SYLLABUS

(With effect from 2018-19 Academic year)

ಪಠ್ಯಕ್ರಮ

(ಶೈಕ್ಷಣಿಕವರ್ಷ 2018–19)

III & IV Semester
Bachelor Degree
in
Information Science and Engineering

Out Come Based Education
with
Choice Based Credit System

P.E.S. College of Engineering,
Mandya - 571 401, Karnataka
(An Autonomous Institution Affiliated to VTU, Belagavi)
Grant -in- Aid Institution
(Government of Karnataka)
Accredited by NBA, New Delhi
Approved by AICTE, New Delhi.

ಪಿ.ಇ.ಎಸ್. ತಾಂತ್ರಿಕ ಮಹಾವಿದ್ಯಾಲಯ
ಮಂಡ್ಯ–571 401, ಕರ್ನಾಟಕ
(ವಿ.ಟಿ.ಯು, ಬೆಳಗಾವಿ ಅಡಿಯಲ್ಲಿನ ಸ್ವಾಯತ್ತ ಸಂಸ್ಥೆ)

Ph : 08232- 220043, Fax : 08232 – 222075,Web : www.pescemandya.org

Preface

PES College of Engineering, Mandya, started in the year 1962, has become autonomous in the academic year 2008-09. Since, then it has been doing the academic and examination activities successfully. The college is running Eight undergraduate and Eight Postgraduate programs. It consists of Six M.Tech programs, which are affiliated to VTU. Other postgraduate programs are MBA and MCA.

India has recently become a Permanent Member by signing the Washington Accord. The accord was signed by the National Board of Accreditation (NBA) on behalf of India on 13th June 2014. It enables not only the mobility of our degree globally but also establishes equivalence to our degrees with that of the member nations such as Taiwan, Hong Kong, Ireland, Korea, Malaysia, New Zealand, Russia, Singapore, South Africa, Turkey, Australia, Canada and Japan. Among other signatories to the international agreement are the US and the UK. Implementation of Outcome Based Education (OBE) has been the core issue for enabling the equivalence and of Indian degrees and their mobility across the countries.

Our Higher Educational Institution has adopted the CBCS based semester structure with OBE scheme and grading system.

The credit based OBE semester system provides flexibility in designing curriculum and assigning credits based on the course content and hours of teaching.

The OBE, emphasize setting clear standards for observable, measurable outcomes of programs in stages. There lies a shift in thinking, teaching and learning processes moving towards Students Centric from Teacher Centric education. OBE standards focus on mathematics, language, science, attitudes, social skills & moral values.

The key features which may be used to judge, if a system has implemented an outcome based education system is mainly Standard based assessments that determines whether students have achieved the stated standard. Assessments may take any form, so long as the process actually measure whether the student knows the required information or can perform the required task. Outcome based education is a commitment that all students of all groups will ultimately reach the same minimum standards. Outcome Based Education is a method or means which begins with the end in mind and constantly emphasizes continuous improvement.

Choice Based Credit System (CBCS) provides choice for students to select from the prescribed courses (core, Foundation, Foundation Elective, elective, open elective and minor or soft skill courses). The CBCS provides a ‘cafeteria’ type approach in which the students can Choose electives from a wide range of courses of their choice, learn at their own pace, undergo additional courses and acquire more than the required credits, adopt an interdisciplinary approach to learning which enables integration of concepts, theories, techniques, and, perspectives from two or more disciplines to advance fundamental understanding or to solve problems whose solutions are beyond the scope of a single discipline. These greatly enhance the skill/employability of students.

In order to increase the Industry/Corporate readiness, many Soft Skills and Personality Development modules have been added to the existing curriculum of the academic year 2015-16. Industry Interactions have been made compulsory to enhance the field experience. In order to enhance creativity and innovation Mini Project and Industrial visit & Interaction are included in all undergraduate programs.

Dr. Umesh D R
Deputy Dean (Academic)
Associate Professor
Dept. of CS & Engg.

Dr. Nagarathna
Dean (Academic)
Professor
Dept. of CS & Engg

PES College of Engineering

Vision

*"*PESCE shall be a leading institution imparting quality engineering and management education developing creative and socially responsible professionals.*"*

Mission

1. Provide state of the art infrastructure, motivate the faculty to be proficient in their field of specialization and adopt best teaching-learning practices.
2. Impart engineering and managerial skills through competent and committed faculty using outcome based educational curriculum.
3. Inculcate professional ethics, leadership qualities and entrepreneurial skills to meet the societal needs.
4. Promote research, product development and industry-institution interaction.

