Course Curriculum and Detailed Syllabi

For

Master of Computer Applications (MCA)

Effective for

Students admitted in the

Academic Session 2020-21

Department of Computer Science & Engineering

School of Engineering

Harcourt Butler Technical University, Kanpur

Kanpur-208002

1. About the Department

The Department of Computer Science & Engineering was established in 1984 with a 4-year B. Tech. program in Computer Science & Engineering having an intake of 30 students. A 2-year Post Graduate Program, Master of Computer Application (MCA), with an intake of 60 students was introduced in 1987. Under IT task force recommendations, B. Tech. Information Technology Program with an intake of 60 students was introduced in the year 2000. Currently, the department is running B. Tech. Computer Science & Engineering, B. Tech. Information Technology and MCA with students' intake of 60, 30 and 60 respectively. One of the youngest, but among the most efficient departments, it is reputed for producing the best quality software engineers who serve in leading companies in India and abroad. The students have an in-depth exposure to computing environment consisting of state-of-the-art machines in different laboratories. In order to identify Industrial projects for the students and to expose them to the industrial environment, the department has continuous interaction with the Industries.

2. Vision

To excel in Computer Science & Engineering education, research, innovation and global employability.

3. Mission

1. Achieve academic excellence in Computer Science & Engineering through an innovative teaching-learning process.
2. Inculcate technical competence and collective discipline in students to excel for global employability, higher education and societal needs.
3. Establish focus research groups in leading areas of Computer Science & Engineering.
4. Sustain quality in Computer Science & Engineering education & research through continuous & rigorous assessment.

4. Program Educational Objectives (PEOs)

1. To inculcate professional culture amongst the students to take up technical/ professional positions for design, development, and problem solving in software industries and R&D organizations.
2. To prepare students as technical, ethical, responsible solution providers and entrepreneurs in various areas of computer applications.
3. To provide the necessary competence and capability in students to pursue higher studies in Institutions of International / National repute.
4. To provide analytical and technical ability to develop and innovate software systems and technologies in the leading areas of computer applications.

5. Program Outcomes (POs)

Engineering Graduates will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

6. Program Specific Outcomes (PSOs)

By the completion of Master of Computer Applications program, students will achieve the following program specific outcomes:-

1. The ability to understand, analyse and develop applications in the field of algorithms, system software, databases, web design, networking and artificial intelligence.
2. The ability to apply standard practices and strategies in software project development using suitable programming environment to deliver quality products.
3. The ability to employ modern computer languages, environments, and platforms in creating innovative career paths, to be an entrepreneur, and a zest for higher studies.
4. The ability to use research based knowledge to do literature survey, formulate problem, design & carry-out experimentation, analyse & interpret experimental results for complex research problems.

7. Consistency/Mapping of PEOs with Mission of the Department

PEO Statements	M1	M2	M3	M4
PEO1: To inculcate professional culture amongst the students to take up technical/ professional positions for design, development, and problem solving in software industries and R&D organizations.	2	3	2	3
PEO2: To prepare students as technical, ethical, responsible solution providers and entrepreneurs in various areas of computer applications.	3	3	1	2
PEO3: To provide the necessary competence and capability in students to pursue higher studies in Institutions of International / National repute.	2	3	3	2
PEO4: To provide analytical and technical ability to develop & innovate systems and technologies in the leading areas of computer applications.	3	3	3	2

1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High) "-": No correlation

8. Components of the curriculum

(*Program curriculum grouping based on course components)*

Sr. No	Curriculum Content			
	Course Component	**% of total number of credits of the Program**	**Total number of contact hours**	**Total Credits**
1.	Basic Sciences (BSC)	6.00	06	06
2.	Humanities and Social Sciences (HMSC)	4.00	05	04
3.	Program Core (PCC)	58.00	66	58
4.	Program Electives (PEC)	16.00	16	16
5.	Summer Training/Internship & Seminar Course (ISC)	3.00	06	03
6.	Project(s) (PRC)	13.00	26	13
	Total Credits	**100**	**130**	**100**

Department of Computer Science & Engineering

Course Structure (Semester wise)

Master of Computer Applications

(Applicable w.e.f. the Session 2020-21)

Semester-I

Sr.	Course	Course	Course Name	Credits	Details of Sessional Marks				ESM	Total
No.	Type	Code			CT	TA	Lab.	Total		Marks
1	PCC	ECA-451	Computer Concepts & Programming in C	5 (3-1-2)	15	20	15	50	50	100
2	PCC	ECA-453	Computer Organization	4 (3-1-0)	30	20	-	50	50	100
3	PCC	ECA-455	Internet & Java Programming	5 (3-1-2)	15	20	15	50	50	100
4	PCC	ECA-457	Operating Systems	4 (3-1-0)	30	20	-	50	50	100
5	BSC	BMA-451	Discrete Mathematical Structures	3 (3-0-0)	30	20	-	50	50	100
6	HSMC	HHS-454	Fundamentals of Management	4 (3-0-2)	15	20	15	50	50	100
			Total Credits	**25**						

Semester-II

Sr.	Course	Course	Course Name	Credits	Details of Sessional Marks				ESM	Total
No.	Type	Code			CT	TA	Lab.	Total		Marks
1	PCC	ECA-452	Software Engineering	4 (3-1-0)	30	20	-	50	50	100
2	PCC	ECA-454	Design & Analysis of Algorithms	4 (3-1-0)	30	20	-	50	50	100
3	PCC	ECA-456	Database Management Systems	5 (3-1-2)	15	20	15	50	50	100
4	PCC	ECA-458	Data Structures Using C	5 (3-1-2)	15	20	15	50	50	100
5	PCC	ECA-460	Computer Graphics & Animation	4 (3-0-2)	15	20	15	50	50	100
6	BSC	BMA-452	Operations Research	3 (3-0-0)	30	20	-	50	50	100
			Total Credits	**25**						

Semester-III

Sr. No.	Course Type	Course Code	Course Name	Credits	Details of Sessional Marks				ESM	Total Marks
					CT	TA	Lab.	Total		
1	PCC	ECA-551	Computer Networks	5 (3-1-2)	15	20	15	50	50	100
2	PCC	ECA-553	Object Oriented Systems Modeling	5 (3-1-2)	15	20	15	50	50	100
3	PCC	ECA-555	Software Project Management	4 (3-0-2*)	15	20	15	50	50	100
3	PCC	ECA-557	Artificial Intelligence	4 (3-1-0)	30	20	-	50	50	100
5	PEC	ECA-	Programme Elective-I	4 (3-1-0)	30	20	-	50	50	100
6	ISC	ECA-559	Summer Training/ Internship and Seminar	3 (0-0-6)	-	100	-	100	-	100
		Total Credits		**25**						

*During Practical hours students will do a Minor Project which may be extended as Major Project in Semester-IV

Programme Elective-I

1. Theory of Automata & Formal Languages (ECA-561)
2. Simulation & Modeling (ECA-563)
3. Information Security & Cyber Laws (ECA-565)
4. E_Commerce (ECA-567)
5. Internet of Things (ECA-569)

Semester-IV

Sr. No.	Course Type	Course Code	Course Name	Credits	Details of Sessional Marks				ESM	Total Marks
					CT	TA	Lab.	Total		
1	PEC	ECA-	Programme Elective-II	4 (3-1-0)	30	20	-	50	50	100
2	PEC	ECA-	Programme Elective-III	4 (3-1-0)	30	20	-	50	50	100
3	PEC	ECA-	Programme Elective-IV	4 (3-1-0)	30	20	-	50	50	100
4	PRC	ECA-592	Project	13 (0-0-26)	-	50	-	50	50	100
			Total Credits	**25**						

Programme Elective-II

1. Compiler Design (ECA-552)
2. Machine Learning (ECA-554)
3. Advanced Database Management Systems (ECA-556)
4. ERP Systems (ECA-558)
5. AI Programming using Pythan (ECA-560)

Programme Elective-III

1. Cloud Computing (ECA-572)
2. Software Quality Engineering (ECA-574)
3. Digital Image Processing (ECA-576)
4. Data Warehousing & Mining (ECA-578)
5. Mobile Application Development (ECA-580)

Programme Elective-IV

1. Cryptography & Network Security (ECA-582)
2. Soft Computing (ECA-584)
3. Embedded System (ECA-586)
4. Virtual Reality (ECA-588)
5. Data Analytics (ECA-590)

Detailed Syllabus

Ist Year

COMPUTER CONCEPTS & PROGRAMMING IN 'C' (ECA-451)

Type	L	T	P	Credits
ESC	3	1	2	5

Prerequisite: NIL

Course Content:

Unit-1:

Introduction to Computers: Computer hardware Components, peripherals and their functions, Number Systems and conversion methods, Concept of an algorithm; termination and correctness. Algorithms to programs: specification, top-down development and stepwise refinement, Introduction to programming environment, use of high level programming language for the systematic development of programs. Introduction to the design and implementation of correct, efficient and maintainable programs, Structured Programming, Trace an algorithm to depict the logic.

Unit-2:

Basic operating System Concepts: Introduction of MS-DOS, WINDOWS, and LINUX Operating Systems, Functional Knowledge of these operating systems, Introduction of basic commands of LINUX and Editors, Managing Files and Directories in LINUX, Programming Environment in LINUX, Writing and executing programs in LINUX.

Unit-3:

Programming in C: History, Introduction to C Programming Languages, Structure of C programs, compilation and execution of C programs, Debugging Techniques, Data Types and Sizes, Declaration of variables, Modifiers, Identifiers and keywords, Symbolic constants, Storage classes (automatic, external, register and static), Enumerations, command line parameters, Macros, The C Preprocessor.

Unit-4:

Operators: Unary operators, Arithmetic & logical operators, Bit wise operators, Assignment operators and expressions, Conditional expressions, Precedence and order of evaluation. Control statements: if-else, switch, break, and continue, the comma operator, goto statement. Loops: for, while, do-while. Functions: built-in and user-defined, function declaration, definition and function call, and parameter passing: call by value, call by reference, recursive functions, Multi-file programs. Arrays: linear arrays, multidimensional arrays, passing arrays to functions, Arrays and strings.

Unit-5:

Structure and Union: definition and differences, self-referential structure. Pointers: value at (*) and address of (&) operator, pointer to pointer, Dynamic Memory Allocation, calloc and malloc functions, array of pointers, function of pointers, structures and pointers. File Handling in C: opening and closing a data file, creating a data file, read and write functions, unformatted data files.

Lab Work:

1. Write C program to find largest of three integers.
2. Write C program to check whether the given string is palindrome or not.
3. Write C program to find whether the given integer is

(i). a prime number
(ii). an Armstrong number.

4. Write C program for Pascal triangle.
5. Write C program to find sum and average of n integer using linear array.
6. Write C program to perform addition, multiplication, transpose on matrices.
7. Write C program to find Fibonacci series of iterative method using user-defined function.
8. Write C program to find factorial of n by recursion using user-defined functions.
9. Write C program to perform following operations by using user defined functions:
   (i) Concatenation
   (ii) Reverse
   (iii) String Matching
10. Write C program to find sum of n terms of series: n – n*2/2! + n*3/3! - n*4/4! + ...
11. Write C program to interchange two values using
    (i). Call by value.
    (ii). Call by reference.
12. Write C program to sort the list of integers using dynamic memory allocation.
13. Write C program to display the mark sheet of a student using structure.
14. Write C program to perform following operations on data files:
    (i) Read from data file.
    (ii) Write to data file.
15. Write C program to copy the content of one file to another file using command line argument.

Text and References Books

1. Kernighan, Ritchie, "The C Programming Language", PHI
2. V. Rajaraman, "Fundamentals of Computers", PHI
3. Peter Norton's, "Introduction to Computers", TMH
4. Gottfried, "Programming in C", Schaum's Series, Tata McGraw Hill
5. Yashwant Kanitkar, "Working with C", BPB
6. E. Balagurusamy, "Programming in ANSI C", TMH

Course Outcomes

1. Identify the parts of the computer system and explain the functioning of its components alongwith the process of problem solving. (Remember, Understand)
2. Design an algorithmic solution for a given problem and translate it into a program. (Design)
3. Understand different operating systems, related concepts and their functions. (Understand)
4. Use the appropriate control statements to solve the given problem. (Apply)
5. Implement different Operations on arrays and use functions to solve the given problem. (Apply)
6. Understand pointers, structures and unions & Implement file Operations in C programming. (Understand, Apply)

CO and PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	0	0	0	0	0	2	0	0	2
CO2	3	3	3	0	0	0	0	0	2	0	0	2
CO3	3	3	1	0	0	0	0	0	2	0	0	2
CO4	3	3	3	0	0	0	0	0	2	0	0	2
CO5	3	3	3	0	0	0	0	0	2	0	0	2
CO6	3	3	3	0	0	0	0	0	2	0	0	2

CO and PSO Mapping

CO/PSO	PSO1	PSO2	PSO3	PSO4
CO1	3	2	1	1
CO2	3	2	1	1
CO3	3	2	1	1
CO4	3	2	1	1
CO5	2	1	-	-

COMPUTER ORGANIZATION (ECA-453)

Type	L	T	P	Credits
PCC	**3**	**1**	**0**	**4**

Prerequisite: NIL

Course Content:

Unit-1:

Representation of Information and Basic Building Blocks: Introduction to Computer, Computer hardware generation, Number System: Binary, Octal, Hexadecimal, Character Codes (BCD, ASCII, EBCDIC), Logic gates, Boolean Algebra, K-map simplification, Half Adder, Full Adder, Subtractor, Decoder, Encoders, Multiplexer, De-Multiplexer, Carry look ahead adder, Combinational logic Design, Flip-Flops, Registers, Counters (synchronous & asynchronous), ALU, Micro-Operation, ALU Chip, Faster Algorithm and Implementation (Multiplication & Division).

Unit-2:

Basic Organization: Von Neumann Architecture, Operational flow chart, Instruction Cycle, Organization of Central Processing Unit, Hardwired & micro programmed control unit, Single Organization, General Register Organization, Stack Organization, Addressing modes, Instruction formats, data transfer & Manipulation, I/O Organization, Bus Architecture, Programming Registers

Unit-3:

Memory Organization: Memory Hierarchy, Main memory (RAM/ROM chips), Auxiliary memory, Associative memory, Cache memory, Virtual Memory, Memory Management Hardware, hit/miss ratio, magnetic disk and its performance, magnetic Tape etc.

Unit-4:

I/O Organization: Peripheral devices, I/O interface, Modes of Transfer, Priority Interrupt, Direct Memory Access, Input Output Processor and Serial Communication. I/O Controllers, Asynchronous data transfer, Strobe Control, Handshaking.

