu/~jurafsky/slp3/

2. Chris Manning and HinrichSchütze. Foundations of Statistical Natural Language Processing. MIT Press.

Reference Books:

1. NitinIndurkhya, Fred J. Damerau “Handbook of Natural Language Processing”, Second Edition, CRC Press, 2010.
2. James Allen “Natural Language Understanding”, Pearson Publication 8th Edition. 2012.

3. Hobson lane, Cole Howard, Hannes Hapke, “Natural language processing in action” MANNING Publications, 2019.
4. Alexander Clark, Chris Fox, Shalom Lappin, “The Handbook of Computational Linguistics and Natural Language Processing”, Wiley-Blackwell, 2012

Web Resources:
1. https://github.com/keon/awesome-nlp
2. https://github.com/khanhnamle1994/natural-language-processing
3. https://github.com/joosthub/PyTorchNLPBook

NPTEL Courses:
1.https://archive.nptel.ac.in/courses/106/105/106105158/

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A (ELECTIVE-4)
CO 4____: SOFTWARE TESTING

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
3	-	-	3	-	70	30	40	60	200

Pre-requisites: Programming, Algorithms, Discrete Mathematics (basics).

Course Objective:

1. To introduce fundamental concepts of Software Testing.
2. To enable understanding of various effective testing techniques for ensuring high quality software.

Course Outcomes:

1. Analyze and understand the use of software testing methods and modern software testing tools.
2. Design a software test process for a software testing project.
3. Make use of software testing methods and modern software testing tools for the testing projects.
4. Apply testing techniques to test object oriented applications, web applications and other domain projects.

Course Contents:

Unit-1: Introduction to Software Testing, Basic Terminologies, Testing based on Models and Criteria, Tools for Automated Testing; Techniques and algorithms for test case design, Graph Based Testing: Basics, Structural Graph Coverage Criteria & Algorithms, Elementary Graph Algorithms, Data Flow Graph Algorithms, Testing Source Code.

Unit-2: Graphs coverage for source code, design elements and requirements: Software Design and Integration Testing, Design Integration Testing, Specification Testing, Finite state Machines, Basics Needed for Software Testing.

Unit-3: Logic Based Testing: Coverage Criteria, Logic Coverage Criteria for Test Code and its issues, Specification based logic coverage, logic coverage on finite state machines.

Unit-4: Functional Testing, Input Space Partitioning: Coverage Criteria and Examples, Syntax-Based Testing; Mutation Testing: Introduction, Mutation Testing vs. Graphs and Logic Based Testing, Mutation for integration.

Unit-5: Testing of Web Applications and Web Services, Testing of Object-Oriented Applications, Testing of Mobile Applications, Symbolic Testing, DART: Directed Automated Random Testing, Non-Functional System Testing, Regression Testing.

Text Books:

1. Mauro Pezze and Michal Young, Software Testing and Analysis: Process, Principles, and Techniques, Willey.
2. Srinivasan Desikan and Gopalaswamy Ramesh, Software Testing: Principles and Practices, Pearson Education.
3. Desai Sandeep and Srivastava Abhishek, Software Testing: A Practical Approach, PHI.

1.

e-Resources:

1. Software Testing Course on NPTEL, https://onlinecourses.nptel.ac.in/noc19_cs71/preview.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-A
CO 44401: SYSTEM OPERATIONS LAB

*Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
-	-	2	-	1	-	-	40	60	100

PRE-REQUISITES: CO24497: Programming Practices, CO24508: Operating System

COURSE OBJECTIVES:
To enable students for creation, design, development, installation, deployment and configuration of computer infrastructures in an organization.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Configure and install Linux and Windows operating systems and softwares.
2. Install and configure different services like apache, file server, apt cacher etc.
3. Configure different security settings and backup plans as per requirements.
4. Design and deploy state-of-art infrastructure setup for real world applications.

COURSE CONTENTS:
THEORY:

LAB 1 Partition management with linux and Installation of Linux, Post Install, Configuring run levels and consoles.

LAB 2 Fstab and Crontab, file system, file and directory, Commands permissions, users, groups, devices, Linux services and daemons, Searching with locate, updatedb.

LAB 3 Concept of modules, Linux Backup and Recovery Services, kernel compilation, Compilation-options, creating initial RAM disk image.

LAB 4 Extra installations using RPMs and Make utility, Creating a Local Area Network using Red Hat Linux, Assigning static IP addresses, Connecting to Internet.

LAB 5 Network services, Network Security, Creating Linux Network-Dynamic IPaddresses, DHCP server and Client Configuration, Host names, Name lookup on LAN.

LAB 6 Filters and scripting for management of server using sed, grep, awk.

LAB 7 NFS-Native file sharing service, Configuring NFS, Samba server, Creating Samba share and Samba users, Configuring Samba windows client Configuring FTP server and Client

LAB 8 Telnet and SSH, Running Telnet and SSH sessions, Configuring CUPS, NIS, DNS, Apache, SQUID, IPTABLES.

LAB 9 Microsoft Windows Advanced server installation, network and service configuration, troubleshooting, user and security management.

LAB 10 Remote management and administration. Backup and Recovery Services.

LAB 11 Database Administration: creation of database and its user, granting rights to users on tables.

LAB 12 Database storage methods, backup and recovery.

DIRECT ASSESMENT:

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. "Linux Administration: A Beginner's Guide", Wale Soyinka, McGraw-Hill Publishers, 8th Edition, April 2020.
2. "Mastering Linux Administration" by AlexandruCalcatinge, Julian Balog, Packt Publishing, 1st edition, June 2021
3. "Linux in Action" David Clinton, Manning Publications, August 2018.

REFERENCE BOOKS:

1. "The Linux Command Line: A Complete Introduction" William E. Shotts Jr., 1st Edition, No Starch Press, 2012.
2. "Linux Bible" by Christopher Negus, Wiley Publication, 10th Edition, 2020.
3. "Pro Linux System Administration: Learn to Build Systems for Your Business Using Free and Open Source Software" by Dennis Matotek, James Turnbull, Peter Lieverdink, Apress Publisher, Second Edition, March 2017.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-B (ELECTIVE-5)
CO 4____: BIOINFORMATICS COMPUTING

*Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
2	-	-	2	-	70	30	-	-	100

PRE-REQUISITES: MA10501: Mathematics-II, CO34451: Skill Development Lab

COURSE OBJECTIVES: To provide knowledge and hands on training on various computational tools and techniques employed in bioinformatics computing.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Describe fundamental principles and theories of bioinformatics computing.
2. Compare and analyze the different bioinformatics methodology.
3. Analyze problems in genomics.
4. Solve the real world problems using bioinformatics.