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

About the Department

The Department of Information science and Engineering takes pride in producing quality engineers over the past 14 years. The credit for all the flowery results goes to the highly motivating staff, from whom all students draw inspiration. The Department was started in the year 2000. The present intake of the undergraduate program is 60. The department has well equipped classrooms, computer laboratories with high-end systems, department library and good collection of software's. Also a research centre is a major credential to our department. We are proud to produce the first PhD student in our college. Faculty members of the department are involved in research activities in different fields such as Medical Image Processing, Pattern Recognition, and Data Mining etc. The department is using Outcome-based education (OBE), which is a recurring education reform model, and it is affiliated to Visvesvaraya Technological University (VTU). The department has achieved good Placement, conducted International /national Conferences and other sponsored short-term courses, workshops, National seminars and symposia. The laboratory facilities and the Internet access are available round the clock to the staff and students of the Information Science and Engineering

Vision

"The department strives to equip our graduates with Knowledge and Skills to contribute significantly to Information Science & Engineering and enhance quality research for the benefit of society".

Mission

M1: To provide students with state of art facilities and tools of Information Science & Engineering to become productive, global citizens and life-long learners.

M2: To prepare students for careers in IT industry, Higher education and Research.

M3: To inculcate leadership qualities among students to make them competent Information Science & Engineering professionals or entrepreneurs.

1.2. State the Program Educational Objectives (PEOs)

Graduates of the program will be able to

PEO1: Establish a productive Information Science & Engineering career in industry, government or academia.

PEO2: Interact with their peers in other disciplines by exhibiting professionalism and team work to contribute to the economic growth of the country.

PEO3: Promote the development of innovative systems and solutions to the problems in Information Science using hardware and software integration.

PEO4: Pursue higher studies in Engineering, Management or Research.

A. List of Program Outcomes (POs)

Engineering Graduates will be able to:

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

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

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

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

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

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

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

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

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

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

PO11. Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

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

B. List of Program Specific Outcomes (PSOs)

Information Science & Engineering Graduates will be able to:

PSO1. Analyze, design, develop and test the principles of System software and Database concepts for computer-based systems.

PSO2. Develop computer communication systems and applications for Information security.

PSO3. Apply the knowledge of Information Science and Engineering to solve any software and hardware related problems and to organize, manage and monitor IT Infrastructure.

III Semester B.E. (IS&E) Scheme of Teaching and Examination 2018- 19

Sl. No	Course Code	Course Title	Teaching Dept.	Hours/ Week			Credits	Examination Marks		
				L	T	P		CIE	SEE	Total Marks
1.	P18MA31	Transform calculus, fouries and numerical techniques	Maths	4	-	-	4	50	50	100
2.	P18IS32	Digital Design	IS&E	3	-	-	3	50	50	100
3.	P18IS33	Data Structures and Algorithms	IS&E	3	-	-	3	50	50	100
4.	P18IS34	Discrete Mathematics & Applications	IS&E	3	-	-	3	50	50	100
5	P18IS35	Computer Organization & Architecture	IS&E	3	-	-	3	50	50	100
6	P18IS36	OOPs with JAVA (FC- I)	IS&E	2	2	-	3	50	50	100
7	P18ISL37	Data Structures Lab	IS&E	-	-	3	1.5	50	50	100
8	P18ISL38	Digital Design Lab	IS&E	-	-	3	1.5	50	50	100
9	P18HU39	*Aptitude and Reasoning Development – Basic BEGINNER (ARDB)	HS&M	2	-	-	-	50	-	-
10	P18HUDIP310	*Comprehensive Communication Development (CCD)	HS&M	2	-	-	[2]	[50]	[50]	[100]
11	P18HUDIP311	*Indian Constitution ,Human Rights & Professional Ethics(ICHRPF)	HS&M	2	-	-	-	[50]	-	-
12	P18MADIP31	* Additional Maths-I	Maths	4	-	-	-	[50]	-	-
Total							22 [24]	400 [450]	400 [450]	800 [900]

*CCD/ICHRPF/Additional Mathematics- I: Lateral Entry (ie. Diploma) students shall have to pass these mandatory learning courses before completion of VI-Semester. CIE only for 50 marks.