Unit-5:

Process Organization: Basic Concept of 8-bit micro Processor (8085) and 16-bit Micro Processor (8086), Assembly Instruction Set, Assembly language program of (8085): Addition of two numbers, Subtraction, Block Transfer, find greatest number, Table search, Numeric Manipulation, Introductory Concept of pipeline, Flynn's and Feng's Classification, Parallel Architectural classification, Concept of Pipelining and Multi-Core Architecture.

Text and References Books

1. William Stalling, "Computer Organization & Architecture", Pearson education Asia
2. Mano Morris, "Computer System Architecture", PHI
3. Zaky & Hamacher, "Computer Organization", McGraw Hill
4. B. Ram, "Computer Fundamental Architecture & Organization",
5. New Age, A.S. Tannenbaum, "Structured Computer Organization", PHI.

Course Outcomes

1. Understand Number systems, Logic Gates, Boolean algebra, Design of Combinational and sequential circuits. (Understand)
2. Understand Von Neumann architecture, instruction cycle and the concept of Hardwired and Micro programmed control unit, addressing modes, register organization. (Understand)
3. Apply the concepts of memory organization in calculating hit-miss ratio and access time of magnetic disks. (Apply)
4. Understand the working of various I/O devices, buses, interrupt and interfaces etc. (Understand)
5. Understand the basics of pipelining and Multicore architecture. (Understand)
6. Design and implement systems using 8085 and 8086 microprocessor with the knowledge of pin diagram, interrupts and instruction format by writing assembly language programming. (Analyze)

CO and PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	1	1	-	-	-	-	-	-	-	-	-
CO2	2	2	1	1	-	-	-	-	-	-	-	-
CO3	2	1	1	1	-	-	-	-	-	-	-	-
CO4	2	1	1	-	-	-	-	-	-	-	-	-
CO5	1	1	1	-	-	-	-	-	-	-	-	-
CO6	2	1	3	2	-	-	-	-	-	-	-	2

CO and PSO Mapping

CO/PSO	PSO1	PSO2	PSO3	PSO4
CO1	2	-	2	-
CO2	2	-	1	-
CO3	2	2	1	-
CO4	2	-	1	-
CO5	2	-	1	-
CO6	2	2	2	1

INTERNET & JAVA PROFRAMMING (ECA-455)

Type	L	T	P	Credits
PCC	3	1	2	5

Prerequisite:

Course Content:

Unit-1

Introduction to Internet and Internet Services, HTML: Formatting tags, Links, Lists, Tables, Frames, Forms, Comments in HTML, DHTML

Core Java: Introduction, Operator, Data types, Variables, Arrays, Control Statements, Methods & Classes, Inheritance, Packages and Interface.

Unit-2

Core Java: Exception Handling, Multithread Programming, I/O, Applet, String handling, Networking, Event Handling, Introduction to AWT, AWT controls, Layout managers, Menus, Images, Graphics, JDBC

Unit-3

Java Servelets: Servelet Life Cycle, HTTP Servelet Class, Request Interface, Response Interface, Session Tracking (Cookies VRL)

JSP: Overview, Relation of Applet and Servelet with JSP, Scripting Element, JSP Expressions, JSP Scriplets, Predefined Variables, Creating Custom JSP Tag Libraries, Using Nested Tags, Structuring Generated Servelet in JSP Pages, Including Files and Applets in JSP Documents, Integrating Servelet and JSP.

Unit-4

Java Swing: Creating a Swing Applet and Applications, Programming using Panes, Pluggable Look and Feel, Labels, Text Fields, Buttons, Toggle Buttons, Checkboxes, Radio Buttons, View Port, Scroll Panes, Scroll Bars, Lists, Combo Box, Progress Bar, Menus and Toolbars, Layered Panes, Tabbed Panes, Split Panes, Layouts, Windows, Dialog Boxes, Inner Frame.

Unit-5

Java Beans: Application Builder Tools, The Bean Developer Kit (BDK), JAR Files, Introspection, Developing a Simple Bean, Using Bound Properties, The Java Bean API, Session Beans, Entity Beans, Introduction to Enterprise Java Beans (EJB), Introduction to Remote Method Invocation (RMI): A Simple Client-Server Application using RMI.

Lab Work

1. Design a HTML page to display your CV
2. Design a HTML form to reserve a railway ticket.
3. Write a Java Script program that finds the greatest common divisor of two numbers.
4. In the form mentioned in problem 2 to reserve a railway ticket add the following validations using java script.
   - From city and to city are two different cities.
   - Age of passengers should not be greater than 150.
   - Name of the passenger should be a string of a maximum length 20.
5. Write a program for illustrating client/server side scripting with help of ASP.
6. Write a piece of code in XML for creating DTD, which specifies set of rules.
7. Create style sheet in CSS/XSL and display the document in Internet Explorer.

Text and Reference Books

1. Margaret Levine Young, "The Complete Reference Internet", TMH
2. Naughton Schildt, "The Complete Reference JAVA2", TMH
3. Balagurusamy E, "Programming in JAVA", TMH
4. Dustin R. Callway, "Inside Servlets", Addison Wesley
5. Mark Wutica, "Java Enterprise Edition", QUE
6. Steven Holzner, "Java2 Black Book", Dreamtech Media

Course Outcomes

1. Understand the basics of web and apply the web concepts for web application development. (Apply)
2. Understand, apply and analyze mark-up languages like HTML, DHTML, and XML for development of different web applications. (Apply, Analyze)
3. Develop interactive web applications using client-side scripting languages. (Apply)
4. Develop and deploy web services to build the server side components in web applications. (Apply)
5. Understand and develop applications using EJB and RMI concepts. (Understand, Apply)

CO and PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	2	-	-	-	-	-	-	-	-	-
CO2	2	2	2	-	2	-	-	-	-	-	-	-
CO3	2	1	3	1	2	-	-	-	-	-	-	-
CO4	2	1	3	1	2	-	-	-	-	-	-	-
CO5	2	1	3	1	2	-	-	-	-	-	-	-

CO and PSO Mapping

CO/PSO	PSO1	PSO2	PSO3	PSO4
CO1	2	2	2	-
CO2	3	2	3	-
CO3	3	2	3	1
CO4	3	2	3	1
CO5	3	2	3	1

OPERATING SYSTEMS (ECA-457)

Type	L	T	P	Credits
PCC	3	1	0	4

Prerequisite:

Course Content:

Unit-1:

Introduction: Operating System and its functions, Evolution of Operating System, Batch, Interactive, Time Sharing and Real Time Operating System, System Protection. Operating System Structure: System Components, System Structure, Operating System Services.

Unit-2:

Process Management: Process Concept, Process State, Process Control Block, Threads. Concurrent Processes: Principle of Concurrency, Mutual Exclusion, Inter Processes Communication, Critical Section Problem, Semaphores, Classical Problems in Concurrency, Producer / Consumer Problem, Readers-Writers Problem, Dining Philosophers Problem.

Unit-3:

CPU Scheduling: Scheduling Concept, Scheduling Techniques, Performance Criteria for Scheduling Algorithm, Evolution, Multiprocessor Scheduling. Deadlock: System Model, Deadlock Characterization, Prevention, Avoidance and Detection, Recovery From Deadlock Combined Approach.

Unit-4:

Memory Management: Basic Machine, Resident Monitor, Multiprogramming with Fixed Partition, Multiprogramming With Variable Partition, Multiple Base Register, Paging, Segmentation, Paged Segmentation. Virtual Memory: Virtual Memory Concept, Demand Paging, Performance, Paged Replaced Algorithm, Allocation of Frames, Thrashing, Cache Memory Organization, Impact on Performance.

Unit-5:

I/O Management & Disk Scheduling: I/O Devices, Organization of I/O Function, Operating System Design Issues, I/O Buffering, Disk Scheduling (FCFS, SCAN, CSCAN). File Management: File Concept, File Organization, File Directories, File Sharing, Allocation Methods, Free Space Management, and Implementation Issues.

Text and References Books:

1. Milenekovik, "Operating System Concept", McGraw Hill.
2. Petersons, "Operating Systems", Addison Wesley.
3. Dietal, "An Introduction to Operating System", Addison Wesley.
4. Tannenbaum, "Operating System Design and Implementation", PHI.
5. Gary Nutt, "Operating System, A Modern Perspective", Addison Wesley.
6. Stalling, Williams, "Operating System", Maxwell Macmillan
7. Silveschatz, Peterson J., "Operating System Concepts", Willey.
8. Crowley, "Operating System", TMH.

Course Outcomes:

1. Understand types and structure of operating systems. (Understand)
2. Construct solutions for problems related to process scheduling, deadlocks and synchronization in a multi-programmed operating system. (Apply)
3. Develop appropriate solutions for memory management considering challenges due to multi-programming and virtual memory. (Apply)
4. Apply knowledge of various software and hardware synchronization tools for solving critical section problem in concurrent processes. (Apply)
5. Construct solutions for problems related to secondary storage management with an understanding of file systems and disk scheduling. (Apply)
6. Design various system protection and security mechanisms in order to design efficient software system. (Apply)

CO and PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	-	-	-	-	-	-	-	-	-	-	-
CO2	2	2	3	-	-	-	-	-	-	-	-	-
CO3	2	2	3	-	-	-	-	-	-	-	-	-
CO4	2	2	3	1	2	-	-	-	-	-	-	2
CO5	2	2	3	-	-	-	-	-	-	-	-	-
CO6	2	2	3	1	-	-	-	-	-	-	-	2

CO and PSO Mapping

CO/PSO	PSO1	PSO2	PSO3	PSO4
CO1	2	-	-	-
CO2	3	3	2	-

CO3	3	3	2	-
CO4	3	3	3	1
CO5	3	2	2	-
CO6	3	2	2	1

DISCRETE MATHEMATICAL STRUCTURES (BMA-451)

Type	L	T	P	Credits
BSC	3	0	0	3

Prerequisite:

Course Content:

UNIT I: Fundamentals of Logic

Propositional Logic: Propositions, Basic logic operations and truth tables, Tautologies, Contradictions, Contingency, Algebra of propositions, Logical equivalence: the laws of logic, Logical implication: Rules of inference, Logical analysis of arguments, Some computing applications (Normal forms), Functionally complete set of operations, Formal proofs.

First Order Logic: Predicates & quantifiers, Nested quantifiers, Use of quantifiers, Rules of inference, Validity of arguments.

Notion of Proofs: Proof by counter example, the contraposition, proof by contradiction, inductive proofs.

UNIT II: Set Theory, Relations and Functions

Set Theory: sets & subsets, Venn diagrams, Set operations and laws, countable and uncountable sets, Cartesian product, Cardinality, Principle of inclusion- exclusion.

Relations: Relation, Representation & properties, n-ray relations and applications, Composition of relations, Closures of relations, Equivalence relation & partitions, partial orders, compatibility relation.

Functions: Functions and its types, Inverse function, Composition of functions, Special functions, Recursively defined functions, Computational Complexity, Analysis of algorithms.

Theorem Proving Techniques: Mathematical induction, strong induction, and well ordering, structural induction, Pigeonhole principle.

UNIT III: Algebraic Structures and Coding Theory

Algebraic Structures: Definition, Properties, Semi group, Monoid, Group, Properties of groups, Subgroup, Cyclic group, Cosets and Lagrange's theorem, Permutation groups, Normal subgroup, Homomorphism and isomorphism of groups, Congruence relation, Rings and Fields. Examples and standard results.

Coding Theory: Elements of coding theory, Hamming matric, Parity-check and generator matrices, Coding and error detection, Group codes: decoding with coset leaders and error correction, Hamming matrices.

UNIT IV: Partially Ordered Structures

Posets: Definitions, ordered set, Hasse diagram, isomorphic ordered set, well ordered set, Minimal and Maximal elements, LUB & GLB etc.

Lattices: Definition & Properties, Product Lattices, Isomorphic Lattices, Applications, Types of Lattices.

Boolean Algebras: Definitions & Properties, SOP & POS forms, Logic gates and minimization of circuits, Karnaugh maps, Quine-McClusky method.

Trees: Definition & Examples and Properties, Rooted tree, Binary tree, Tree traversal, application in computer science and engineering.

UNIT V: Combinatorics and Graph Theory:

Combinatorics: Basic counting techniques, Discrete numeric functions and properties, Recurrence relations and their applications (modelling), various methods of solutions, system of recurrence relations, OGF & EGF, properties, applications: solution of recurrence relations and combinatorial problems. Polya's enumeration theorem and applications.

Graphs: Graphs and graph models, terminology, matrices associated with graphs, Isomorphism, Special types of graphs, connectedness, Euler and Hamilton graphs with their applications, trees with properties, MST, planer graphs and applications, criteria of planarity, Graph coloring and coloring models, directed graphs.

Text and Reference Books:

1. Trembley, J.P. & R. Manohar, "Discrete Mathematical Structures with applications to Computer Science", McGraw Hill.
2. Kenneth H. Rosen, "Discrete Mathematics and its Applications", McGraw Hill.
3. Ralph, P. Garimaldi, "Discrete& Combinatorial Mathematics" Pearson Publication, Asia.
4. Deo, narsingh, "Graph Theory with applications to Engineering & Computer Science", PHI.
5. Krishnamurthy, V., "Combinatorics Theory & Application", East-West Press Pvt. Ltd., New Delhi.

Course Outcomes:

1. Understand concepts of Logic and various inference mechanisms using logic. (Understand)
2. Understand Set theory, functions, relations and the concepts of theorem proving. (Understand)
3. Explain algebraic structure and coding theory. (Understand)
4. Understand and apply concepts of partially ordered structures, Boolean algebra and trees in various application of computer science domain. (Understand, Apply)
5. Understand and apply graph theory and concepts of recurrence relation in system modeling. (Understand, Apply)

CO and PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	3	3	2	3	1	2	-	3	3
CO2	3	3	3	3	3	2	3	1	2	-	3	3
CO3	3	3	3	3	3	2	3	1	3	-	3	3
CO4	3	3	3	3	3	2	3	1	3	-	3	3
CO5	3	3	3	3	3	2	3	-	3	-	3	3

FUNDAMENTAL OF MANAGEMENT (HSS-454)

Type	L	T	P	Credits
HSMC	3	0	2	4

Prerequisite:

Course Content:

Unit 1: Principles of Management
Introduction, Functions of Management, Planning, Organising, Directing, Controlling, Coordinating

Unit 2: People Management
Introduction, Functions, Human Resource Planning, Recruitment, Selection, Performance Appraisal, Training, Salary Management

Unit 3: Finance and Marketing Management
Functions of Finance Management, Financing, Investment and Dividend Decisions; Functions of Marketing Management, E-retailing, Interactive advertising and IT

Unit 4: Behavioural Management
Organisational Structure, Motivation, Leadership, Team Management, Change Management, Conflict, Stress Management

Unit 5: Business Communication
Fundamentals of communication, Elements of written communication, Business letters, Technical Reports, Business Presentations, Listening.