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction to bioinformatics, Proteomics, Strategies for Protein Separation, Secondary structure and Tertiary structure, Strategies for Protein Identification, Quantitation, Structural Proteomics, Protein Chips, Methods of Protein Engineering.

UNIT 2 Introduction to Molecular biology, Molecular Dynamics, Monte Carlo and Molecular Dynamics in Various Ensembles. System biology, biological sequences, patterns in biological sequences, genetic, genetic alterations and genomics, Engineering of Macromolecules.

UNIT 3 DNA, RNA, Application of Recombinant DNA Technology. DNA re-association kinetics, repetitive and unique sequences, kinetics and sequence complexities, DNA polymorphism, nucleotides, DNA sequences, DNA engineering.

UNIT 4 Biological database, DNA and protein database, DNA Data Bank of Japan (DDBJ), DHCP database.

UNIT 5 Applications of Bioinformatics in molecular medicine, personalized medicine, preventative medicine, gene therapy, agriculture, animal, waste cleanup etc. case studies.

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Bryan Bergeron M.D., Bioinformatics Computing, Pearson publication, 2002.
2. Hancock J M, Bioinformatics and Computational Biology, Second Edition, Wiley publication, 2014.
3. Vince Buffalo, Bioinformatics Data Skills, O'Reilly publication, 2015.

REFERENCE BOOKS:

1. Pazos, Florencio, Practical Protein Bioinformatics, Springer International Publishing, 2015.
2. David Mount, Bioinformatics: Sequence and Genome Analysis, Cold Spring Harbor Laboratory Press, Second Edition, 2004

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-B (ELECTIVE-5)
CO 4____: HIGH PERFORMANCE COMPUTING

*Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
2	-	-	2	-	70	30	-	-	100

PRE-REQUISITES: CO24009: Computer Architecture**,** CO24508: Operating Systems

COURSE OBJECTIVES: Our objective is to develop such high-performance, practical solutions, and contribute to the needs of both scientific and engineering community

COURSE OUTCOMES:
After completing the course student should be able to:

COURSE CONTENTS:
THEORY:

UNIT 1 Introduction to modern processors-: General Purpose cache based architecture-performance metric and benchmarks, Moore's Law, pipelining, super clarity, SIMD. Memory Hierarchies, Multicore processors, Multi threaded processors, .Basic Optimizations for serial codes:- Scalar profiling, common sense optimizations, Simple measures and their impacts, role of compilers, C++ optimizations.MIPS Assembly Language Program, Memory layout, SPIM Simulator.

UNIT 2 Superscalar architecture: An overview, Instruction flow optimization: Handling branches, Branch predictors, Advanced optimization in instruction flow, register flow techniques: Register renaming and out of order execution, Out of order execution, Advanced data flow techniques: Instruction reuse and value prediction, Memory data flow, Advanced memory data flow architectures, Limits of superscalar architectures

UNIT 3 Beyond ILP, Multi-threading, Simultaneous multithreaded (SMT) architectures, SMT architecture: Choices, SMT performance on various designs, SMT architecture: OS impact and adaptive architectures, OpenMIP, OpenMP. CUDA programming.

UNIT 4 VLIW architectures, Multiscalar architecture, Multi-core Architectures, Multicore Interconnect – NOC Network-on-Chip,Tiled Chip Multicore Processors(TCMP), Routing Techniques in Network on Chip(NoC), NoC Router Microarchitecture, TCMP and NoC: Design and Analysis,

UNIT 5 Super-scalar and Memory Hierarchy Introduction.Basic and Advanced Optimization Techniques in Cache Memory, Cache Optimization using gem5, Cache Coherence, Cache Consistency model**.**Case Study: Locking hardware and algorithmTask Scheduling on Multicore System GP-GPU Architecture, CPU-GPU Integration

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. George Hager and Gerhard Wellein , “ Introduction to high performance Computing for scientists and engineers”, CRC Press
2. Charles Severance, Kevin Dowd, “High Performance Computing”, 2nd Edition, O'Reilly.
3. Patterson, D.A., and Hennessy, J.L. , “Computer Organization and Design: The Hardware/Software Interface”, Morgan Kaufmann Publishers, 4th Edition, Inc.2005
4. Patterson, D.A., and Hennessy, J.L. , “Computer Architecture : A Quantitative Approach ”, Morgan Kaufmann Publishers, 4th Edition, Inc.2005
5. Hamacher, V.C., Vranesic, Z.G., and Zaky, S.G., “Computer Organization”, 5/e. McGraw-Hill. 2008
6. Tanenbaum, A.S, “Structured Computer Organization”, Prentice-Hall of India. 1994

REFERENCE BOOKS:

1. Stalling W, “Computer Organization and Architecture ”, Pearson Education India. 2008
2. D V Hall, Microprocessors and Interfacing, TMH, 1995
3. Brey. Barry B, The Intel Microprocessors 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, and Pentium Pro Processor Architecture, Programming, and Interfacing, Prentice Hall India, 2005

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-B (ELECTIVE-5)
CO 4____: MACHINE LEARNING FOR SECURITY

*Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
2	-	-	2	-	70	30	-	-	100

PRE-REQUISITES: CO34554: Foundation of Information Security, CO34553: Machine Learning

COURSE OBJECTIVES:

1. To enable students to understand different machine learning techniques and their application in domain of security.
2. To understand data collection from a computer network and system for application of machine learning

COURSE OUTCOMES:

After completing the course student should be able to:

1. Describe supervised and un-supervised machine learning models.
2. Apply machine learning techniques to different real-world information security problems.
3. Analyze data from various system and networks in order to perform analysis and make predictions.
4. Analyze the performance parameters of machine learning models in information security domain.