IV Semester B.E. Scheme of Teaching and Examination 2018- 19

Sl. No	Course Code	Course Title	Teaching Dept.	Hours/ Week			Credits	Examination Marks		
				L	T	P		CIE	SEE	Total Marks
1.	P18MA41	Complex analysis, statistics, probability and numerical techniques	Maths	4	-	-	4	50	50	100
2.	P18IS42	Database Management System	IS&E	3	-	-	3	50	50	100
3.	P18IS43	Finite Automata and Formal Language	IS&E	3	-	-	3	50	50	100
4.	P18IS44	Design and Analysis of Algorithms	IS&E	3	-	-	3	50	50	100
5	P18IS45	Software Engineering	IS&E	3	-	-	3	50	50	100
6	P18IS46	AVR Microcontroller (FC-II)	IS&E	2	2	-	3	50	50	100
7	P18ISL47	Design and Analysis of Algorithms Lab	IS&E	-	-	3	1.5	50	50	100
8	P18ISL48	Java Programming Lab	IS&E	-	-	3	1.5	50	50	100
9	P18HU49	Aptitude and Reasoning Development – INTERMEDIATE (ARDI)	HS&M	2	-	-	1	[50]	[50]	[100]
10	P18EVDIP50	*Environmental Studies	HS&M	2	-	-	-	[50]	-	-
11	P18MADIP41	* Additional Maths-II	HS&M	4	-	-	-	[50]	-	-
Total							23	450	450	900

*Additional Mathematics- II: Environmental Studies: Lateral Entry (ie. Diploma) students shall have to pass these mandatory learning courses before completion of VI-Semester. CIE only for 50 marks.

Course Title: TRANSFORM CALCULUS, FOURIER SERIES AND NUMERICAL TECHNIQUES			
Course Code: P18MA31	Semester: III	L:T:P:H: 4:0:0:4	Credits: 4-0-0
Contact Period-Lecturer: 52Hrs. Exam: 3Hrs		Weightage:CIE:50%; SEE:50%	

(Common to All Branches)

UNIT-I

Numerical Methods-I: Finite differences**:** Forward and Backward differences, Gregory-Newton forward and backward interpolation formulae, Newton's divided difference formula, Lagrange's interpolation formula and inverse interpolation formula. (All formulae without proof) – Problems only

Central differences: Gauss Forward and Backward difference formulae, Sterling's, and Bessel's formulae (All formulae without proof) – problems.

Self-Study Component: Problems using Everett's formula in Central differences

10 Hrs.

UNIT-II

Numerical differentiation using Newton's forward and backward interpolation formulae, Newton's divided difference formula and Sterling's formula (All formulae without proof)-problems only and Applications to Maxima and Minima of a tabulated function.

Numerical integration: Newton- Cotes quadrature formula, Trapezoidal rule, Simpson's (⅓) $^{rd.}$rule, Simpson's (⅜)th rule, Boole's rule and Weddle's rule (All rules without proof)-Illustrative problems.

Self-Study Component: Derive Newton- Cotes quadrature formula.

10 Hrs.

UNIT-III

Fourier series: Periodic functions, Euler's formula, Dirichlet's conditions. Discontinuous functions, even and odd functions, functions of arbitrary intervals. Half–range Fourier series expansions, complex form of Fourier series, Practical harmonic analysis- Illustrative examples from engineering field.

Self-Study Component: Derivations of Euler's formulae

11 Hrs.

UNIT-IV

Fourier Transforms: Infinite Fourier transforms-properties. Fourier sine and cosine transforms, properties. Inverse infinite Fourier and inverse Fourier sine & cosine transforms –Illustrative examples.

Difference equations and Z-transforms: Definition, Z-transform of standard functions, linearity property, damping rule, shifting rules, initial value theorem and final value theorem (All rules and theorems without proof). Inverse Z – transforms. Difference equations- basic definitions. Application of Z-transforms to solve difference equations.
Self-Study Component: Convolution theorem, Parseval'sidentities.related problems.

10 Hrs.

UNIT-V

Partial differential equations (PDE's):
Formation of PDE by eliminating orbitrary constants and functions. Solution of non-homogeneous PDE by the method of direct integration. Solutions of homogeneous PDE involving derivative with respect to one independent variable only (both types with given set of conditions). Method of separation of variables (first and second order equations). Solution of the Lagrange's linear PDE's of the type: Pp + Qq = R.

Applications of PDE's:
One- dimensional wave and heat equations (No derivation), and various possible solutions of these by the method of separation of variables. D'Alembert's solution of wave equation. Two dimensional

Laplace's equation (No derivation)–various possible solutions. Solution of all these equations with specified boundary conditions (Boundary value problems). Illustrative examples from engineering field.