Text Books

1. Robbins, S. P., Management, Prentice hall of India, 1998
2. Newstrom,J. , Organisational Behaviour: Human Behaviour at Work, McGraw Hill Education
3. L. M. Prasad, Management,
4. Business Communication, Monipally, Tata McGraw-Hill Publication

Reference Books

1. Fred Luthans, 'Organizational Behaviour', McGraw Hill Education, Asia, 2007.
2. Mamoria, C.B., Personnel Management, Himalayan Publishing, India
3. Dwivedi, R S , 'Human Relations and Organizational Behaviour: a Global Perspective', Macmillan India Ltd., Delhi
4. Krishna Mohan & Meera Banerjee , Developing Communication Skills, Macmillan India

Course Outcomes (CO)

1. Understand the fundamental principles of management.
2. Learn people management in an organization.
3. Learn basics of finance and marketing concept of businesses
4. Understand human behavior and the role in organizational context
5. Communicate their ideas in the contemporary global competitive environment effectively.

CO and PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	-	-	-	1	-	1	1	1	2	2	1	3
CO2	-	-	-	1	-	1	1	3	3	3	1	3
CO3	-	-	-	1	-	2	1	3	3	3	2	3
CO4	-	-	-	2	-	2	2	3	3	3	2	3
CO5	-	-	-	2	-	1	2	2	2	2	2	3

SOFTWARE ENGINEERING (ECA-452)

Type	L	T	P	Credits
PCC	3	1	0	4

Prerequisite:

Course Content:

Unit-1:

Introduction to Software Engineering, Software Components, Software Characteristics, Software Crisis, Software Engineering Processes, Similarity and Differences from Conventional Engineering Processes, Software Development Life Cycle (SDLC) Models: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models, Selection of Software Development Models,

Unit-2:

Software Requirement Specifications (SRS) Requirement Engineering Process: Elicitation, Analysis, Documentation, Review and Management of User Needs, Feasibility Study, Information Modeling, Data Flow Diagrams, Entity Relationship Diagrams, Decision Tables, SRS Document, IEEE Standards for SRS, Estimation of various Parameters such as Cost, Efforts, Schedule/Duration, Constructive Cost Models (COCOMO), Resource Allocation Models, Software Risk Analysis and Management.

Unit-3:

Software Design Basic Concept of Software Design, Architectural Design, Low Level Design: Modularization, Design Structure Charts, Pseudo Codes, Flow Charts, Coupling and Cohesion Measures, Design Strategies: Function Oriented Design, Object Oriented Design, Top-Down and Bottom-Up Design. Software Measurement and Metrics: Various Size Oriented Measures: Halestead's Software Science, Function Point (FP) Based Measures, Cyclomatic Complexity Measures: Control Flow Graphs.

Unit-4:

Software Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Testing for Functionality and Testing for Performance, Top-Down and Bottom-Up Testing Strategies: Test Drivers and Test Stubs, Structural Testing (White Box Testing), Functional Testing

(Black Box Testing), Test Data Suit Preparation, Alpha and Beta Testing of Products. Static Testing Strategies: Formal Technical Reviews (Peer Reviews), Walk Through, Code Inspection, Compliance with Design and Coding Standards.

Unit-5:

Software Maintenance: Software as an Evolutionary Entity, Need for Maintenance, Categories of Maintenance: Preventive, Corrective and Perfective Maintenance, Cost of Maintenance, Software Re-Engineering, Reverse Engineering, Software Configuration Management Activities, Change Control Process, Software Version Control, Defect Detection and Removal: Defect Amplification Model, An Overview of CASE Tools.

Lab Work:

Lab exercises or a Mini Project (as per list given below) to be carried out using languages like C++, Java, C# and tools like Visio, ARGOUML, Rational Rose etc. Design and Implementation of an Object based application using any one of the above languages/tools is desirable.

- Hotel Automation System
- Book Shop Automation Software
- Word processing Software
- Software Component Cataloguing Software
- Payroll System
- Banking System
- Purchase Order System
- Library Management System
- Railway Reservation System
- Bill Tracking System
- University Admission System
- Estate Management System.

Text and References Books:

1. R. S. Pressman, Software Engineering: A Practitioners Approach, McGraw Hill.
2. K. K. Aggarwal and Yogesh Singh, Software Engineering, New Age International Publishers.
3. Ian Sommerville, Software Engineering, Addison Wesley.
4. Pankaj Jalote, Software Engineering, Narosa Publication
5. Pfleeger, Software Engineering, Macmillan Publication.

Course Outcomes:

1. Understand and explain various concepts of software engineering and software life cycle development models. (Understand)
2. Prepare SRS and Compute cost and effort required to complete a given project, using various estimation techniques and models. (Apply)

3. Understand various concepts of Software design and Construct Data Flow Diagrams, Data Dictionaries and UML diagrams for a given software requirement specification. (Understand, Apply)
4. Understand various testing techniques and use these concepts to design optimal test cases. (Understand, Apply, Analyze)
5. Understand software configuration management, version control, reverse engineering, defect tracking etc. (Understand)
6. Build a project report as a team which contains the requirement specification, plan, schedule and design documents based on the knowledge of software development lifecycle. (Apply)

CO and PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	2	-	-	-	-	-	-	-	-	1
CO2	2	2	1	1	1	-	-	-	-	-	3	1
CO3	2	2	3	1	3	1	1	1	3	3	3	1
CO4	2	2	1	-	1	-	-	-	-	-	3	1
CO5	2	2	1	-	1	-	-	-	-	-	3	1
CO6	3	3	3	3	3	1	1	1	3	3	3	-

CO and PSO Mapping

CO/PSO	PSO1	PSO2	PSO3	PSO4
CO1	3	3	1	1
CO2	3	3	1	2
CO3	3	3	1	2
CO4	3	3	1	1
CO5	2	2	-	1
CO6	3	3	2	2

DESIGN & ANALYSIS OF ALGORITHMS (ECA-454)

Type	L	T	P	Credits
PCC	3	1	0	4

Prerequisite:

Course Content:

Unit-1:
Algorithms definition and introduction, Analysis of algorithms, Growth of Functions, Master's Theorem, Designing of Algorithms, Partitioning Algorithms, Divide and Conquer design and analysis techniques: Merge Sort and Quick Sort, Sorting and order Statistics: Heap sort, Sorting in linear time, Medians and Order Statistics.

Unit-2:
Advanced Data Structures: Introduction of Red-Black Trees, Augmenting Data Structure, B-Trees, Binomial Heaps, Fibonacci Heaps, Data Structure for Disjoint Sets, Amortized Analysis.

Unit-3:
Advanced Design and Analysis Techniques: Dynamic Programming, Greedy Algorithms, Back Tracking, Branch and Bound with their applications.

Unit-4:
Graph Algorithms: Elementary Graphs Algorithms, Minimum Spanning Trees, Single-source Shortest Paths, All-Pairs Shortest Paths, Traveling Salesman Problem and Maximum Flow

Unit-5:
Selected Topics: Randomized Algorithms, String Matching, Non-deterministic Algorithms: P, NP, NP Hard and NP Completeness, Approximation Algorithms, PRAM Algorithms.

Text and References Books:

1. Coreman, Rivest, Lisserson: “Algorithm", PHI.
2. Basse, "Computer Algorithms: Introduction to Design & Analysis", Addison Wesley.
3. Horowitz & Sahni, "Fundamental of Computer Algorithm", Universities Press

Course Outcomes:

1. Understand and apply mathematical preliminaries to the analysis and design stages of different types of algorithms. (Understand, Apply)
2. Analyze worst-case time complexity of various algorithms using asymptotic methods. (Analyze)
3. Understand and apply the divide-and-conquer paradigm and synthesize divide-and-conquer algorithms on problems of Sorting, Searching, finding MST etc. (Understand, Apply)
4. Describe the greedy paradigm and explain when an algorithmic design situation calls for it. For a given problem develop the greedy algorithms. (Apply, Analyze)
5. Apply the dynamic-programming paradigm to model engineering problems using graph and write the corresponding algorithm to solve the problems. (Apply)
6. Explain the ways to analyze randomized and approximation algorithms (Apply, Analyze)

CO and PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	1	-	-	-	-	-	-	-	-	-
CO2	3	3	1	-	-	-	-	-	-	-	-	-
CO3	3	2	3	-	-	-	-	-	-	-	-	-
CO4	3	2	3	-	-	-	-	-	-	-	-	-
CO5	3	2	3	-	-	-	-	-	-	-	-	-
CO6	2	2	-	-	-	-	-	-	-	-	-	-

CO and PSO Mapping

CO/PSO	PSO1	PSO2	PSO3	PSO4
CO1	3	1	-	1
CO2	2	1	-	1
CO3	2	2	1	1
CO4	2	2	1	1
CO5	2	2	1	1
CO6	2	1	-	2

DATABASE MANAGEMENT SYSTEMS (ECA-456)

Type	L	T	P	Credits
PCC	3	1	2	5

Prerequisite:

Course Content:

Unit-1:

Introduction: An overview of database management system, database system Vs file system, Database system concepts and architecture, data models schema and instances, data independence and database language and interfaces, Data definitions language, DML, Overall Database Structure. Data Modeling using the Entity Relationship Model: ER model concepts, notation for ER diagram, mapping constraints, keys, Concepts of Super Key, candidate key, primary key, Generalization, aggregation, reduction of an ER diagrams to tables, extended ER model, relationships of higher degree.

Unit-2:

Relational data Model and Language: Relational data model concepts, integrity constraints: entity integrity, referential integrity, Keys constraints, Domain constraints, relational algebra, relational

calculus, tuple and domain calculus, Introduction to SQL: Characteristics of SQL, Advantage of SQL.SQL data types and literals, Types of SQL commands, SQL operators and their procedure, Tables, views and indexes. Queries and sub queries, Aggregate functions. Insert, update and delete operations, Joins, Unions, Intersection, Minus, Cursors in SQL.

Unit-3:

Data Base Design & Normalization: Functional dependencies, normal forms, first, second, third normal forms, BCNF, inclusion dependences, loss less join decompositions, normalization using FD, MVD, and JDs, alternative approaches to database design.

Unit-4:

Transaction Processing Concepts: Transaction system, Testing of serializability, Serializability of schedules, conflict & view serializable schedule, recoverability, Recovery from transaction failures, log based recovery, checkpoints, deadlock handling.

Unit-5:

Concurrency Control Techniques: Concurrency control, locking Techniques for concurrency control, Time stamping protocols for concurrency control, validation based protocol, multiple granularity, Multi version schemes, Recovery with concurrent transaction.

Lab Work:

1. Write the queries for Data Definition and Data Manipulation language.
2. Write SQL queries using Logical operators (=, <, >, etc.).
3. Write SQL queries using SQL operators (Between…. AND, IN(List), Like, ISNULL and with negating expressions).
4. Write SQL query using character, number, date and group functions.
5. Write SQL queries for Relational Algebra (UNION, INTERSECT, and MINUS, etc.).
6. Write SQL queries for extracting data from more than one table (Equi-Join, Non-Equi-Join, Outer Join)
7. Write SQL queries for sub queries, nested queries.
8. Write programs by the use of PL/SQL.
9. Concepts for ROLL BACK, COMMIT & CHECK POINTS.
10. Create VIEWS, CURSORS, and TRIGGRS & write ASSERTIONS.
11. Create FORMS and REPORTS.

*Students are advised to use **Developer 2000/Oracle-9i** version or other latest version for above listed experiments. However depending upon the availability of software's, students may use **Power Builder /SQL SERVER**. Students may also work on a Mini Project to understand the important concepts of Database.

Text and References Books:

1. Date C J, "An Introduction to Database System", Addision Wesley
2. Korth, Silbertz, Sudarshan, "Database Concepts", McGraw Hill
3. Elmasri, Navathe, "Fundamentals of Database Systems", Addision Wesley
4. Leon & Leon, "Database Management System", Vikas Publishing House.
5. Bipin C. Desai, "An introduction to Database Systems", Galgotia Publication
6. Majumdar & Bhattacharya, "Database Management System", TMH
7. Ramakrishnan, Gehrke, "Database Management System", McGraw Hill
8. Kroenke, "Database Processing: Fundamentals, Design and Implementation", Pearson Education.

9. Maheshwari Jain, “DBMS: Complete Practical Approach”, Firewall Media, New Delhi.

Course Outcomes

1. Understand and Develop Entity Relationship (ER) and Relational Models for a given application. (Understand, Apply)
2. Develop and manipulate relational database using Structured Query Language and relational languages. (Apply)
3. Develop a normalized database for a given application by incorporating various constraints like integrity and value constraints. (Apply)
4. Understand and apply transaction processing concepts and convert schedules to serializable schedules. (Understand, Apply)
5. Illustrate different concurrency control mechanisms to preserve data consistency in a multi-user environment. (Apply)

CO and PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	2	-	-	-	-	-	-	-	-	-
CO2	3	2	2	-	-	-	-	-	-	-	-	-
CO3	3	3	2	-	-	-	-	-	-	-	-	-
CO4	3	3	2	-	-	-	-	-	-	-	-	-
CO5	3	3	2	-	-	-	-	-	-	-	-	-

CO and PSO Mapping

CO/PSO	PSO1	PSO2	PSO3	PSO4
CO1	3	2	1	2
CO2	2	2	-	1
CO3	2	2	-	1
CO4	2	3	-	1
CO5	2	3	-	1

DATA STRUCTURE USING C (ECA-458)

Type	L	T	P	Credits
PCC	3	1	2	5

Prerequisite: Computer Concepts & Programming in 'C' (ECA-451)

Course Content:

Unit -1:

Introduction: Basic Terminology, Elementary Data Organization, Structure operations, Algorithm Complexity and Time-Space trade-off.

Arrays: Array Definition, Representation and Analysis, Single and Multidimensional Arrays, address calculation, application of arrays, Character String in C, Character string operation, Array as Parameters, Ordered List, Sparse Matrices and Vectors.