COURSE CONTENTS:

THEORY:

UNIT 1 **Classification and Clustering-**Machine Learning: Problems and Approaches, Model Families, Loss Functions, Optimization; Supervised Classification Algorithms-Logistic Regression. Decision Trees, Decision Forests, Support Vector Machines, Naive Bayes, k-Nearest neighbors, Neural Networks; Practical Considerations in Classification: Selecting a Model Family, Training data Construction, Feature Selection, Overfitting and Underfitting, Choosing Thresholds and Comparing Models

Clustering: Clustering Algorithms, Evaluating Clustering Results

UNIT 2 **Anomaly Detection-** Anomaly Detection Versus Supervised Learning, Intrusion Detection with Heuristics, Data-Driven Methods, Feature Engineering for Anomaly Detection: Host Intrusion Detection, Network Intrusion Detection, Web Application Intrusion Detection, Anomaly Detection with Data and Algorithms: Forecasting Supervised machine Learning, Statistical Metrics, Goodness-of-Fit, Unsupervised Machine Learning Algorithms, Density-Based Methods, Challenges of Using Machine Learning in Anomaly Detection, Response and Mitigation, Practical System Design Concerns, Optimizing for Explainability, Maintainability of Anomaly Detection Systems, Integrating Human Feedback,Mitigating Adversarial Effects

UNIT 3 **Malware Analysis-** Understanding Malware: Defining Malware Classification, Feature generation: Data Collection, Generating Features, Feature Selection, From Features to Classification: How to Get Malware Samples and Labels

UNIT 4 **Network Traffic Analysis:**Access Control and Authentication, Intrusion Detection, Detecting In-Network Attackers, Data-Centric Security, HoneypotsBuilding a Predictive Model to Classify Network Attacks: Exploring the Data, Data Preparation, Classification, ,Supervised Learning, Semi-Supervised , Learning, Unsupervised Learning, Advanced Ensembling

UNIT 5 **Production Systems**- Machine Learning System Maturity and Scalability, Data Quality: Bias in Datasets, Missing Data, Data quality; Model Quality: Hyperparameter Optimization, Feature: Feedback Loops, A/B Testing of Models, Repeatable and Explainable Results; Performance: Goal: Low Latency, High Scalability, Performance Optimization; Maintainability: Problem: Check pointing, Versioning, and Deploying Models, Graceful Degradation, Tuning and Configurable, Monitoring and Alerting; Security and Reliability: Robustness in Adversarial Contexts, Data Privacy Safeguards and Guarantees, Feedback and Usability

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Clarence Chio and David Freeman, Machine Learning & Security: Protecting Systems With Data And Algorithms Protecting Systems, Latest Edition, O'Reilly.
2. Mark Stamp, Introduction To Machine Learning With Applications In Information Security, CRC Press, Taylor & Francis Group.

REFERENCE BOOKS:

1. Marcus A. Maloof, Machine Learning and Data Mining for Computer Security Methods and Applications (Advanced Information and Knowledge Processing).
2. Tony Thomas, Athira P. Vijayaraghavan, Sabu Emmanuel, Machine Learning Approaches in Cyber Security Analytics, Springer.
3. Gupta, Brij Sheng, Quan Z, Machine learning for computer and cyber security principles, algorithms, and practices-CRC Press (2019).

DEPARTMENT OF COMPUTER ENGINEERING
B. E. IV YEAR (4YDC)
SEMESTER-B (ELECTIVE-5)
CO44608: GAME DESIGN

*Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
2	-	-	2	-	70	30	-	-	100

PRE-REQUISITES: Any Programming Language (C/C++/Java/Python)

COURSE OBJECTIVES:

1. To understand the basic concepts of game design, such as rules and play.
2. To analyse various types of game design rules, concepts and technique.
3. To develop game design solutions having theoretical, narrative, perceptual and aesthetic coherence.
4. To implement several real world games to gain experience of creating playable games.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Understand fundamental concepts relating to games and design, such as rules and play.
2. Compare and contrast different types of game design rules, concepts and technique.
3. Develop game design solutions having theoretical, narrative, perceptual and aesthetic coherence.
4. Implement several real world games to gain experience of creating playable games.

COURSE CONTENTS:

THEORY:

UNIT 1 Introduction Game Terminologies: Language,Objects and verbs,Scenes,Context, Dialogue, Resistance, Storytelling, Classifications of games: genres of games,Commercial and psychological impact of games; Crowdsourcing knowledge game in science,Games for healthy minds, Fair play.

UNIT 2 Game Design Interfaces: Region based interface,Offline vs Online Interface,Hardware vs software interface,Game Design Controls ,Defining behavior interface of objects, like car,gun,soldiers etc.

UNIT 3 Graphics and effects:Sprites:Layering with depth, sheets and the GPU,scaling , Animation,Tiling,Illusion of Depth, Particle System: Particle,Creatingeffects,BlendingTypes,Types of effects and Effect System,GPU Programming.

UNIT 4 Tools in game:Unity,Stencyl,GameMaker,Blender,Photoshop,Blenderetc.Game design framework:MDA, Data Structures for game design:STL etc. Game Design Protocols and Communication Overheads.

UNIT 5 Case study of game:on quality and performance parameters,user engagement and interactivity ,commercial and psychological impact.

DIRECT ASSESMENT:

ASSESSMENT OF THEORY:

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc.(30%)
2. End semester Theory Exam (70%).

ASSESSMENT OF PRACTICAL:

1. Internal Assessment for continuous evaluation (40%): Lab assignments, demonstration, Viva, File etc.
2. End semester Practical Exam (60%): Quiz/Programming test, lab journal, demo, viva etc.

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Making Games for Impact KURT SQUIRE
2. Nystrom Robert, Game Programming Patterns, 3rd edition, Genever Benning, 2014.
3. The evolution of social impact of video game economics.
4. 2D Graphic Programming for games. Game Graphics programming

REFERENCE BOOKS:

1. Anna Anthropy and Naomi Clark, A Game Design Vocabulary: Exploring the Foundational Principles Behind Good Game Design, 2014, Addison-Wesley Professional.
2. Sung, K., Pavleas, J., Arnez, F., Pace, J., Build Your Own 2D Game Engine and Create Great Web Games: Using HTML5, JavaScript, and WebGL, 2015, Apress Publishers.
3. Zach Hiwiller, Players Making Decisions, 2nd Edition, 2019, New Riders Publishers.
4. Patrick Alessi, Beginning IOS Game Development, 2011, Wrox Publishers.
5. The business of Gamification A critical analysis Edited by MikolajDynmek and Peter Zackarisson.
6. Game psychology and behavior(Springer)
7. DATA STRUCTURES AND ALGORITHMS FOR GAME DEVELOPERS by Allen Sherrod.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-B (ELECTIVE-5)
EC 4____: DIGITAL SIGNAL PROCESSING

*Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
2	-	-	2	-	70	30	-	-	100

PRE-REQUISITES: MA10001: Mathematics - I, EC24509: Digital Communication

COURSE OUTCOMES:

After completing the course student should be able to:

1. Define time and frequency domain representation of discrete time signals and systems.
2. Find the response of a discrete time system for arbitrary inputs to design DTS.
3. Evaluate numerically the response of DTS for finite time inputs.
4. Design and realize digital IIR, FIR filters.