Self-Study Component: Finding the solution of non-linear equations of first order: Char pit's Method -simple problem. **11 Hrs.**

Text Books:

1. Higher Engineering Mathematics: B.S. Grewal, Khanna Publishers, New Delhi, 42nd Ed. 2012.
2. Advanced Engineering Mathematics: - E. Kreyszig, John Wiley & Sons, 6th Ed.2007.

References:

1. Advanced Modern Engineering Mathematics: - Glyn James, Pearson Education Ltd., 3rd Ed., 2007.
2. Advanced Engineering Mathematics: Peter V O' Neil Thomson, Brooks/Cole, 5th edition, 2007.
3. Higher Engineering Mathematics: - B.V. RAMANA, McGraw Hill Education, 2007

Course Articulation Matrix

Mapping of Course Outcomes (CO) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

Sem: 3	Course code : P18MA31	Title : Engineering Mathematics –III											
CO's	Statement	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO-1	Apply forward, backward difference formulae and central differences formulae in solving interpolation- extrapolation problems in engineering field.	1	**2**	-	-	-	-	-	-	-	-	-	-
CO-2	Numerical differentiation and integration rules in solving engineering where the handling of numerical methods are inevitable	**2**	**2**	-	-	-	-	-	-	-	-	-	-
CO-3	Apply the knowledge of periodic function, Fourier series, complex Fourier series, Fourier sine/cosine series of a function valid in different periods. Analyze engineering problems arising in control theory/fluid flow phenomena using harmonic analysis.	**3**	**3**	-	-	-	-	-	-	-	-	-	-
CO-4	Understand complex/infinite Fourier transforms, Fourier sine and Fourier cosine transforms with related properties Analyze the engineering problems arising in signals and systems, digital signal processing using Fourier transform techniques. Define Z-transforms& find Z-transforms of standard functions to solve the specific problems by using properties of Z-transforms. Identify and solve difference equations arising in engineering applications using inverse Z–transforms techniques.	**2**	**3**	-	-	-	-	-	-	-	-	-	-
CO-5	Define Partial Differential Equations (PDE's),order, degree and formation of PDE's and, to solve PDE's by various methods of solution Explain one - dimensional wave and heat equation and Laplace's equation and physical significance of their solutions to the problems selected from engineering field.	**2**	**3**	-	-	-	-	-	-	-	-	-	-

Note: - Eachunit contains **two** full questions of **20 marks** each. Students are required to Answer any fivefull questions choosing at least **one** full question from each unit

Course title: Digital Design			
Course Code: P18IS32	**Semester: III**	**L-T-P-H : 3-0-0-4**	**Credit:3**
Contact Period: Lecture: 52 Hrs, Exam: 3 Hrs		**Weightage: CIE:50%, SEE: 50%**	

Course Learning Objectives (CLOs)

This course aims to

1. Explain how digital circuit of large complexity can be built in a methodological way, starting from Boolean logic and applying a set of rigorous techniques
2. Discuss Arithmetic Circuits and Data Processing Circuits.
3. Design different units that are elements of typical computer's CPU using VHDL.
4. Discuss flip-flops, latches and registers.
5. Analyze and design Asynchronous and Synchronous Sequential circuits.

UNIT – I

Binary System: Digital Computers and Digital Systems, Binary Numbers, Number Base Conversions, Octal and Hexadecimal Numbers, Complements, Signed Binary Numbers, Binary Codes, Binary Storage and Registers, Binary Logic. **Digital Logic:** The Basic Gates, Universal Logic gates, AND-OR-Invert Gates, Positive and Negative Logic.

Combinational Logic Circuits: Boolean Laws and Theorems, Sum-of-products method, Truth table to Karnaugh Map, Pairs, Quads, and Octets, Karnaugh Simplification.

Self Study: Introduction to HDL. **11 Hrs**

UNIT - 2

Combinational Logic Circuits: Don't Care Conditions, Product-of-Sum Method, Product-of-sum Simplification, Simplification by Quine-McClusky Method, Hazards and Hazard Covers.

Data Processing Circuits: Multiplexers, Demultiplexers, Decoder, BCD-to-Decimal Decoders, Seven segment Decoders, Encoders, Exclusive-OR Gates, Parity Generators and Checkers, Magnitude Comparator, Read-only memory, Programmable Array Logic, Programmable logic.