Stacks: Array Representation and Implementation of stack, Operations on Stacks: Push & Pop, Array Representation of Stack, Linked Representation of Stack, Operations associated with Stacks, Applications of stack: Conversion of Infix to Prefix and Postfix Expressions, Evaluation of postfix expression using stack, Applications of recursion in problems like 'Tower of Hanoi'.

Unit-2:

Queues: Array and linked representation and implementation of queues, Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, D-queues and Priority Queues**.**

Linked list: Representation and Implementation of Singly Linked Lists, Two-way Header List, Traversing and Searching of Linked List, Overflow and Underflow, Insertion and deletion to/from Linked Lists, Insertion and deletion Algorithms, Doubly linked list, Linked List in Array, Polynomial representation and addition, Generalized linked list, Garbage Collection and Compaction.

Unit-3:

Trees: Basic terminology, Binary Trees, Binary tree representation, algebraic Expressions, Complete Binary Tree, Extended Binary Trees, Array and Linked Representation of Binary trees, Traversing Binary trees, Threaded Binary trees, Traversing Threaded Binary trees, Huffman algorithm**.**

Searching and Hashing: Sequential search, binary search, comparison and analysis, Hash Table, Hash Functions, Collision Resolution Strategies, Hash Table Implementation.

Unit-4:

Sorting: Insertion Sort, Bubble Sort, Quick Sort, Two Way Merge Sort, and Heap Sort, Sorting on Different Keys, Practical consideration for Internal Sorting**.**

Binary Search Trees: Binary Search Tree (BST), Insertion and Deletion in BST, Complexity of Search Algorithm, Path Length, AVL Trees, B-trees.

Unit-5:

Graphs: Terminology & Representations, Graphs & Multi-graphs, Directed Graphs, Sequential Representations of Graphs, Adjacency Matrices, Traversal, Connected Component and Spanning Trees, Minimum Cost Spanning Trees.

File Structures: Physical Storage Media File Organization, Organization of records into Blocks, Sequential Files, Indexing and Hashing, Primary indices, Secondary indices, B+ Tree index Files, B Tree index Files, Indexing and Hashing Comparisons.

Text and Reference Books:

1. Horowitz and Sahani, "Fundamentals of data Structures", Galgotia Publication Pvt. Ltd., New Delhi.
2. R. Kruse et. al, "Data Structures and Program Design in C", Pearson Education Asia, Delhi-2002
3. A. M. Tenenbaum, "Data Structures using C & C++", Prentice-Hall of India Pvt. Ltd., New Delhi.
4. K Loudon, "Mastering Algorithms with C", Shroff Publisher & Distributors Pvt. Ltd.
5. Bruno R Preiss, "Data Structures and Algorithms with Object Oriented Design Pattern in C++", Jhon Wiley & Sons, Inc.
6. Adam Drozdek, "Data Structures and Algorithms in C++", Thomson Asia Pvt. Ltd.(Singapore)

Lab Work:

Write Program in C or C++ for the following

1. Array implementation of Stack, Queue, Circular Queue, List.
2. Implementation of Stack, Queue, Circular Queue, List using Dynamic memory Allocation.
3. Implementation of Tree Structures, Binary Tree, Tree Traversal, Binary Search Tree, Insertion and Deletion in BST.
4. Implementation of Searching and Sorting Algorithms.
5. Graph Implementation, BFS, DFS, Min. cost spanning tree, shortest path algorithm.

Course Outcomes:

1. Analyze the algorithms to determine the time and computation complexity and justify the correctness. (Analyze)
2. Implement Arrays, Stacks, Queues and linked list based problems and analyze the algorithm to determine the time complexity. (Apply, Analyze)
3. Implement search and traversal algorithms on Trees and Graphs and determine the time complexity. (Apply, Analyze)
4. Algorithms for Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort and compare their performance in term of space and time complexity. (Apply, Analyze, Evaluate)
5. Understand file structures and file handling. (Understand)

CO and PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	1	1	-	-	-	-	-	-	-	-	-
CO2	2	1	-	-	-	-	-	-	-	-	-	-
CO3	3	3	2	-	-	-	-	-	-	-	-	-
CO4	3	3	2	-	-	-	-	-	-	-	-	-
CO5	2	-	-	-	-	-	-	-	-	-	-	-

CO and PSO Mapping

CO/PSO	PSO1	PSO2	PSO3	PSO4
CO1	3	2	1	1
CO2	3	2	1	1
CO3	3	2	1	1
CO4	3	2	1	1
CO5	2	1	-	-

COMPUTER GRAPHICS & ANIMATION (ECA-460)

Type	L	T	P	Credits
PCC	3	0	2	4

Prerequisite: NIL

Course Content:

Unit-1:
Line generation: Points and Lines, Planes, Pixels and Frame buffers, vector and character generation. Graphics Primitives: Display devices, Primitive devices, Display File Structure, Display control text, Line-drawing Algorithms**:** DDA Algorithm Bresenham's line Algorithm, Circle-generating Algorithm**:** Midpoint Circle of Algorithm, Polygon Filling Algorithm.

Unit-2:
2-D Viewing and Clipping: Point Clipping, Line Clipping, Cohen-Sutherland Line Clippings, Cyrus-Beck Line Clipping Algorithm, Polygon Clipping: Sutherland Hodgman Algorithm, Polygon: Polygon Representation, Entering polygons, Filling polygons, Segments: Segments table, Creating deleting and renaming segments, Visibility.

Unit-3:
2-D and 3-D Transformations: Basic Transformations: Translation, Rotation, Scaling, Shear, Composite Transformations: Rotations about a point, Reflection about a line, Homogeneous Coordinate Systems, 3-D Transformations, 3-D geometry primitives, Viewing Transformation, Projections: Parallel Projection, Orthographic & Oblique Projections, Perspective Projections. Interaction: Hardware input devices handling algorithms, Event handling echoing, Interactive techniques.

Unit-4:
Hidden Line and Surface: Back face removal algorithms, hidden line methods, Rendering and Illumination: Introduction to curve and Surfaces generation, Bezier, Hermite and B-spline algorithms and their comparisons.

Unit-5:
Multimedia and Animation: Basic of Animation, Types of Animation, Simulating, Accelerations, Computer Animation Tools, Multimedia Applications, Concepts of Hypertext/Hypermedia, Images, Audio and Video**,** Multimedia Tools.

Text and Reference Books:

1. Rogers, "Procedural Elements of Computer Graphics", McGraw Hill
2. Baker and Hearn, "Computer Graphics", PHI Publication.
3. Newman and Sproul, "Principle of Interactive Computer Graphics", McGraw Hill
4. Steven Harrington, "Computer Graphics", A Programming Approach, 2nd Edition
5. Rogar and Adams, "Mathematical Elements of Computer Graphics", McGraw Hill.

Lab Work:

Write Program in C or C++ for the following.

1. Implementation of line generation using slope's method, DDA and Bresenham's algorithms.
2. Implementation of circle generation using Mid-point method and Bresenham's algorithm.
3. Implementation of ellipse generation using Mid-point method.
4. Implementation of polygon filling using Flood-fill, Boundary-fill and Scan-line algorithms.
5. Implementation of 2-D transformation: Translation, Scaling, Rotation, Mirror Reflection and Shearing (write a menu driven program).
6. Implementation of Line Clipping using Cohen-Sutherland algorithm and Bisection Method.
7. Implementation of Polygon Clipping using Sutherland-Hodgman algorithm.
8. Implementation of 3-D geometric transformations: Translation, Scaling and rotation.
9. Implementation of Curve generation using Interpolation methods.
10. Implementation of Curve generation using B-spline and Bezier curves.
11. Implementation of any one of Back face removal algorithms: Depth-Buffer algorithm, Painter's algorithm, Warnock's algorithm, Scan-line algorithm.

Course Outcomes:

1. Understand and use various mathematical concepts and supporting composite 2-D & 3-D graphics transformations for hidden surface detection/ removal and various graphical algorithms. (Understand, Apply)
2. Design algorithms for various graphics shapes like ellipse, hyperbola, triangle etc. (Apply)
3. Use of various graphical tools and software in 3D Graphics API (e.g. OpenGL or DirectX). (Apply)
4. Understand and apply geometrical transformation and computer graphics in multidisciplinary field of engineering. (Apply)
5. Understand the hardware system architecture for computer graphics - graphics pipeline, frame buffers, and graphic accelerators/co-processors. (Understand)
6. Analyze and implement interactive graphics applications using programming language and graphics application programming interfaces. (Apply, Analyze)

CO and PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	3	-	-	-	-	-	-	-	-	-
CO2	2	1	3	-	-	-	-	-	-	-	-	-
CO3	2	2	3	-	3	-	-	-	-	-	-	-
CO4	2	1	1	2	-	2	-	-	-	-	-	2
CO5	2	1	-	-	-	-	-	-	-	-	-	-
CO6	2	2	3	2	-	-	-	-	2	-	-	-

CO and PSO Mapping

CO/PSO	PSO1	PSO2	PSO3	PSO4
CO1	2	2	2	-
CO2	2	3	1	-
CO3	2	3	3	-
CO4	2	3	3	2
CO5	2	1	-	-
CO6	2	3	3	1

OPERATIONS RESEARCH (BMA-452)

Type	L	T	P	Credits
BSC	3	0	0	3

Prerequisite: NIL

Course Content:

Unit-1: Linear Programming Problems (LPP)

OR model, Formulation of LPP. model, Graphical LPP solution and sensitivity analysis, simplex method, M-method, Two-phase method, Special cases in simplex method application, Duality theory, Dual simplex method, Revised simplex method, Degeneracy, Sensitivity analysis, Various industrial application of LP.

Unit-2: Transportation Models, Assignment Models and Integer Programming

Formulation and Optimal solution of transportation models, Assignment models, Trans shipment models, Degeneracy in TP model, Industrial application, Formulation and Solution of integer linear

programming problems; Cutting-plane algorithm, Branch and Bound algorithm, 0-1 ILPP, applications, Knapsack problem, facility-location problem.

Unit-3: Sequencing and Scheduling Model

Sequencing problems- Travelling salesman problem, Machine-scheduling problem (Job shop), Network based planning models, Objectives of CPM and PERT, Characteristics of CPM/PERT projects, Network diagram, Terminology, Critical path, Project duration, PERT Network, Activity time, Probabilities of project completion, Optimal crashing of project activities.

Unit-4: Replacement and Inventory models

Replacement Problems: Optimal age of equipment replacement, capital equipment discounting cost, Replacement of items that fail, Individual and group replacement policies.

Inventory Models: Deterministic inventory models, Classic EOQ model, EOQ with price breaks, Multi-term, stochastic inventory models under probabilistic demand and lead times.

Unit-5: Dynamic Programming and Genetic Algorithms

Dynamic programming: Bellman's principle of optimality, computations in DP, Forward and Backward recursions, Dynamic Programming formulations, Investment problem, General allocation problem, Storage coach problem, Production scheduling.

Genetic Algorithms: Working principles, similarities and differences between Gas and Traditional methods, Gas for constrained optimization, Applications of Gas to solve simple problems.

Text and Reference Books:

1. S. S. Rao, "Optimization: Theory and Applications" Willey Eastern Limited.
2. H.A. Taha, "Operations Research- AN Introduction", Macmillan.
3. Hiller, F. S., G.J. Lieberman, "Introduction to Operations Research", Hoiden-Day.
4. Kalyanmoy Deb, "Optimization for Engineering Design: Algorithms & Examples "Prentice-Hall of India.
2. B. E. Gillet, Introduction Operations Research- A Computer Oriented Algorithmic Approach, McGraw Hill 1989.

Course Outcomes:

1. Understand and solve Linear Programming Problems.
2. Formulate and solve Transportation Models, Assignment Models and Integer Linear Programming Problems.
3. Formulate and solve Sequencing and Scheduling Models.
4. Formulate and solve Replacement and Inventory Models.
5. Learn and use Dynamic Programming and Genetic Algorithms.

CO and PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	2	3	2	2	-	1	2	2	3
CO2	3	3	3	2	3	2	2	-	1	2	2	3
CO3	3	3	3	3	2	2	2	-	1	2	2	3
CO4	3	3	3	3	2	2	2	-	1	2	2	3

(ECA-551) COMPUTER NETWORKS 5 (3-1-2)

Course Content:

Unit-1:

Introduction Concepts: Goals and Applications of Networks, Network structure and architecture, The OSI reference model, services, Network Topology Design – Delay Analysis, Back Bone Design, Local Access Network Design. Physical Layer Transmission Media, Switching methods, ISDN, Terminal Handling.

Unit-2:

Medium Access sub layer: Medium Access sub layer - Channel Allocations, LAN protocols - ALOHA protocols - Overview of IEEE standards - FDDI. Data Link Layer - Elementary Data Link Protocols, Sliding Window protocols, Error Handling.

Unit-3:

Network Layer: Network Layer - Point - to Pont Networks, routing, Congestion control Internetworking -TCP / IP - IP packet, IP address, IPv6. '

Unit-4:

Transport Layer: Transport Layer - Design issues, connection management, session Layer-Design issues, remote procedure call. Presentation Layer-Design issues, Data Compression techniques, cryptography - TCP - Window Management.

Unit-5:

Application Layer: Application Layer: File Transfer, Access and Management, Electronic mail, Virtual Terminals, Other application, Example Networks - Internet and Public Networks.

Text and References Books:

1. Forouzen, "Data Communication and Networking", TMH
2. A.S. Tannenbaum, “Computer Networks”, 3rd Edition, Prentice Hall India, 1997.
3. S. Keshav, “An Engineering Approach on Computer Net working”, Addison Wesley, 1997
4. W. Stallings, “Data and Computer Communication”, Mac-millan Press, 1989

Course Outcomes:

1. Explain the functions of the different layer of the OSI Protocol. (Understand)
2. Design of wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs) based on available network devices. (Apply, Analyze)
3. Develop network programing application for a given problem related to TCP/IP protocol stack. (Apply, Analyse)
4. Understand and analyze different routing algorithms. (Understand, Analyze)
5. Understand the use of IP addressing schemes as per IPV4 and IPV6. (Understand)

(ECA-553) OBJECT ORIENTED SYSTEM MODELING 5(3-1-2)

Course Content:

Unit-1:

Object Oriented Design and Modeling: Object oriented fundamentals, Objects and Classes, ObjectOriented Design Process, importance of modeling, principles of modeling, OOAD Methods, Software Development Life Cycle, Introduction to Unified Process, Introduction to UML: UML Terminology, conceptual model of the UML, Use of UML in Unified Process.