COURSE CONTENTS:

THEORY:

UNIT 1 Discrete time signals & systems : Introduction, types of signals, discrete time signal sequences, discrete time systems, linear shift invariant systems, Stability & causality, linear constant coefficient difference equation, frequency domain representation of discrete time systems & signals, properties of the Discrete Time Fourier transform (DTFT), Sampling and discrete time processing of continuous-time signals.

UNIT 2 Z-Transform and Transform analysis of LTI systems: Z-transform, Inverse Z transform, properties of Z-transform, one sided Z-transform and its applications, system function, frequency response of LTI systems, minimum phase and linear phase systems.

UNIT 3 Discrete Fourier transform (DFT),and its computation: Discrete Fourier Series, Discrete Fourier Transform, Linear convolution using Discrete Fourier Transform, Computation of DFT, Decimation in time FFT algorithms, Decimation in frequency algorithms, FFT algorithms for N (a composite number), chirp Z-transform algorithm.

UNIT 4 Implementation of digital filters: Signal flow graph representation, Realization of IIR & FIR systems, direct form, Transposed form, Parallel form, Cascade form, Lattice structure for IIR and FIR filters, Parameter quantization effects

UNIT 5	Digital filter design techniques: Design of IIR digital filters using Impulse-invariant and bilinear transformation methods, Design of FIR filter using Windowing methods, Design examples.

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Oppenheim & Schafer, Discrete Time Signal Processing, 3rd ed., 2010, Pearson Education.
2. Proakis and Manolikis, Digital Signal Processing, 4th ed., 2005, Pearson Education.
3. MitraSanjit, Digital Signal Processing A Computer Based Approach, 3rded., 2005, TMH.

REFERENCE BOOKSRECOMMENDED:

1. Schaum's Outline Series, Digital Signal Processing,2007, TMH.
2. Ludeman L.C., Fundamentals of DSP, 1986, John Wiley.
3. Salivahanan, Vallavaraj, Digital signal processing, 2nd ed.,2011, TMH.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-B (ELECTIVE-5)
CO 4____: SECURITY IN RESOURCE CONSTRAINED ENVIRONMENT

*Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
2	-	-	2	-	70	30	-	-	100

PRE-REQUISITES:

1. CO34881: Internet of Things
2. CO_______: Embedded Systems
3. CO34554: Foundation of Information Security

COURSE OBJECTIVES:

To provide the students a basic understanding on IoT security, various types of security attacks in IoT and case studies.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Explain IoT general models and security challenges.
2. Recognize IoT security and vulnerability threats.
3. Demonstrate different IoT protocols and their security measures.
4. Identify different types of attacks in the IoT domain and propose solutions to provide security.

COURSE CONTENTS:

THEORY:

UNIT 1 Introduction – IOT Functional View, IoT Security Challenges, Hardware Security Risks, Resource Constrained Computations, Devices Physical Security, Software Security Risks; Software Vulnerabilities, Data Interception, Lack of Industrial Standards.

UNIT 2 Security Requirements: Data Confidentiality, Data Encryption, Data Authentication, Secured Access Control, Secret Key, Authentication/Authorization for Smart Devices, Constrained System Resources, Device Heterogeneity, Fixed Firmware, IoT Attacks, and Side channel Attacks, Spoofing, and Sniffing etc

UNIT 3 Security Protocols: Infrastructure: IPv6, 6 LowPAN , Identification: Electronic Product Code, uCode, Transport: Bluetooth, BLE, Zigbee, Data: MQTT ,CoAP, Multilayer Frameworks: Alljoyn, IoTivity.

UNIT 4 Test Device Range, Latency and Capacity, Manufacturability Test, Secure from Physical Attacks, Trusted IoT Application Platforms, Secure Firmware Updating, Network Enforced Policy, Secure Analytics Visibility and Control.

UNIT 5 Case Studies: MIRAI Botnet Attack, Iran’s Nuclear Facility Stuxnet Attack, Tesla Crypto jacking attack, The RENDnet Webcam Attack etc.

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. “Security and Privacy in Internet of Things (IOT's): Models, Algorithms and Implementations” by Fei Hu., CRC Press.
2. “Security in Resource Constrained Devices” by Angelo Spognardi.

REFERENCE BOOKS:

1. “Security Management of Controlled Networks in a Resource-Constrained Environment” by Michael Joseph Chapple, Proquest, Umi Dissertation Publishing.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. II YEAR (4YDC)
Elective-5
SEMESTER-B
CO 44_______: Advanced Graph Theory

Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
2	-	-	2	-	70	30	-	-	100

PRE-REQUISITES: MA10001: Mathematics - I, MA10501: Mathematics - II

COURSE OBJECTIVES: To enable a student to learn the basic concepts of graph theory, its implementations and its applications in computer science.

COURSE OUTCOMES:
After completing the course student should be able to:

5. Solve problems which involve graph theory.
6. Explain the notion of mathematical thinking, mathematical proofs, logical thinking, and combinatorics and be able to apply them in problem solving.
7. Describe the basic terminology and properties of graphs and trees and apply them to solve practical problems.
8. Apply algebraic techniques effectively to analyse basic computation of graph theory and algorithms for real world applications.

UNIT 1 Graphs and Trees: Basic terminologies, Types, Properties, Shortest path Algorithms; Cutsets; Hamiltonian and Eulerian paths and circuits; Tree Traversals; Spanning Trees; Applications in computer science. Connection, Bipartite Graphs, Eulerian Circuits, Vertex Degrees and Counting, Degree-sum formula, The Chinese Postman Problem and Graphic Sequences.
Trees and Distance, Properties of Trees, Spanning Trees and Enumeration, Matrix-tree computation, Cayley's Formula, Prufer code.

Unit 2 Matchings and Covers, Hall's Condition, Min-Max Theorem, Independent Sets, Covers and Maximum Bipartite Matching, Augmenting Path Algorithm, Weighted Bipartite Matching, Hungarian Algorithm.
Stable Matchings and Faster Bipartite Matching, Factors & Perfect Matching in General Graphs, Matching in General Graphs: Edmonds' Blossom Algorithm

Unit 3 Connectivity and Paths: Cuts and Connectivity, k-Connected Graphs, Network Flow Ford-Fulkerson Labeling Algorithm, Max-Flow Min-cut Theorem, Menger's Proof using Max-Flow Min-Cut Theorem.
Vertex Coloring and Upper Bounds, Brooks' Theorem and Color-Critical Graphs, Counting Proper Colorings.