Unit-2:

Basic Structural Modeling: Classes, Relationships, common Mechanisms, and diagrams. Class & Object Diagrams: Terms, concepts, modeling techniques for Class & Object Diagrams, Links and Associations, Link Attributes and Link Classes, Generalization and Inheritance, Aggregation and Composition, Qualified Association, Handling multiplicity in Object creation, Abstract Classes, Specifying constraints in Class Diagrams, Advanced Structural Modeling: Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages, Use Case Modeling: Use Cases and Use Case Diagrams, Use Case driven Methodology.

Unit-3:
Behavioral Modeling: Interactions and Interaction Diagrams, Use-Case Realization: Scenario, Events Trace Diagram, Collaboration Diagrams, State Chart Diagrams, Nested State Diagrams, Activity Diagrams, Advanced Behavioral Modeling Concepts, Architectural Modeling: Component, Deployment, Component diagrams and Deployment diagrams. Elementary Design Patterns, The MVC Architecture Pattern, Features of Elegant Software Design: Elegant variable, Elegant Classes, Elegant Methods, Elegant Packages, Introduction to Object Oriented Software Quality Metrics.

Unit-4:
Java Programming: Introduction to Java Programming, Operator, Data type, Variable, Arrays, Control Statements, Methods & Classes, Inheritance, Package and Interface, Exception Handling, Multithread programming, I/O, Java Applet, String handling, Networking, Event handling.

Unit-5:
Introduction to Advance Java Programming: Demonstration of concepts through example programs for AWT, Java Swing, Java Beans, Java Servlets, JSP, Modern Object Technologies and Web Services.

Lab Work:
1. Write a program in Java, to implements the Stack data Structure.
2. Write a program in Java to implement a simple Bank Account.
3. Write a program in Java showing the action from three threads using a suitable example
4. Write a program of threads in Java showing inter leaving of actions from two threads: t1 & t2 synchronizing on a shared object. Let t1 print message Ping and t2 prints message Pong. Take as command line arguments the following inputs to the program:
Sleep interval for thread t1
Sleep interval for thread t2
Messages per cycle
Number of Cycles
5. Write a program in Java which converts a text file into all capital letters.6. Write a program to create a sequential file that could store details about five products. Details include product code, cost, no. Of items available and number of items available and are provided through keyboard.
7. Create a Person class with private instance variables for Person‟s name and birth date. Add appropriate accessor methods to access the variables. Then create a subclass CollegeGraduate with private instance variables for the student‟s GPA and year of graduation and appropriate accessors for these variables. Don‟t forget to include appropriate constructors for your classes. Then create a class with a main() method that manages your classes.
8. Develop an applet that receives three numeric values from the user and displays the largest of the three on the screen. Write a HTML page that embeds this applet.
9. Write an applet which draws a human face with ovals and arcs.
10. Write servlets that accepts user preferences (color, hobby etc.) from user, saves it as cookie on user machine and reads the cookie from the user machine.
11. Write an AWT application with checkbox such that all cable TV channels will be displayed from the selected category.

12. Create a simple Swing based applet that displays two buttons. Each time a button is clicked, a message is displayed that states which button was clicked.
13. Create JSP code that uses a persistant cookie (i.e. a cookie with an expiration date in the future) to keep track of how many times the client computer has visited the page. Use setMaxAge method to remain on the client‟s computer for one month. Display the number of page hits (i.e. cookie‟s value) every time the page loads.
14. Write JSP program that asks user his favourite color as request parameter and sets it as the background color of the page or sets the background color white if the parameter value is null.
15. Write a program in Java to show the mouse click event. The program should change the background colour of window randomly at each mouse click.

Text and Reference Books:

1. Balagurusamy E, “Programming in JAVA”, TMH
2. Herbert Schildt, “The Complete Reference JAVA”, TMH
3. Bruce Eckel, “Thinking in Java”, Prentice Hall PTR.
4. Grady Booch, James Rumbaugh, Ivar Jacobson: The Unified Modeling Language User Guide, Pearson Education.
5. Mark Priestley: Practical Object-Oriented Design with UML, TATA Mc-GrawHill.
6. Meilir Page-Jones: Fundamentals of Object Oriented Design in UML, Pearson Education.
7. Pascal Roques: Modeling Software Systems Using UML2, WILEY-Dreamtech India Pvt. Ltd.
8. Appling UML and Patterns: An introduction to Object – Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.
9. Atul Kahate: Object Oriented Analysis & Design, The McGraw-Hill Companies.

Course Outcomes:

1. Analyse information systems in real-world settings and use an object-oriented method for analysis and design. (Analyse)
2. Understand features of object-oriented design such as encapsulation, polymorphism, inheritance, and UML. (Understand)
3. Understand and prepare different types of UML diagrams like use case diagrams, interaction diagrams, nested state diagrams, state chart diagrams, activity diagram etc. (Understand, Apply)
4. Understand and appreciate the use of Design Patterns in the Software Development. (Understand, Apply)
5. Understand the core and advance Java Programming features and apply them in complex problem solving. (Understand, Apply)

(ECA-555) SOFTWARE PROJECT MANAGEMENT 4 (3-0-2*)

Course Content:

Unit-1:

Introduction and Software Project Planning: Fundamentals of Software Project Management (SPM), Need Identification, Vision and Scope document, Project Management Cycle, SPM Objectives, Management Spectrum, SPM Framework, Software Project Planning, Planning Objectives, Project Plan, Types of project plan, Structure of a Software Project Management Plan.

Unit-2:

Project Organization and Scheduling: Project Elements, Work Breakdown Structure (WBS), Types of WBS, Functions, Activities and Tasks, Project Life Cycle and Product Life Cycle, Ways to Organize Personnel, Project schedule, Scheduling Objectives, Building the project schedule, Scheduling terminology and techniques, Network Diagrams: PERT, CPM, Bar Charts: Milestone Charts, Gantt Charts.

Unit-3:
Project Monitoring and Control: Dimensions of Project Monitoring & Control, Earned Value Analysis, Earned Value Indicators: Budgeted Cost for Work Scheduled (BCWS), Cost Variance (CV), Schedule Variance (SV), Cost Performance Index (CPI), Schedule Performance Index (SPI), Interpretation of Earned Value Indicators.

Unit-4:
Software Quality Assurance: Concept of Software Quality, Software Quality Attributes, Software Quality Metrics and Indicators, Software Quality Assurance (SQA), SQA Activities, SQA Plans, Software Quality Frameworks, ISO 9000 Models, SEI Capability Maturity Model (CMM), Software verification and validation, Formal SQA Approaches: Proof of correctness, Statistical quality assurance, Clean room process.

Unit-5:
Software project estimation, Estimation methods, Estimation models, Decision process. Risk Management: Risks and risk types, Risk Breakdown Structure (RBS), Risk Management Process: Risk identification, Risk analysis, Risk planning, Risk monitoring, Cost Benefit Analysis, Software Project Management Tools: CASE Tools, Planning and Scheduling Tools like MS-Project.

Text and References Books:
1. Software Project Management, M. Cotterell, Tata McGraw-Hill Publication.
2. Information Technology Project Management, Kathy Schwalbe, Vikas Pub. House.
3. Software Project Management, S. A. Kelkar, PHI Publication

Course Outcomes:
1. Understand various concepts of Software Project Planning and Management. (Understand)
2. Understand various techniques of Human Resource Organization and Develop schedule of software projects using PERT/CPM. (Understand, Apply)
3. Understand cost benefit analysis, risk management and techniques of monitoring & control of software projects. (Understand)
4. Use concepts of software quality assurance in the development of software projects. (Apply)
5. Assess the project to develop the scope of work, provide accurate size, cost, time and effort estimates for software projects. (Apply, Analyze)

(ECA-557) ARTIFICIAL INTELLIGENCE 4(3-1-0)

Course Content:
Unit-1:
Introduction to Artificial Intelligence, Brief history, Various approaches to AI, Areas of application, Simulation of sophisticated & Intelligent Behavior in different area, Problem solving in games, natural language processing, automated reasoning, and visual perception, Knowledge and its role in AI, Heuristic algorithm versus solution guaranteed algorithms, Introduction to soft computing.

Unit-2:
Representing problems in state space, Informed versus uninformed search, Production System Model, Evaluation of the Production System, Depth First Search and Breadth First Search, Heuristics, Heuristic Search Techniques: Hill Climbing, Best First search, A* Algorithm, Branch and Bound, Cryptarithmatic Problem, Means End Analysis, AO* Algorithm, Game Playing: MINMAX Search, Alpha-Beta Pruning, Heuristic Estimation.

Unit-3:

Knowledge Representation and Reasoning : Propositional Logic, First Order Predicate Logic, Graphs, Associative Network, Semantic Networks, Conceptual Dependencies, Frames, Scripts, Horn Clauses, Introductory Examples from PROLOG, Case Grammar Theory, Production Rules Knowledge Base, The Interface System, Forward & Backward Deduction, Inference System in Propositional and Predicate Logic, Reasoning under Uncertainty.

Unit-4:
Understanding Natural Languages, Various Approaches of NLP, Parsing techniques, Context free and transformational grammars, Transition nets, Augmented transition nets, Fillmore's grammars, Grammar free analyzers, Sentence generation, and translation, Introduction to Pattern Recognition, Structured Description, Symbolic Description, Machine Perception, Object Identification, Speech Recognition.

Unit-5:
Expert Systems: Architecture of Expert System, Representing and using domain knowledge, Expert System Shell, Explanation System, Knowledge Acquisition System, Case study of Existing Expert Systems like DENDRAL, MYCIN, Development of a small Expert System using programming Languages and tools like LISP, PROLOG, JESS.

Text and References Books:
1. N. J. Nilsson, “Artificial Intelligence: A New Synthesis”, Elsevier Publications.
2. Charnick, “Introduction to A.I.”, Addison Wesley.
3. Rich & Knight, “Artificial Intelligence”, McGraw-Hill Publication.
4. Winston, “LISP”, Addison Wesley
5. Marcellous, “Expert System Programming”, PHI
6. Elamie, “Artificial Intelligence”, Academic Press
7. Lioyed, “Foundation of Logic Processing”, Springer Verlag
8. D. W. Patterson, “Introduction to Artificial Intelligence and Expert Systems”, PHI.

Course Outcomes:
1. Understand different types of AI agents (Understand).
2. Understand and apply various AI search algorithms (uninformed, informed, heuristic, constraint satisfaction, genetic algorithms) (Understand, Apply).
3. Understand the fundamentals of knowledge representation, reasoning, and machine learning techniques and apply them to real world problems. (Understand, Apply)
4. Know how to build simple knowledge based systems using languages like LISP, Prolog, and AI tools like JESS. (Apply)
5. Carry out independent (or in a small group) research and communicate it effectively in a seminar. (Apply, Analyze)

Programme Elective-I 4 (3-1-0)
(Any one of the following five courses)

Theory of Automata & Formal Languages (ECA-561)

Course Content:
Unit-1:
Defining Languages and Grammars, Chomsky hierarchy, Kleene closures, Regular Expressions, Finite Automata (FA), Transition Graph, Generalised Transition Graph.

Unit-2:

Nondeterministic finite Automata (NFA), Deterministic finite Automata (DFA), Construction of DFA from NFA and optimization, Partitioning Algorithm, Equivalence of DFA and NFA and their optimization, FA with output: Moore machine, Mealy machine and their Equivalence, Applications and Limitation of FA.

Unit-3:
Arden Theorem, Pumping Lemma for regular expressions, Myhill-Nerode theorem, Context free grammar: Ambiguous Grammars and Simplification, Normal forms for CFGs, Pumping lemma for CFLs, Decidability of CFGs, Ambiguous to Unambiguous CFG.

Unit-4:
Push Down Automata (PDA): Description and definition, Working of PDA, Acceptance of a string by PDA, PDA and CFG Equivalence, Deterministic and non-deterministic PDA, Introduction to auxiliary PDA and Two Stack PDA.

Unit-5:
Turing machines (TM): Basic model, definition and representation, Language acceptance by TM, TM and Type – 0 Grammar, Integer function computation by TM, Halting problem of TM, Modifications in TM, Universal TM, Properties of recursive and recursively enumerable languages, decision problem, Un-decidability of Post Correspondence Problem, Church‟s Thesis, Recursive function theory, Godel Numbering.

Text and References Books:
1. Hopcroft, Ullman, “Introduction to Automata Theory, Language and Computation”, Nerosa Publishing House
2. K.L.P. Mishra and N. Chandrasekaran, “Theory of Computer Science (Automata, Languages and Computation)”, PHI.
3. Martin J. C., “Introduction to Languages and Theory of Computations”, TMH
4. Papadimitrou, C. and Lewis, C.L., “Elements of theory of Computations”, PHI
5. Cohen D. I. A., “Introduction to Computer theory”, John Wiley & Sons
6. Kumar Rajendra, “Theory of Automata (Languages and Computation)”, PPM

Course Outcomes:
1. Describe the capabilities and limitations of the abstract machines including finite automata, pushdown automata, and Turing machines and their associated languages. (Understand)
2. Construct finite automata, pushdown automata, Turing machines for the given grammar and vice versa. (Apply)
3. Show that a language is not regular / not context-free using pumping lemma. (Apply)
4. Outline the characteristics of P, NP and NP Complete problems in the context of Turing machines. (Understand)

SIMULATION AND MODELLING (ECA-563)

Course Content:
Unit-1
System definition and components, stochastic activities, continuous and discrete Systems, System modeling, types of models, static and dynamic physical models, Static and dynamic mathematical models, Full corporate model, types of system study.

Unit-2

System simulation, Why to simulate and when to simulate, Basic nature of simulation, technique of simulation, comparison of simulation and analytical methods, types of system simulation, real time simulation, hybrid simulation, simulation of pure-pursuit problem single-server queuing system and an inventory problem, Monte Carlo simulation, Distributed Lag models, Cobweb model.

Unit-3
Simulation of continuous systems, analog vs. digital simulation, simulation of water reservoir system, simulation of a servo system, simulation of an autopilot Discrete system Simulation, Fixed time-step vs. event-to-event model, generation of random numbers, Test for randomness, Generalization of non-uniformly distributed random numbers, Monte-Carlo computation vs. stochastic simulation.

Unit-4
System dynamics, exponential growth models, exponential decay models, modified exponential growth models, logistic curves, generalization of growth models, System Dynamics diagrams, Feedback in Socio-Economic systems, world model.

Unit-5
Simulation of PERT networks, Critical path computation, uncertaintities in Activity duration, Resource allocation and consideration. Simulation software, Simulation languages, continuous and discrete simulation languages, Expression based languages, object-oriented simulation, generalpurpose vs. application-oriented simulation packages, CSMP-III, MODSIM-III.