Unit 4 Planar Graphs, Characterization of Planar Graphs, Kuratowski's Theorem, Wagner's Theorem.

Unit 5 Line Graphs and Edge-coloring, Hamiltonian Graph, Traveling Salesman Problem and NP-Completeness, Dominating Sets.

Books and references

1. D.B. West, Introduction to Graph Theory, Prentice Hall, 2001
2. Jon Kleinberg and Eva Tardos, Algorithm Design, Addison-Wesley, 2005
3. J.A.Bondy and U.S.R.Murty: Graph Theory, Springer, 2008.
4. R.Diestel: Graph Theory, Springer(low price edition) 2000.
5. F.Harary: Graph Theory, Narosa, (1988)
6. C. Berge: Graphs and Hypergraphs, North Holland/Elsevier, (1973)

COMPUTER ENGINEERING DEPARTMENT
B.Tech. IV YEAR(4YDC)
Elective-5 (SEM B)

CO4_____: INTRODUCTION TO SOFT COMPUTING

PERIOD PER WEEK			CREDITS			MAXIMUM MARKS				
T	P	Tu	T	P	Tu	THEORY		PRACTICAL		TOTAL MARKS
						CW	Th.	SW	Prac.	
2	-	-	2	-	-	30	70	-	-	100

PREREQUISITES: CO34451: Skill Development Lab

COURSE OBJECTIVES:
1. To introduce basic concepts, theories and techniques of Soft Computing.
2. Help students to learn the applications of Soft Computing algorithms in real world use-cases.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Describe in depth about Fuzzy techniques and defuzzyfication techniques .
2. Analyze the nature of the data given and employ the most appropriate Soft Computing algorithm.
3. Compare and Contrast Different Soft Computing algorithms and their pragmatic usages.
4. Describe the working and applications of Artificial neural Network.

UNIT-1 Introduction to Soft Computing, Introduction to Fuzzy logic, Fuzzy membership functions,Operations on Fuzzy sets, Fuzzy relations, Fuzzy propositions, Fuzzy implications, Fuzzy inferences.

UNIT-2 Defuzzyfication Techniques-AI (adaptive integration), BADD (basic defuzzification distributions), BOA (bisector of area), CDD (constraint decision defuzzification), COA (center of area), COG (center of gravity), ECOA (extended center of area), Fuzzy logic controller, Expert Systems.

UNIT-3 Solving optimization problems, Concept of GA, GA Operators: Encoding,GA Operators: Selection, GA Operators: Crossover, GA Operators: Mutation, Problem Solving; Introduction to Genetic Programming, Evolutionary Programming, and Evolutionary Strategies.

UNIT-4 Introduction to EC , MOEA Approaches: Non-Pareto, MOEA Approaches: Pareto

UNIT-5 Motivation behind ANN - Biological neuron, Introduction to ANN, ANN Architecture, Activation Functions, loss functions, gradient descent algorithm, ANN Training, back propagation algorithm, Applications of ANN.

DIRECT ASSESMENT:

ASSESMENT OF THEORY

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. An Introduction to Genetic Algorithm Melanic Mitchell (MIT Press)
2. Evolutionary Algorithm for Solving Multi-objective, Optimization Problems (2nd Edition), Collelo, Lament, Veldhnizer (Springer)
3. Fuzzy Logic with Engineering Applications Timothy J. Ross (Wiley)

REFERENCE BOOKS:

1. Neural Networks and Learning Machines Simon Haykin (PHI)

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-B (ELECTIVE-6)
CO 4____: ADVANCED OPERATING SYSTEMS

*Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
2	-	-	2	-	70	30	-	-	100

PRE-REQUISITES: CO24497: Programming Practices, CO24508: Operating Systems

COURSE OBJECTIVES: To enable a student in high level understanding of advanced operating systems and their applications in diverse domains.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Demonstrate understanding of design issues of advanced operating systems and
2. Compare different types of operating systems.
3. Analyze design aspects for different subsystems of diverse Operating Systems.
4. Compare & Contrast different architectures used in Distributed Operating Systems
5. Analyze synchronization amongst various components of a typical operating system.

COURSE CONTENTS:
THEORY:

UNIT 1 Review of Operating System Fundamentals; Different Types, Dedicated Operating Systems; The Genesis of Modern Operating Systems; Operating Systems Design Strategies/Kernel Architectures – Microkernels, Exokernels etc.
Operating Systems Services, System Calls and their Implementation.

UNIT 2 File Systems and Main Memory Management: File Concept, Different Modules of a File System; File Protection; Disk Partitioning; Kernel I/O Subsystem; Advancements for improving File System Performance; System Calls for File Systems Management.
Review of Main Memory Management Techniques including Virtual Memory.

UNIT 3 Process Management : Review of Process and Scheduling Concepts; System Calls for Process Management; IPC; Concept of Threads: Process v/s Threads, User Level & Kernel Level Threads, Threads Scheduling, Threading Issues, Solutions to Critical Section Problem and Synchronization for Threads;
Scheduling in Multi-core Systems. Load Balancing Techniques for Multicore and Multiprocessor systems- Dynamic Load Balancing, Process Migration.

UNIT 4 Distributed Operating Systems: Design Issues; Overview of Distributed File Systems, Distributed Process Management and Distributed Memory Management.
Embedded and IoT Operating Systems: Introduction, Characteristics and Features, Challenges and Issues in Designing the Operating Systems for Resource Constrained Systems.

UNIT 5 Virtualization: Basic Concepts, Benefits and Features, Building Blocks, Virtualization and O/S Components, Hypervisors; Virtual Machines, CPU and Memory Virtualization.
Case Studies: Unix/Linux, Windows and Contemporary Embedded Operating Systems like Embedded Linux; Study of Source Code of Open Source Operating System like Linux.