Text and Reference Books:
1. Geoftrey Gordon,“ System Simulation”, PHI
2. Narsingh Deo, “ System Simulation with digital computer”, PHI
3. Averill M. Law, W. David Kelton, “Simulation Modeling and Analysis”, TMH

Course Outcomes:
1. Understand the modeling process and various types of modeling techniques (Understand)
2. Understand various advantages of modeling & simulation over the actual experimentation analytical and experimentation techniques. (Understand)
3. Study various types of simulation techniques and understand their relative advantages and disadvantages. (Understand)
4. Understand and analyze the system dynamics with specific reference to various types of growth models. (Understand, Analyze)
5. Understand and use optimization techniques like PERT/CPM and various modeling & simulation tools. (Understand, Apply)

INFORMATION SECURITY & CYBER LAWS (ECA-565)

Course Content:
Unit-1:
Introduction to information systems, Types of information systems, Development of Information systems, Introduction to information security, Need for Information security, Threats of Information Systems, Information Assurance, Cyber Security and Security Risk Analysis.

Unit-2
Application security (Database, E-mail and Internet), Data Security Considerations - Backups, Archival Storage and Disposal of Data, Security Technology - Firewall and VPNs, Intrusion Detection, Access Control, Security Threats - Viruses, Worms, Trojan Horse, Bombs, Trapdoors, Spoofs, E-mail viruses, Macro viruses, Malicious Software, Network and Denial of Services Attack, Security Threats to E-

Commerce – Electronic Payment System, e-Cash, Credit/Debit Cards, Digital Signature, public Key Cryptography.

Unit-3
Developing Secure Information Systems, Application Development Security, Information Security Governance & Risk Management, Security Architecture & Design, Security Issues in Hardware, Data Storage & Downloadable devices, Physical Security of IT Assets, Access Control, CCTV and intrusion Detection Systems, Backup Security Measures.

Unit-4
Security Policies, why Policies should be developed, WWW Policies, Email Security Policies, Policy Review Process- Corporate policies- Sample Security Policies, Publishing and Notification requirement of the Policies.

Unit-5
Information Security Standards- ISO, IT Act, Copyright Act, Patent Law, IPR. Cyber Laws in India: IT Act 2000 Provisions, Intellectual Property Law: Copy Right Law, Software License, Semiconductor Law and Pattern Law.

Text and Reference Books:
1. Charles, P., and Shari Lawrence Pfleeger, “Analyzing Computer Security”. Pearson Education India.
2. V.K. Pachghare, “Cryptography and information security”, PHI Learning Pvt. Ltd., Delhi India.
3. Dr Surya Prakash Tripathi, Ritendra Goyal, and Praveen Kumar Shukla, "Introduction to Information Security and Cyber Law", Willey Dreamtech Press.
4. Schou, Shoemaker, “Information Assurance for the Enterprise”, Tata McGraw Hill.
5. Chander Harish, “Cyber Laws and their Protection”, PHI Learning Private Limited, Delhi, India.

Course Outcomes:
1. Understand information, information systems, information security, Cyber Security and Security Risk Analysis. (Understand)
2. Understand and apply application security, data security, security technology, security threats from malicious software. (Understand, Apply)
3. Understand the concepts of security threats to e-commerce applications such as electronic payment system, e-Cash, Credit/Debit Cards etc. (Understand)
4. Understand and apply Information Security Governance & Risk Management, Security of IT Assets and Intrusion Detection Systems. (Understand, Apply)
5. Understand various types of Security Policies, Cyber Ethics, IT Act, IPR and Cyber Laws in India. (Understand)

E-COMMERCE (ECA-567)

Course Content:
Unit -1
Introduction: Electronic Commerce - Technology and Prospects, Definition of E-Commerce, Economic potential of electronic commerce, Incentives for engaging in electronic commerce, forces behind E-Commerce, Advantages and Disadvantages, Architectural framework, Impact of Ecommerce on business. Network Infrastructure for E-Commerce: Internet and Intranet based E-commerce issues, problems and prospects, Network Infrastructure, Network Access Equipment, Broadband telecommunication (ATM, ISDN, FRAME RELAY).

Unit-2
Mobile Commerce: Introduction, Wireless Application Protocol, WAP technology, Mobile Information device, Mobile Computing Applications.

Unit-3
Web Security: Security Issues on web, Importance of Firewall, components of Firewall, Transaction security, Emerging client server, Security Threats, Network Security, Factors to consider in Firewall design, Limitation of Firewalls.

Unit-4
Encryption: Encryption techniques, Symmetric Encryption- Keys and data encryption standard, Triple encryption, Asymmetric encryption- Secret key encryption, public and private pair key encryption, Digital Signatures, Virtual Private Network.

Unit -5
Electronic Payments: Overview, The SET protocol, Payment Gateway, certificate, digital Tokens, Smart card, credit card, magnetic strip card, E-Checks, Credit/Debit card based EPS, online Banking EDI Application in business, E- Commerce Law, Forms of Agreement, Govt. policies and Agenda.

Text and Reference Books:
1. Ravi Kalakota, Andrew Winston, "Frontiers of Electronic Commerce", Addison Wesley.
2. Bajaj and Nag, "E-Commerce the cutting edge of Business", TMH
3. P. Loshin, John Vacca, "Electronic commerce", Firewall Media, New Delhi

Course Outcomes:
1. Understand the E-commerce, its advantages & disadvantages and infrastructure requirements. (Understand)
2. Understand various concepts of mobile commerce (Understand)
3. Understand various issues related to web security and their limitations. (Understand)
4. Understand concepts of various encryption techniques, digital signatures and VPN (Understand)
5. Understand details of various electronic payments systems. (Understand)

INTERNET OF THINGS (ECA-569)

Course Content:
Unit-1: Introduction
What is the Internet of Things? : History of IoT, About IoT, Overview and Motivations, Examples of Applications, Internet of Things Definitions and Frameworks: IoT Definitions, IoT Architecture, General Observations, ITU-T Views, Working Definition, IoT Frameworks, Basic Nodal Capabilities

Unit-2: Fundamentals of IoT Mechanisms and Key Technologies Identification of IoT Objects and Services, Structural Aspects of the IoT, Environment Characteristics, Traffic Characteristics, Scalability, Interoperability, Security and Privacy, Open Architecture, Key IoT Technologies, Device Intelligence, Communication Capabilities, Mobility Support, Device Power, Sensor Technology, RFID Technology, Satellite Technology

Unit-3: Radio Frequency Identification Technology
RFID: Introduction, Principle of RFID, Components of an RFID system, Issues EPC Global Architecture Framework: EPCIS & ONS, Design issues, Technological challenges, Security challenges, IP for IoT, Web of Things. Wireless Sensor Networks: History and context, WSN Architecture, the node, Connecting nodes, Networking Nodes, Securing Communication WSN specific IoT applications,

challenges: Security, QoS, Configuration, Various integration approaches, Data link layer protocols, routing protocols and infrastructure establishment.

Unit-4: Resource Management in the Internet of Things
Clustering, Software Agents, Clustering Principles in an Internet of Things, Architecture, Design Guidelines, and Software Agents for Object Representation, Data Synchronization. Identity portrayal, Identity management, various identity management models: Local, Network, Federated and global web identity, user-centric identity management, device centric identity management and hybrid-identity management, Identity and trust.

Unit-5: Internet of Things Privacy, Security and Governance
Vulnerabilities of IoT, Security requirements, Threat analysis, Use cases and misuse cases, IoT security tomography and layered attacker model, Identity establishment, Access control, Message integrity, Non-repudiation and availability, Security model for IoT. Business models for Internet of Things: Business Models and Business Model Innovation, Value Creation in the Internet of Things, Business Model Scenarios for the Internet of Things.

Text and Reference Books:
1. Daniel Minoli, "Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications", ISBN: 978-1-118-47347-4, Willy Publications
2. Bernd Scholz-Reiter, Florian Michahelles, "Architecting the Internet of Things", ISBN 978-3642-19156-5 e-ISBN 978-3-642-19157-2, Springer.
3. Parikshit N. Mahalle& Poonam N. Railkar, "Identity Management for Internet of Things", River Publishers, ISBN: 978-87-93102-90-3 (Hard Copy), 978-87-93102-91-0 (ebook).
4. Hakima Chaouchi, "The Internet of Things Connecting Objects to the Web" ISBN: 978-184821-140-7, Willy Publications.
5. Olivier Hersent, David Boswarthick, Omar Elloumi, The Internet of Things: Key Applications and Protocols, ISBN: 978-1-119-99435-0, 2nd Edition, Willy Publications.
6. Daniel Kellmereit, Daniel Obodovski, "The Silent Intelligence: The Internet of Things", Publisher: Lightning Source Inc; 1 edition (15 April 2014). ISBN-10: 0989973700, ISBN-13: 978-09-89973-70-0.

Course Outcomes:
1. Understand framework and architecture of Internet of Things. (Understand)
2. Understand key technologies in Internet of Things. (Understand)
3. Explain wireless sensor network architecture and its framework along with WSN applications. (Understand)
4. Explain resource management in the Internet of Things. (Understand)
5. Understand Security measures and design applications based on Internet of Things. (Understand, Apply)

Programme Elective-II 4 (3-1-0)
(Any one of the following five courses)

COMPILER DESIGN (ECA-552)

Course Content:
Unit-1:
Introduction to Compiler, Phases and passes, Bootstrapping, Finite automata & regular expressions and their applications to lexical analysis, Implementation of lexical analyzers, lexical-analyzer generator, LEX-compiler, The syntactic specification of Programming languages: Context free

grammars, derivation and parse trees, capabilities of CFG, Application of grammars in syntax analysis, ambiguity and BNF notation, YACC.

Unit-2:
Basic Parsing Techniques: Parsers, top down parsing, Shift reduces parsing, operator precedence parsing, predictive parsers. Automatic Construction of efficient Parsers: LR parsers, the canonical Collection of LR(0) items, constructing SLR parsing tables, constructing Canonical LR parsing tables, Constructing LALR parsing tables, using ambiguous grammars, an automatic parser generator, implementation of LR parsing tables, constructing LALR sets of items.

Unit-3:
Syntax-directed Translation: Syntax-directed Translation schemes, Implementation of Syntax directed Translators, Intermediate code, postfix notation, Parse trees & syntax trees, three address code, quadruple & triples, translation of assignment statements, Boolean expressions, statements that alter the flow of control, postfix translation, translation with a top down parser. More about translation: Array references in arithmetic expressions, procedures call, declarations, Case statements.

Unit-4:
Symbol Tables: Data structure and representing scope information, Run-Time Administration: Implementation of simple stack allocation scheme, storage allocation in block structured language. Error Detection & Recovery: Lexical Phase errors, syntactic phase errors semantic errors.

Unit-5:
Introduction to code optimization: Loop optimization, the DAG representation of basic blocks, value numbers and algebraic laws, Global Data-Flow analysis.

Text and References Books:
1. Aho, Sethi & Ullman, "Compiler Design", Addition Wesley.
2. Kenneth C. Louden, “Compiler Construction: Principles and Practice”, Thomson Brooks Publication.
3. Allen I. Holub, “Compiler Design in C”, PHI Publications.

Course Outcomes:
1. Describe the role of each phase of a compiler with its construction tools. (Understand)
2. Develop a Lexical Analyzer for recognizing tokens of a given language with an understanding of symbol table management and error handling. (Apply)
3. Construct top-down, bottom-up, operator precedence and SLR parsers with an understanding of Context Free Grammars and syntax analysis. (Apply)
4. Design and develop semantic analyzers for type-checking and intermediate code generators to translate the source program into an intermediate code. (Apply)
5. Construct code optimizers to optimize the target code generated. (Apply)

MACHINE LEARNING (ECA-554)

Course Content:
Unit-1: Introduction to Machine Learning
Why Machine learning, Examples of Machine Learning Problems, Structure of Learning, Learning versus Designing, Training versus Testing, Characteristics of Machine learning tasks, Predictive and descriptive tasks, Machine learning Models: Geometric Models, Logical Models, Probabilistic Models. Features: Feature types, Feature Construction and Transformation, Feature Selection.

Unit-2: Classification and Regression
Classification: Binary Classification- Assessing Classification performance, Class probability Estimation- Assessing class probability Estimates, Multiclass Classification. Regression: Assessing performance of Regression- Error measures, Overfitting: Catalysts for Overfitting, Case study of Polynomial Regression. Theory of Generalization: Effective number of hypothesis, Bounding the Growth function, VC Dimensions, Regularization theory.

Unit-3: Linear Models
Least Squares method, Multivariate Linear Regression, Regularized Regression, Using Least Square regression for Classification. Perceptron, Support Vector Machines, Soft Margin SVM, Obtaining probabilities from Linear classifiers, Kernel methods for non-Linearity.

Unit-4: Logic Based and Algebraic Models
Distance Based Models: Neighbours and Examples, Nearest Neighbours Classification, Distance based clustering-K means Algorithm, Hierarchical clustering, Rule Based Models: Rule learning for subgroup discovery, Association rule mining. Tree Based Models: Decision Trees, Ranking and Probability estimation Trees, Regression trees, Clustering Trees.

Unit-5: Probabilistic Models
Normal Distribution and Its Geometric Interpretations, Naïve Bayes Classifier, Discriminative learning with Maximum likelihood, Probabilistic Models with Hidden variables: EstimationMaximization Methods, Gaussian Mixtures, and Compression based Models.
Trends in Machine Learning: Model and Symbols- Bagging and Boosting, Multitask learning, Online learning and Sequence Prediction, Data Streams and Active Learning, Deep Learning, Reinforcement Learning.

Text and Reference Books:
1. Peter Flach, Machine Learning: The Art and Science of Algorithms that Make Sense of Data, Cambridge University Press, Edition 2012.
2. Hastie, Tibshirani, Friedman: Introduction to Statistical Machine Learning with Applications in R, Springer, 2nd Edition-2012.
3. C. M. Bishop, Pattern Recognition and Machine Learning, Springer 1st Edition-2013.
4. Ethem Alpaydin, Introduction to Machine Learning, PHI 2nd Edition-2013. 5. Parag Kulkarni, Reinforcement and Systematic Machine Learning for Decision Making, Wiley, IEEE Press, Edition July 2012.