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Silberschatz, Galvin, Gagne, “Operating System Concepts’’, Wiley, 9/E
2. William Stallings, “Operating Systems: Internals and Design Principles”, Pearson
3. Pradeep K. Sinha, “Distributed Operating System: Concept and Design’’, PHI
4. Matthew Portnoy, “Virtualization Essentials”, Sybex

REFERENCE BOOKS:

1. Andrew S. Tanenbaum, Albert S. Woodhull, “Operating Systems: Design and
2. Implementation”, Pearson
3. Wang K.C., “Embedded and Real-Time Operating Systems”, Springer
4. 3 Bovet &Cesati, “Understanding the Linux Kernel”, O’Reily
5. Maurice J. Bach, “The Design of Unix Operating System”, Prentice Hall

REFERENCE LINKS:-

1. http://www.kernel.org/
2. http://www.linux.org/
3. http://www.linuxquestions.org/

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-B (ELECTIVE-6)
CO 4____: NETWORK MANAGEMENT AND MAINTENANCE

*Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
2	-	-	2	-	70	30	-	-	100

PRE-REQUISITES: CO34007: Computer Network

COURSE OBJECTIVES:

COURSE OUTCOMES:
After completing the course student should be able to:

1. Explain Directory Services, Network protocols and services and Remote Access.
2. Demonstrate the set-up and use of Virtual Private Network.
3. Configure various services on Windows server platform.
4. Identify issues in a network and provide solutions to troubleshoot those issues.

COURSE CONTENTS:
THEORY:

UNIT 1 Directory Services: Define Directory Service, Definition of Novelle Directory, Windows Domain, MS Active Directory, X500 Directory Access Protocol, Lightweight Directory Access Protocol, Forests, Trees, Roots and Leaves. Active Directory Architecture: Object Types, Object Naming, Canonical Names, LDAP Notation, Globally unique identifiers, User Principle Names, Domain, Trees & Forests. Virtual Private Network: VPN Protocols, Types of VPN, VPN Clients, SSL VPNs

UNIT 2 Dynamic Host Control Protocol(DHCP): DHCP Origins, Reverse Address Resolution Protocol (RARP), The Bootstrap Protocol (BOOTP), DHCP Objectives, IP Address assignments, DHCP Architecture. Introduction to Domain Name Systems (DNS): DNS Objectives, Domain Naming, Top Lavel Domains, Second Level Domains, Sub-domains, DNS Functions, Resource Records, DNS Name Resolution, Resolves, DNS Requests, Root Name Servers, Resolving a Domain Name, DNS Name Registration. Network Printing Concepts: Locally Connected Print Devices, Setting up local Print Devices, Shared Print Devices, Sharing Locally Attached Print Devices, Describe Windows Network Printing and Add print Wizard.

UNIT 3 Designing Network – Accessing Network Needs, Applications, Users, Network Services, Security and Safety, Growth and Capacity Planning, Meeting Network Needs – Choosing Network Type, Choosing Network Structure, Choosing Servers. Installing and Configuring Windows Server - Preparing for Installation, Creating windows server boot disk, Installing windows server, Configuring server/ client. Setting windows server - Creating Domain controller, Adding the DHCP and WINS roles, Adding file server and print server, Adding Web based Administration.

UNIT 4 Working With User Accounts - Adding a User, Modifying User Account, Deleting or Disabling a User Account. Working With Windows Security Groups – Creating Group, Maintaining Group Membership. Working with Shares – Understanding Share Security, Cresting Shares, Mapping Drives. Administering Printer Shares – Setting up Network Printer. Working with Windows Backup – Using Windows Servers Backup Software

UNIT 5 Understanding the Problem – Troubleshooting, Segmenting the Problem, and Isolating the Problem, Setting Priorities. Troubleshooting Tools – Hardware Tools, Software Tools, Monitoring and Troubleshooting Tools

DIRECT ASSESMENT:
ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. The Complete Reference Networking Craig Zacker Tata McGraw Hill
2. Introduction to Networking Bruce Hallberg Tata McGraw-Hill
3. Networking + Certification Training Kit Richard, A. McMahon, Microsoft Press
4. The Real World Network Troubleshooting Manual Alan Sugano Firewall Media
5. MCSE Training Kit Networking Essential Plus, Microsoft Press

DEPARTMENT OF COMPUTER ENGINEERING
B. Tech. IV YEAR (4YDC)
SEMESTER-B (ELECTIVE-6)
CO 44706: SOFTWARE PROJECT MANAGEMENT

*Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
2	-	-	2	-	70	30	-	-	100

PRE-REQUISITES: CO34014: Agile Software Methodology

COURSE OBJECTIVES: To enable students to understand the fundamental principles of Software Project Management and be familiar with the process and techniques used for software project management.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Demonstrate basic concepts and issues of software project management.
2. Demonstrate Planning, Execution and Evaluation of software projects.
3. Apply mechanisms for monitoring, tracking and risk management of software projects.
4. Design activities necessary to perform quality management and successful completion of Software Projects.

COURSE CONTENTS:
THEORY:

UNIT 1 Project Evaluation and Project Planning: Importance of Software Project Management, Activities Methodologies, Categorization of Software Projects, Setting Objectives, Management Principles, Management Control, Project portfolio Management, Cost, Benefit Evaluation Technology, Stepwise Project Planning.

UNIT 2 Project Life Cycle and Effort Estimation: Software Process and Process Models, Choice of Process Models, Agile Methods: Importance, Introduction to Extreme Programming and SCRUM; Basics of Software Estimation: Effort and Cost Estimation Techniques.

UNIT 3 Activity Planning and Risk Management: Objectives of Activity Planning, Project Schedules, Activities, Sequencing and Scheduling, Network Planning Models, Forward Pass and Backward Pass Techniques, Risk Identification, Assessment, Monitoring, Resource Allocation, Creation of Critical Patterns, Cost Schedules.

UNIT 4 Project Management, Control and Staffing: Framework for Management and Control, Visualizing Progress, Cost Monitoring, Earned Value Analysis, Project Tracking, Change Control, Software Configuration Management, Managing People, Motivation, Decision Making, Team Structures, Project Closure: Analysis, Report Generation.

UNIT 5 Quality Management: Introduction, Product versus Process Quality Management, Quality Management Systems and Planning, Process Capability Models, Testing and Defect Prevention Planning; Case Studies on Project Management.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Bob Hughes, Mike Cotterell and Rajib Mall, “Software Project Management”, 5th Edition, Tata McGraw Hill, New Delhi, 2012.
2. Pankaj Jalote, “Software Project Management in Practice”, Pearson Publication, 2014.