Course Outcomes:
1. Understand Machine learning and Machine Learning Models. (Understand)
2. Apply various classification and regression techniques and assess their performance. (Apply)
3. Apply various clustering algorithms for the problems to be solved with machine learning. (Apply)
4. Assessment of various machine learning models. (Analyze)
5. Understand probabilistic learning models and trends in machine learning. (Understand)

Advanced Database Management Systems (ECA-556)

Course Content:
Unit-1
Query Processing, Optimization & Database Tuning: Algorithms for Executing Query Operations. Heuristics For Query Optimizations, Estimations Of Query Processing Cost, Join Strategies For Parallel Processors, Database Workloads, Tuning Decisions, DBMS Benchmarks, Clustering & Indexing, Multiple Attribute Search Keys, Query Evaluation Plans, Pipelined Evaluations, System Catalogue In

RDBMS.

Unit-2
Extended Relational Model & Object Oriented Database System: New Data Types, User Defined Abstract Data Types, Structured Types, Object Identity, Containment, Class Hierarchy, Logic Based Data Model, Data Log, Nested Relational Model and Expert Database System.

Unit-3
Distributed Database System:
Structure of Distributed Database, Data Fragmentation, Data Model, Query Processing, Semi Join, Parallel & Pipeline Join, Distributed Query Processing in R* System, Concurrency Control in Distributed Database System, Recovery in Distributed Database System, Distributed Deadlock Detection and Resolution, Commit Protocols.

Unit-4
Enhanced Data Model For Advanced Applications:
Database Operating System, Introduction to Temporal Database Concepts, Spatial and Multimedia Databases, Data Mining, Active Database System, Deductive Databases, Database Machines, Web Databases, Advanced Transaction Models, Issues in Real Time Database Design.

Unit-5
Introduction to Expert Database and Fuzzy Database System: Expert Databases: Use of Rules of Deduction in Databases, Recursive Rules. Fuzzy Databases: Fuzzy Set & Fuzzy Logic, Use of Fuzzy Techniques to Define Inexact and Incomplete Databases.

Text and Reference Books:
1. Majumdar & Bhattacharya, “Database Management System”, TMH.
2. Korth, Silbertz, Sudarshan, “Database Concepts”, McGraw Hill.
3. Elmasri, Navathe, “Fundamentals of Database Systems”, Addison Wesley.
4. Date C J,” An Introduction to Database System”, Addison Wesley.
5. Ramakrishnan Gehrke, “Database Management System”, McGraw Hill.
6. Bernstein, Hadzilacous, Goodman, “Concurrency Control & Recovery”, Addison Wesley.
7. Ceri & Palgatti, “Distributed Databases”, McGraw Hill.

Course Outcomes:
1. Understand the concepts and algorithms for Query Processing, Optimization & Database Tuning. (Understand)
2. Understand the Extended Relational Model & Object Oriented Database System. (Understand)
3. Explain the Structure of Distributed Database, Query Processing, Concurrency Control, Prevention and recovery from deadlock. (Understand)
4. Understand and analyze Enhanced Data Model for Advanced Applications including Issues in Real Time Database Design. (Understand, Analyze)
5. Understand the concept of Fuzzy logic and development of Fuzzy Database Systems. (Understand)

ERP Systems (ECA-558)

Course Content:
Unit-I
Enterprise wide information system, Custom built and packaged approaches, Needs and Evolution of ERP Systems, Common myths and evolving realities, ERP and Related Technologies, Business Process Reengineering and Information Technology, Supply Chain Management, Relevance to Data Warehousing, Data Mining and OLAP, ERP Drivers, Decision support system.

Unit-II
ERP Domain, ERP Benefits classification, Present global and Indian market scenario, milestones and pitfalls, Forecast, Market players and profiles, Evaluation criterion for ERP product, ERP Life Cycle: Adoption decision, Acquisition, Implementation, Use & Maintenance, Evolution and Retirement phases, ERP Modules.

Unit- III
Framework for evaluating ERP acquisition, Analytical Hierarchy Processes (AHP), Applications of AHP in evaluating ERP, Selection of Weights, Role of consultants, vendors and users in ERP implementation; Implementation vendors evaluation criterion, ERP Implementation approaches and methodology, ERP implementation strategies, ERP Customization, ERP-A manufacturing Perspective.

Unit- IV
Critical success and failure factors for implementation, Model for improving ERP effectiveness, ROI ofERP implementation, Hidden costs, ERP success inhibitors and accelerators, Management concern for ERP success, Strategic Grid: Useful guidelines for ERP Implementations.

Unit- V
Technologies in ERP Systems and Extended ERP, Case Studies Development and Analysis of ERP Implementations in focusing the various issues discussed in above units through Soft System approaches or qualitative Analysis tools, Learning and Emerging Issues, ERP and E-Commerce.

Text and Reference Books:
1. Lexis Leon, “Enterprise Resource Planning”, TMH
2. Brady, Manu, Wegner, “ Enterprise Resource Planning”, TMH

Python Programming (ECA-560)

Course Content:
Unit 1: Introduction: The Programming Cycle for Python, Python IDE, Interacting with Python Programs, Elements of Python, Type Conversion. Basics: Expressions, Assignment Statement, Arithmetic Operators, Operator Precedence, Boolean Expression.

Unit 2: Conditionals and Loops: Conditional statement in Python: if-else statement, its working and execution, Nested-if statement and Elif statement in Python, Expression Evaluation & Float Representation, Loops: Purpose and working of loops, while loop including its working, For Loop, Nested Loops, Break and Continue.

Unit 3: Strings and Functions: Strings: Length of the string, Concatenation and Repeat operations, Indexing and Slicing of Strings. Python Data Structure: Tuples, Unpacking Sequences, Lists, Mutable Sequences, List Comprehension, Sets, Dictionaries, Functions: Parts of a Function, Execution of a Function, Keyword and Default Arguments, Scope Rules, Higher Order Functions: Treat functions as first class Objects, Lambda Expressions.

Unit 4: Classes and Files: Generate prime numbers with the help of Sieve of Eratosthenes algorithm, File I/O: File input and output operations in Python Programming Exceptions and Assertions Modules: Introduction, Importing Modules, Abstract Data Types: Abstract data types and ADT interface in Python Programming, Classes: Definition and operations in the classes, Special Methods (such as _init_, _str_, comparison methods and Arithmetic methods etc.), Class Example, Inheritance, Inheritance and OOP.

Unit 5: Iterators & Recursion: Recursive Fibonacci, Tower of Hanoi, Search: Simple Search, Binary Search, Estimating Search Time in Simple Search and Binary Search, Sorting & Merging: Selection Sort, Merge List, Merge Sort, Higher Order Sort.

Text and Reference Books:

1. Allen B. Downey, 'Think Python: How to Think Like a Computer Scientist', 2nd edition, Updated for Python 3, Shroff/O, Reilly Publishers, 2016, (http://greenteapress.com/wp/thinkpython/)
2. Guido van Rossum and Fred L. Drake Jr, An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd., 2011.
3. John V Guttag, —Introduction to Computation and Programming Using Python, Revised and expanded Edition, MIT Press, 2013.
4. Robert Sedgewick, Kevin Wayne, Robert Dondero, Introduction to Programming in Python: An Inter-disciplinary Approach, Pearson India Education Services Pvt. Ltd., 2016.
5. Timothy A. Budd, Exploring Python‖, Mc-Graw Hill Education (India) Private Ltd., 2015.
6. Kenneth A. Lambert, Fundamentals of Python: First Programs‖, CENGAGE Learning, 2012.
7. Charles Dierbach, Introduction to Computer Science using Python: A Computational. Problem Solving Focus, Wiley India Edition, 2013.
8. Paul Gries, Jennifer Campbell and Jason Montojo, Practical Programming: An Introduction to Computer Science using Python 3‖, Second edition, Pragmatic Programmers, LLC, 2013. Mapped With: https://ict.iitk.ac.in/product/pythonprogramming-a-practical

Course Outcomes:

1. Understanding Fundamentals of Python Programming
2. Understand and implement Control Structures.
3. Learn and implement Strings and Functions in Python.
4. Understand and implement advance functions like iteration and recursion.
5. Implement Object Oriented Programming concepts in Python

Programme Elective-III 4 (3-1-0)

(Any one of the following five courses)

CLOUD COMPUTING (ECA-572)

Course Content:

Unit-1

Distributed computing models and technologies, Enabling Technologies and System Models for Cloud Computing, techniques, and architectures, Introduction to Cloud Computing including benefits, challenges, and risks.

Unit-2

Cloud Computing Models including Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), Software-as-a-Service (SaaS), virtualization, security and privacy issues, performance and systems issues, capacity planning, disaster recovery, Public cloud, private cloud and hybrid clouds.

Unit-3

Cloud OS, Cloud architectures including federated clouds, challenges in implementing clouds, data centers, hypervisor CPU and memory management, Scalability, Performance, and QoS.

Unit-4

Cloud hosted applications, Data centers for Cloud Computing, Principles of Virtualization platforms and other advanced and research topics in cloud computing.

Unit-5
Security and Privacy issues in the Cloud, VM Ware ESX Memory Management, Capacity Planning and Disaster Recovery in Cloud Computing.

Text and Reference Books:
1. Distributed and Cloud Computing, 1st edition, Morgan Kaufmann, 2011.
2. Technical papers from major journals and major conferences on computing, networking, cloud computing.

Course Outcomes
1. Understand various basic concepts related to cloud computing technologies. (Understand)
2. Understand the architecture and concept of different cloud models: IaaS, PaaS, SaaS. (Understand)
3. Apply the underlying principle of cloud virtualization, cloud storage, data management and data visualization. (Apply)
4. Use different cloud programming platforms and tools. (Apply)
5. Design and deploy cloud application using cloud platforms (Analyze)

SOFTWARE QUALITY ENGINEERING (ECA-574)

Course Content:
Unit-1:
Introduction: Defining Software Quality, Software Quality Attributes and Specification, Cost of Quality, Defects, Faults, Failures, Defect Rate and Reliability, Defect Prevention, Reduction, and Containment, Overview of Different Types of Software Review, Introduction to Measurement and Inspection Process, Documents and Metrics.

Unit-2:
Software Quality Metrics: Product Quality Metrics: Defect Density, Customer Problems Metric, Customer Satisfaction Metrics, Function Points, In-Process Quality Metrics: Defect Arrival Pattern, Phase-Based Defect Removal Pattern, Defect Removal Effectiveness, Metrics for Software Maintenance: Backlog Management Index, Fix Response Time, Fix Quality, Software Quality Indicators.

Unit-3:
Software Quality Management and Models: Modeling Process, Software Reliability Models: The Rayleigh Model, Exponential Distribution and Software Reliability Growth Models, Software Reliability Allocation Models, Criteria for Model Evaluation, Software Quality Assessment Models: Hierarchical Model of Software Quality Assessment.

Unit-4:
Software Quality Assurance: Quality Planning and Control, Quality Improvement Process, Evolution of Software Quality Assurance (SQA), Major SQA Activities, Major SQA Issues, Zero Defect Software, SQA Techniques, Statistical Quality Assurance, Total Quality Management, Quality Standards and Processes.

Unit-5:
Software Verification, Validation & Testing: Verification and Validation, Evolutionary Nature of Verification and Validation, Impracticality of Testing all Data and Paths, Proof of Correctness,

Software Testing, Functional, Structural and Error-Oriented Analysis & Testing, Static and Dynamic Testing Tools, Characteristics of Modern Testing Tools.

Text and References Books:

1. Jeff Tian, Software Quality Engineering (SQE), Wiley-Inter Science, 2005; ISBN 0-471-71345-7.
2. Metrics and Models in Software Quality Engineering, Stephen H. Kan, Addison-Wesley (2002

Course Outcomes:

1. Understand the concept of quality, quality attribute, quality metrics and software technical reviews. (Understand)
2. Understand and discuss the needs for software quality assessment models and apply professional practices in the development of quality software. (Understand, Apply)
3. Understand and apply Software Quality Management Models in the development of software. (Understand, Apply)
4. Understand the concept of software quality assurance and use software quality standards in the development of software. (Understand)
5. Apply the concepts of software verification & validation, error tracking to enforce quality into the software. (Apply)

DIGITAL IMAGE PROCESSING (ECA-576)

Course Content:

Unit-1:

Introduction and Fundamentals Motivation and Perspective, Applications, Components of Image Processing System, Element of Visual Perception, A Simple Image Model, Sampling and Quantization. Image Enhancement in Spatial Domain Introduction; Basic Gray Level Functions–Piecewis e-Linear Transformation Functions: Contrast Stretching; Histogram Specification; Histogram Equalization; Local Enhancement; Enhancement using Arithmetic/Logic Operations–Image Subtraction, Image Averaging; Basics of Spatial Filtering; Smoothing - Mean filter, Ordered Statistic Filter; Sharpening – The Laplacian.

Unit-2:

Image Enhancement in Frequency Domain Fourier Transform and the Frequency Domain, Basis of Filtering in Frequency Domain, Filters – Low-pass, High-pass; Correspondence Between Filtering in Spatial and Frequency Domain; Smoothing Frequency Domain Filters–Gaussian Lowpass Filters; Sharpening Frequency Domain Filters – Gaussian Highpass Filters; Homomorphic Filtering. Image Restoration A Model of Restoration Process, Noise Models, Restoration in the presence of Noise only Spatial Filtering – Mean Filters: Arithmetic Mean filter, Geometric Mean Filter, Order Statistic Filters – Median Filter, Max and Min filters; Periodic Noise Reduction by Frequency Domain Filtering– Bandpass Filters; Minimum Mean-square Error Restoration.

Unit-3:

Color Image Processing Color Fundamentals, Color Models, converting Colors to different models, Color Transformation, Smoothing and Sharpening, Color Segmentation, Morphological Image Processing Introduction, Logic Operations involving Binary Images, Dilation and Erosion, Opening and Closing, Morphological Algorithms–Boundary Extraction, Region Filling, Extraction of Connected Components, Convex Hull, Thinning, Thickening

Unit-4:

Registration Introduction, Geometric Transformation – Plane to P lane transformation, Mapping, Stereo Imaging – Algorithms to Establish Correspondence, Algorithms to Recover Depth

Segmentation Introduction, Region Extraction, Pixel-Based Approach, Multi-level Thresholding, Local Thresholding, Region-based Approach, Edge and Line Detection: Edge Detection, Edge Operators, Pattern Fitting Approach, Edge Linking and Edge Following, Edge Elements Extraction by Thresholding, Edge Detector Performance, Line Detection, Corner Detection.

Unit-5:
Feature Extraction Representation, Topological Attributes, Geometric Attributes Description Boundary-based Description, Region-based Description, Relationship, Object Recognition Deterministic Methods, Clustering, Statistical Classification, Syntactic Recognition, Tree Search, Graph Matching.

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

Course Outcomes:
1. Apply sampling and quantization techniques for conversion of an analog image into digital form. (Apply)
2. Enhance the image using various types of filtering, segmentation and edge detection techniques. (Apply)
3. Analyze and interpret the effects of high pass and low pass filter in an image. (Analyse)
4. Restore the image in the presence of noise by using modern restoration software. (Apply)
5. Use the techniques of morphological image processing, image registration and image recognition. (Apply)
6. Apply various tools and techniques in multidisciplinary engineering and medical fields like embedded programming, CAD, web applications, MRI, CT-Scan, Angiography etc. (Apply)

DATA WAREHOUSING & MINING (ECA-578)

Course Content:
Unit-1:
Overview, Motivation (for Data Mining),Data Mining-Definition & Functionalities, Data Processing, Form of Data Preprocessing, Data Cleaning: Missing Values, Noisy Data, Binning, Clustering, Regression, Computer and Human inspection, Inconsistent Data, Data Integration and Transformation. Data Reduction:-Data Cube Aggregation, Dimensionality reduction, Data Compression, Numerosity Reduction, Clustering, Discretization and Concept hierarchy generation.

Unit-2:
Concept Description:- Definition, Data Generalization, Analytical Characterization, Analysis of attribute relevance, Mining Class comparisons, Statistical measures in large Databases. Measuring Central Tendency, Measuring Dispersion of Data, Graph Displays of Basic Statistical class Description, Mining Association Rules in Large Databases, Association rule mining, mining SingleDimensional Boolean Association rules from Transactional Databases: Apriori Algorithm, Mining Multilevel Association rules from Transaction Databases and Mining Multi-Dimensional Association rules from Relational Databases

Unit-3:
Classification and Predictions:

What is Classification & Prediction, Issues regarding Classification and prediction, Decision tree, Bayesian Classification, Classification by Back propagation, Multilayer feed-forward Neural Network, Back propagation Algorithm, Classification methods K-nearest neighbor classifiers, Genetic Algorithm.
Cluster Analysis:
Data types in cluster analysis, Categories of clustering methods, partitioning methods. Hierarchical Clustering- CURE and Chameleon. Density Based Methods-DBSCAN, OPTICS. Grid Based Methods-STING, CLIQUE. Model Based Method –Statistical Approach, Neural Network approach, Outlier Analysis

Unit-4:
Data Warehousing: Overview, Definition, Delivery Process, Difference between Database System and Data Warehouse, Multi-Dimensional Data Model, Data Cubes, Stars, Snow Flakes, Fact Constellations, Concept hierarchy, Process Architecture, 3-Tier Architecture, Data Mart.

Unit-5:
Aggregation, Historical information, Query Facility, OLAP function and Tools, OLAP Servers, ROLAP, MOLAP, HOLAP, Data Mining interface, Security, Backup and Recovery, Tuning Data Warehouse, Testing Data Warehouse.

Text and Reference Books:
1. M. H. Dunham, ”Data Mining: Introductory and Advanced Topics”, Pearson Education
2. Jiawei Han, Micheline Kamber, ”Data Mining Concepts & Techniques”, Elsevier
3. Sam Anahory, Dennis Murray, “Data Warehousing in the Real World: A Practical Guide for Building Decision Support Systems, 1/e “ Pearson Education
4. Mallach, ”Data Warehousing System”, McGraw –Hill

Course Outcomes:
1. Understand importance of abstraction of Knowledge from unstructured sources at sufficient level. (Understand)
2. Use of high level operational skills and real world case studies for knowledge discovery and data warehousing based principles. (Apply)
3. Understand the areas of probability, statistics and machine learning algorithms which underpin the knowledge discovery enterprise. (Understand)
4. Design data mining and data warehousing systems and solutions to meet user requirements and specifications. (Apply, Analyze)
5. Compare and contrast OLAP and data mining as techniques for extracting knowledge from a data warehouse. (Evaluate)

MOBILE APPLICATION DEVELOPMENT (ECA-580)

Course Content:
Unit-1
Introduction to mobile computing, Characteristics of mobile applications, History of mobile application frameworks, Android Development Environment, Factors in Developing Mobile Applications, Mobile Software Engineering, Frameworks and Tools, Generic UI Development, VUIs and Mobile Apps, Text-to-Speech Techniques, Designing the Right UI, Multichannel and Multi modal UIs.

Unit-2
Overview of mobile application development languages: Java and Android Studio.

Unit-3:
Application models of mobile application frameworks, User-interface design for mobile applications, Managing application data, Integrating with cloud services, Integrating networking, OS and hardware into mobile-applications

Unit-4:
Addressing enterprise requirements in mobile applications – performance, scalability, modifiability, availability and security, Security and Hacking, Active Transactions, Hacking Android

Unit-5:
Testing methodologies for mobile applications, Publishing, deployment, maintenance and management, Platforms and Additional Issues, Development Process, Architecture, Design, Technology Selection, Mobile App Development Hurdles.

Text and Reference Books:

1. Rajiv Ramnath, Roger Crawfis, and Paolo Sivilotti, Android SDK 3 for Dummies, Wiley.2. Bill Phillips, Chris Stewart, Brian Hardy, and Kristin Marsicano, Android Programming: The Big Nerd Ranch Guide, Big Nerd Ranch LLC, 2nd edition, 2015.
3. Christian Keur and Aaron Hillegass, iOS Programming: The Big Nerd Ranch Guide, 5th edition, 2015.
4. Valentino Lee, Heather Schneider, and Robbie Schell, Mobile Applications: Architecture, Design and Development, Prentice Hall, 2004.
5. Tomasz Nurkiewicz and Ben Christensen, Reactive Programming with RxJava, O"Reilly Media, 2016.
6. Raoul-Gabriel Urma, Mario Fusco, and Alan Mycroft, Java 8 in Action: Lambdas, Streams, and Functional-Style Programming, Manning Publications, 2015.
7. Benjamin J. Evans and Martijn Verburg, The Well-Grounded Java Developer: Vital Techniques of Java 7 and Polyglot Programming, Manning Publications, 2013.
8. Brian Fling, Mobile Design and Development, O"Reilly Media, Inc., 2009.

Course Outcomes

1. Understand technology and business trends impacting mobile applications. (Understand)
2. Understand and implement mobile application development languages. (Understand, Apply)
3. Understand the characterization and architecture of mobile applications. (Understand)
4. Understand and design enterprise scale requirements of mobile applications. (Understand, Apply, Analyze)
5. Design and develop mobile applications using application development framework. (Apply, Analyze)

Programme Elective-IV 4 (3-1-0)

(Any one of the following five courses)

CRYPTOGRAPHY & NETWORK SECURITY (ECA-582)

Course Content:

Unit-1:
Introduction to security attacks, services and mechanism, introduction to cryptography. Conventional Encryption: Conventional encryption model, classical encryption techniques substitution ciphers and transposition ciphers, cryptanalysis, stereography, stream and block ciphers. Modern Block Ciphers: Block ciphers principals, Shannon's theory of confusion and diffusion, fiestal structure, data encryption standard(DES), strength of DES, differential and linear crypt

analysis of DES, block cipher modes of operations, triple DES, IDEA encryption and decryption, strength of IDEA, confidentiality using conventional encryption, traffic confidentiality, key distribution, random number generation.

Unit-2:
Introduction to graph, ring and field, prime and relative prime numbers, modular arithmetic, Fermat's and Euler's theorem, primality testing, Euclid's Algorithm, Chinese Remainder theorem, discrete logarithms. Principals of public key crypto systems, RSA algorithm, security of RSA, key management, Diffle-Hellman key exchange algorithm, introductory idea of Elliptic curve cryptography, Elganel encryption.

Unit-3:
Message Authentication and Hash Function: Authentication requirements, authentication functions, message authentication code, hash functions, birthday attacks, security of hash functions and MACS, MD5 message digest algorithm, Secure hash algorithm(SHA). Digital Signatures: Digital Signatures, authentication protocols, digital signature standards (DSS), proof of digital signature algorithm.

Unit-4:
Authentication Applications: Kerberos and X.509, directory authentication service, electronic mail security-pretty good privacy (PGP), S/MIME.

Unit-5:
IP Security: Architecture, Authentication header, Encapsulating security payloads, combining security associations, key management. Web Security: Secure socket layer and transport layer security, secure electronic transaction (SET). System Security: Intruders, Viruses and related threads, firewall design principals, trusted systems.

Text and References Books:
1. William Stallings, "Cryptography and Network Security: Principals and Practice", Prentice Hall, New Jersy.
2. Johannes A. Buchmann, "Introduction to Cryptography", Springer-Verlag.
3. Bruce Schiener, "Applied Cryptography".

Course Outcomes:
1. Understand and deploy cryptographic techniques to secure data in networks. (Understand, Apply)
2. Analyze the vulnerabilities in any computing system and design a security solution. (Apply, Analyse)
3. Understand and use standard algorithms for confidentiality, integrity and authenticity. (Understand, Apply)
4. Apply various key distribution and management schemes in network system. (Apply)
5. Apply security protocols in various IT applications. (Apply)

SOFT COMPUTING (ECA-584)

Course Content:
Unit 1: Introduction to Intelligent Systems and Soft Computing Characteristic behavior of Intelligent systems, Knowledge based systems, Knowledge Representation and Processing, Soft Computing characteristics, Constitutes of Soft Computing-Fuzzy Logic and Computing, Neural Computing, Evolutionary Computing, Rough Sets, Probabilistic Reasoning and Machine Learning.

Unit 2: Neuro Computing - Supervised Learning Biological background, Pattern recognition tasks, Features of artificial neural networks, Activation functions, Perceptron model, Perceptron for

classification and its limitations, Architectures of multilayer feed-forward neural networks, Back-propagation learning algorithm, Limitations of MLP.

Unit 3: Neuro Computing - Unsupervised Learning Hebb's learning rule for competitive learning, Kohonen's self-organizing map and network topology, applications of SOM, Hopfield network and its topology, Boltzman Machines, Adaptive Resonance Theory.

Unit 4: Fuzzy Logic and Fuzzy Systems: Evolution of fuzzy logic, fuzzy sets, fuzzy logic operations, fuzzy relations, Fuzzy arithmetic and fuzzy measures. Fuzzy rules and reasoning, Fuzzy inference systems, Fuzzy modeling and decision making, Neuro-fuzzy modeling.

Unit 5: Evolutionary Computing Biological background and Overview of evolutionary computing, Genetic algorithm and search space, Operators in genetic algorithm- encoding, selection, crossover, and mutation, Classification of GA, Evolutionary Programming and Strategies, Applications of fuzzy in pattern recognition-character recognition. Applications of evolutionary computing in Image processing and computer vision, applications of Soft computing in mobile ad-hoc networks, Information Retrieval, Semantic web, and Software Engineering.

Text and Reference Books:

1. Fakhreddine O. Karray, Clarence De Silva, 'Soft Computing and Intelligent systems design' Pearson Education, ISBN 978-81-317-2324-1.
2. B. K. Tripathy, J. Anuradha, 'Soft Computing: advances and applications', Cengage learning, ISBN-13: 978-81-315-2619-4.
3. S. N. Sivanandam, S. N. Deepa, Principles of Soft Computing, Wiley publications, 2nd Edition.
4. J. S. R. Jang, C. T. Sun, E. Mizutani, 'Neuro-Fuzzy and Soft Computing- A computational approach to Learning and Machine Intelligence' PHI.
5. David E. Goldberg, Genetic Algorithms - Pearson Education, 2006.
6. Satish Kumar, "Neural Networks - A Classroom Approach", Tata McGraw-Hill.

Course Outcomes:

1. Understand differential behavior of Human and Intelligent Systems. (Understand)
2. Understand and use supervised and un-supervised learning techniques in ANN. (Understand)
3. Understand and apply different soft computing techniques like Genetic Algorithms, Fuzzy Logic, Neural Network and their combination. (Understand, Apply)
4. Correlate human-like processing in problem solving with current technologies in various domains like Bio Informatics, Multimedia Systems, Big Data Analytics, etc.
5. Apply evolutionary computing techniques in real life problems. (Apply)

EMBEDDED SYSTEMS (ECA-586)

Course Content:

Unit-1:

Introduction to Embedded Systems Definition and Classification – Overview of Processors and hardware units in an embedded system – Software embedded into the system – Exemplary Embedded Systems – Embedded Systems on a Chip (SoC) and the use of VLSI designed circuits.

Unit-2:

Devices and Buses for Devices Network I/O Devices -Device I/O Types and Examples – Synchronous - Iso-synchronous and Asynchronous Communications from Serial Devices -Examples of Internal Serial-Communication Devices -UART and HDLC -Parallel Port Devices -Sophisticated interfacing

features in Devices/Ports-Timer and Counting Devices -„12C", „USB", „CAN and advanced I/O Serial high speed buses-ISA, PCI, PCI-X, cPCI and advanced buses.

Unit-3:
Programming Concepts and Embedded Programming in C, Programming in assembly language (ALP) vs. High Level Language, C Program Elements, Macros and functions -Use of Pointers -NULL Pointers-Use of Function Calls–Multiple function calls in a Cyclic Order in the Main Function Pointers – Function Queues and Interrupt Service Routines Queues Pointers – Concepts of 'C' Program compilers–Cross compiler–Optimization of memory codes.

Unit-4:
Real Time Operating Systems Timing and clocks in embedded system, Task modelling and management: RTOS Task scheduling models -Handling of task scheduling and latency and deadlines as performance metrics – Co-operative Round Robin Scheduling – Cyclic Scheduling with Time Slicing (Rate Monotonics Co-operative Scheduling) – Preemptive Scheduling Model strategy by a Scheduler – Critical Section Service by a Pre emptive Scheduler – Fixed (Static) Real time scheduling of tasks

Unit-5:
Embedded control and control hierarchy, communication strategies for embedded system: encoding and flow chart. Fault tolerance and formal verification.

Text and References Books:
1. William Stalling, "Computer Organization & Architecture", Pearson education Asia
2. Mano Morris, "Computer System Architecture", PHI

Course Outcomes:
1. Understand the difference between general computing system and embedded system. (Understand)
2. Understand the working of devices, buses and types of communications like serial, parallel etc. (Understand)
3. Understand cross compiler and implement embedded programming using assembly and C languages. (Understand, Apply)
4. Understand real time operating system concepts and develop real time embedded systems. (Understand, Apply)
5. Understand and apply the concept of embedded control and communication. (Understand, Apply)

Virtual Reality (ECA-588)

Data Analytics (ECA-590)