REFERENCE BOOKS:

1. S. A. Kelkar, “Software Project Management: A Concise Study”, 3rd Edition, PHI Publication, 2013.
2. Robert K. Wysocki, “Effective Software Project Management”, Wiley Publication, 2011.
3. Walker Royce, “Software Project Management”, Addison Wesley, 1998.
4. Gopalaswamy Ramesh, “Managing Global Software Projects”, McGraw Hill Education (India), 14th Reprint, 2013.

DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. IV YEAR (4YDC)
SEMESTER-B (ELECTIVE-6)
CO 4____: IMAGE PROCESSING AND COMPUTER VISION

*Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
2	-	-	2	-	70	30	-	-	100

PRE-REQUISITES: MA10501: Mathematics-II, MA24003: Mathematics-III, CO___: Computer Graphics

COURSE OBJECTIVES: This course offers in depth knowledge about image processing and computer vision which can be applied for advanced image processing like medical application, crime recognition etc.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Describe in-depth about theories, methods, and algorithms of image processing and computer vision.
2. Compare and contrast traditional algorithms of image processing and computer vision.
3. Examine the nature of a computer vision based problem at hand and determine appropriate and efficient solutions.
4. Design and implement image processing and computer vision algorithms for solving real-life problems.

COURSE CONTENTS:
THEORY:

UNIT 1 Digital Image Formation and low-level processing, Overview and State-of-the-art, Fundamentals of Image Formation, Transformation: Orthogonal, Euclidean, Affine, Projective, etc; Fourier Transform, Convolution and Filtering, Image Enhancement, Restoration, Histogram Processing. Perspective, Binocular Stereopsis: Camera and Epipolar Geometry; Homography, Rectification, DLT, RANSAC, 3-D reconstruction framework; Auto-calibration.

UNIT 2 Feature Extraction, Edges - Canny, LOG, DOG; Line detectors (Hough Transform), Corners - Harris and Hessian Affine, Orientation Histogram, SIFT, SURF, HOG, GLOH, RANSAC, morphological features, Scale-Space Analysis- Image Pyramids and Gaussian derivative filters, Gabor Filters and DWT.

UNIT 3 Image Segmentation, Region Growing, Edge Based approaches to segmentation, Graph-Cut, Mean-Shift, MRFs, Texture Segmentation; Object detection, Object localization, Region Analysis, Projective geometry, Inverse perspective Projection, Photogrammetry -from 2D to 3D, Image matching.

UNIT 4 Pattern Analysis, Clustering: K-Means, K-Medoids, Mixture of Gaussians, Classification: Discriminant Function, Supervised, Un-supervised, Semi-supervised; Classifiers: Bayes, KNN, ANN models; Dimensionality Reduction: PCA, LDA, ICA, Auto Encoders; Nonparametric methods. Background Subtraction and Modeling, Optical Flow, KLT, Spatio-Temporal Analysis, Dynamic Stereo; Motion parameter estimation.

UNIT 5 Shape from X, Light at Surfaces; Phong Model; Reflectance Map; Albedo estimation; Photometric Stereo; Use of Surface Smoothness Constraint; Shape from Texture, color, motion and edges.

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. "Digital Image Processing" by Rafael Gonzalez, Richard Woods, Pearson Publication, 4th edition, May 2017.
2. Richard Szeliski, Computer Vision: Algorithms and Applications, Springer-Verlag London Limited 2011.
3. Computer Vision: A Modern Approach, D. A. Forsyth, J. Ponce, Pearson Education, 2003.
4. Milan Sonka,VaclavHlavac, Roger Boyle: Image Processing, Analysis, and Machine Vision , 1st edition, Thomson Learning

REFERENCE BOOKS:

1. KrishnenduKar, Mastering Computer Vision with TensorFlow 2.x, 2020, Packt Publishing
2. "Hands-On Image Processing with Python: Expert techniques for advanced image analysis and effective interpretation of image data", by SandipanDey, Pakt Publication, January 2018.
3. "Python 3 Image Processing" by Ashwin Pajankar, BPB Publications, January, 2019.

B. E. IV YEAR (4YDC)
SEMESTER-B (ELECTIVE-6)
CO 44707: BLOCK CHAIN TECHNOLOGY

*Hours per Week			Th. Credit	Pr. Credit	MAXIMUM MARKS				
L	T	P			TH	CW	SW	Pr.	Total
2	-	-	2	-	70	30	-	-	100

PRE-REQUISITES: Data Structure and Network Security

COURSE OBJECTIVES:

1. To explain the functional/operational aspects of Blockchain Technology.
2. To illustrate different consensus and cryptographic algorithm applied in Blockchain.
3. To elaborate functional/operational challenges of different cryptocurrencylike Bitcoin.
4. To demonstrate designing of Smart contract based platform Ethereum and enterprise based Hyperledger fabric.

COURSE OUTCOMES:

After completing the course student should be able to:

1. Define the functional/operational aspects of Blockchain Technology.
2. Identify and anlyse the consensus and cryptographic algorithm applied in Blockchain system.
3. Define the functional/operational challenges of different cryptocurrency ecosystem.
4. Creating and developing an Ethereum Smart contract for real world problem and also compare contrast Ethereum and Hyperledgerblockchain platforms.

COURSE CONTENTS:

THEORY:

UNIT 1 Introduction to Blockchain:Need for Distributed Record Keeping,Blockchain architecture, blockheader detailed design, Consensus algorithms:Bitcoin consensus Proof of Work (PoW) , liveness and fairness - Proof of Stake (PoS) based Chains - Hybrid models (PoW + PoS),Types of Blockchain.

UNIT 2 Introduction to cryptographic basics for cryptocurrency - a short discription of Hashing, signature schemes, encryption schemes and elliptic curve cryptography veriable random functions,.

UNIT 3 Introduction to Bitcoin - Wallet - Blocks - Merkley Tree - hardness of mining - transaction verifiability, anonymity - forks - double spending - mathematical analysis of properties of Bitcoin. Blockchain 1.0,2.0 etc.

UNIT 4 Introduction to Ethereum - Ethereum Virtual Machine (EVM) - Wallets for Ethereum - Solidity - Smart Contract attacks on smart contracts , The Turing Completeness of Smart Contract Languages and verication challenges,

UNIT 5 Introduction to Blockchain Consensus Algorithm challenges, and solutions ,Modeling faults and adversaries,Byzantine Generals problem,and Zero Knowledge proofs and protocols in Blockchain

DIRECT ASSESMENT:

ASSESMENT OF THEORY-

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder. Bitcoin and cryptocurrency technologies: a comprehensive introduction. Princeton University Pre2016.

REFERENCE BOOKS:

1. Draft version of "S. Shukla, M. Dhawan, S. Sharma, S. Venkatesan, 'Blockchain Technology: Cryptocurrency and Applications', Oxford University Press, 2019.
2. Josh Thompson, 'Blockchain: The Blockchain for Beginnings, Guild to Blockchain Technology and Blockchain Programming', Create Space Independent Publishing Platform, 2017.

COMPUTER ENGINEERING DEPARTMENT

B.Tech. IV YEAR(4YDC)

CO4_____: DATA MINING

PERIOD PER WEEK			CREDITS			MAXIMUM MARKS				
T	P	Tu	T	P	Tu	THEORY		PRACTICAL		TOTAL MARKS
						Th.	CW	SW	Prac.	
2	-	-	2	-	-	70	30	-	-	100

PREREQUISITES: CO34451: Skill Development Lab

COURSE OBJECTIVES: This course focuses on enabling students to learn, interpret, and employ data mining algorithms, which include the study of statistical properties of data and preparing it suitably for applying data mining techniques focusing on performance evaluation of the models.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Describe in depth about different data preprocessing techniques.
2. Examine the nature of the data at hand and determine the best suitable data mining algorithm.
3. Compare and Contrast Different Data Mining algorithms and their pragmatic usages.
4. Find and analyze the optimal hyper parameters of the data mining model.

UNIT-1 Introduction, Knowledge Discovery Process, Data Preprocessing, Association Rules, Apriori algorithm, Rule generation, Classification algorithms, Decision Trees, entropy, information gain, gini index, Regression Trees, Bayes Classifier, Multivariate Bayes Classifier

UNIT-2 K-Nearest Neighbour, Distance-weighted k-NN, Issues, Distance Metrics, Condensed Nearest Neighbor, Classifier Evaluation- confusion matrix, cost matrix, Model Evaluation, ROC Curve.

UNIT-3 Support Vector Machines - Problem formulation, Interpretation & Analysis, hard and soft margin, Hinge loss, SVM dual, SVM tuning parameters, SVM Kernels.

UNIT-4 Artificial Neural Networks - Perceptron, Gradient Decent, Backpropagation Algorithm, Linear Regression, Dimensionality Reduction: PCA.

UNIT-5 Clustering:Partition Clustering, K-Means Clustering, Mean Shift Clustering,Hierarchical Clustering, Agglomerative clustering,Density-based Clustering

DIRECT ASSESMENT:

ASSESMENT OF THEORY

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).
3. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Introduction to Data Mining, Tan, Steinbach and Vipin Kumar, Pearson Education, 2016
2. Data Mining: Concepts and Techniques, Pei, Han and Kamber, Elsevier, 2011

REFERENCE BOOKS:

1. Pang-Ning Tan, “Introduction to Data Mining”, Pearson Edu., 2007.

COMPUTER ENGINEERING DEPARTMENT
B.Tech. IV YEAR(4YDC)
Elective-6 (Sem B)

CO4_____: USER CENTRIC COMPUTING FOR HUMAN COMPUTER INTERACTION

PERIOD PER WEEK			CREDITS			MAXIMUM MARKS				
T	P	Tu	T	P	Tu	THEORY		PRACTICAL		TOTAL MARKS
						CW	Th.	SW	Prac.	
2	-	-	2	-	-	30	70	-	-	100

PREREQUISITES: CO34451: Skill Development Lab

COURSE OBJECTIVES: This course, delivers the knowledge and skill for developing user friendly interfaces and user centric design of computation.

COURSE OUTCOMES:
After completing the course student should be able to:

1. Describe the concept of usability and asthetics in designing user interfaces.
2. Apply concept of model based design for developing UIs
3. Design UIs using FSMs, state charts & petrinets.
4. Design & model UIs based on object oriented modeling.

UNIT-1 Introduction: Course objective andoverview, Historical evolution ofthe field. Interactive system design(theory and practice): Concept of usability -definition and elaboration, HCI and softwareengineering, GUI design andaesthetics, Prototypingtechniques.

UNIT-2 Model-based Design and evaluation: Basic idea, introduction to different types of models, GOMS family of models (KLM and CMN-GOMS), Fitts' law and Hick- Hyman's law, Model-based design case studies. Guidelines in HCI: Shneiderman's eight golden rules, Norman's seven principles, Norman's model of interaction, Nielsen's ten heuristics with example of its use, Heuristic evaluation, Contextual inquiry, Cognitive walkthrough.

UNIT-3 Empirical research methods in HCI: Introduction (motivation, issues, research question formulation techniques), Experiment design and data analysis (with explanation of one-way ANOVA), Hierarchical task analysis (HTA), Engineering task models and Concur Task Tree (CTT).

UNIT-4 Dialog Design: Introduction to formalism in dialog design, design using FSM (finite state machines), State charts and (classical) Petri Nets in dialog design, Introduction to CA, CA types, relevance of CA in IS design, Model Human Processor (MHP).

UNIT-5 Object Oriented Programming: OOP- Introduction, OOM- Object Oriented Modeling of User Interface Design, Design -Case Studies

DIRECT ASSESMENT:
ASSESMENT OF THEORY

1. Internal Assessment for continuous evaluation, mid-term tests, Tutorials, Quizzes, Class Performance, etc. (30%).

2. End semester Theory Exam (70%).

INDIRECT ASSESMENT:

1. Feedback of students on attainment of cos.
2. Feedback of students on classroom learning.
3. External examiners feedback on Cos.

TEXT BOOKS RECOMMENDED:

1. Yvonne Rogers, Helen Sharp, Jennifer Preece; Interaction Design3rd Edition Wiley 2011
2. Frank Bentley,Edward Barrett Building Mobile ExperiencesMIP Press Cambridge 2012
3. Selected research papers (details will be provided at the end of relevant materials).
4. Jacob Nieilsen; *Useability Engineering*; Morgan Kaufmann,Academic Press, London, 1993.

REFERENCE BOOKS:

1. Dix A., Finlay J., Abowd G. D. and Beale R. Human Computer Interaction, 3rd edition, Pearson Education, 2005.
2. Preece J., Rogers Y., Sharp H., Baniyon D., Holland S. and Carey T. Human Computer Interaction, Addison-Wesley, 1994.
3. B. Shneiderman; Designing the User Interface, Addison Wesley 2000 (Indian Reprint).