10 Hrs

Self study: HDL Implementation models, HDL implementation of data processing circuits.

UNIT - 3

Arithmetic Circuits: Binary addition, Binary Subtraction, Unsigned Binary numbers, Sign Magnitude numbers, 2's complement representation, 2's complement arithmetic, Arithmetic Building block, The adder-subtractor, Fast adder, Arithmetic Logic unit, Binary Multiplication and division.

Flip-Flops: RS Flip flops, Gated Flip flops, Edge-triggered RS flip flop, Edge-triggered D flip flop, Edge-triggered JK flip flop, JK master slave flip flop, various representations of flip flops, Conversion of flip flops. **11 Hrs**

Self Study: Arithmetic circuits using HDL, HDL implementation of flip flop.

UNIT - 4

Registers: Types of registers, serial in serial out , serial in parallel out, parallel in serial out, parallel in parallel out, Application of shift registers: Ring counter, Johnson counter, sequence detector and sequence generator. **Asynchronous and synchronous counter:** Asynchronous counters, Decoding gates, synchronous counters, changing the counter modulus, decade counter, Presettable counters, counter design as a synthesis problem. **10 Hrs**

Self Study: Register Implementation in HDL, Counter Design using HDL.

UNIT - 5

Design of Synchronous and Asynchronous Sequential Circuit : Model Selection, State Transition Diagram, State Synthesis Table, Design Equations and Circuit Diagram, Implementation using Read Only Memory, Algorithmic State Machine, State Reduction Technique, Analysis of Asynchronous Sequential Circuit, Problems with Asynchronous Sequential Circuits, Design of Asynchronous Sequential Circuit. **10 Hrs**

Self Study: FSM Implementation in HDL.

Text Books:

1. Digital Principles and Applications: Donald P Leach, Albert Paul Malvino & Goutham Saha, TMH, 8th Edition, 2014.
2. A Verilog HDL Primer, 2nd Edition, J. Bhaskar, BS Publications
3. Morris mano, Digital logic and computer design(1st ed.), Pearson education india,2016

Reference Books:

1. Digital Principles & Design by Donald D Givone, 4th Reprint, Tata McGraw Hill 2009

Course Outcomes

After learning all the units of the course, the student is able to,

1. Apply the principles of Boolean algebra/K - Map to manipulate and minimize logic expressions/functions
2. Analyze and design Arithmetic Circuits and Data processing Circuits
3. Design different units that are elements of typical computer's CPU using VHDL
4. Design logic circuits using flip-flops/latches/registers
5. Analyze and design Asynchronous and Synchronous Sequential circuits

Course Articulation Matrix (CAM)															
Course Outcomes	**Program Outcomes (PO's)**											**PSO's**		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO 1	2	3	2			1							1		2
CO 2	2	3	3	2		1								3	3
CO 3	3	3	3			2								3	3
CO 4	2	2	3			2							2		3
CO 5	2	3	3			2									3

Course title: Data Structures and Algorithms			
Course Code: P18IS33	Semester: III	L-T-P-H : 3-0-0-4	Credit:3
Contact Period: Lecture: 52 Hrs, Exam: 3 Hrs		Weightage: CIE:50%, SEE: 50%	

Prerequisite: Computer Concepts and C Programming

Course Learning Objectives (CLOs)

This course aims to

1. Analyze the need for data structuring techniques, and Design and Implement standard data structures like stack using recurssion.
2. Learn the different types of linked list and Design and implement operations on SLL, DLL, Circular SLL and Circular DLL using header nodes.
3. Learn the Basic operations on - Linear queue, Circular queue, Priority Queue and Double ended Queue and Design and Implement different types of queues Using SLL.
4. Identify the different tree traversal techniques and Design and implement different tree traversal techniques using iteration and recursion.
5. Learn the different sorting and searching techniques and Analyze the performance of the different sorting and searching techniques.

Relevance of the Course: This course is one of the foundation courses in Information Science and Engineering program, which helps the student to understand the importance of Data Structures and its applications usage in problem solving in Information Science and Engineering .

Data structure is one of the ways of organizing and storing the data in different formats such as Stack, Queue, Linked list and Trees. Every Data structure has its advantages and disadvantages. This Course is helps the student to understand which data structure is best suited for problem need to be solved in Information Science and Engineering.

Course Content

UNIT – I

Introduction to data Structures-Definition, Classification of Data Structures.

Stacks: Representing stack in C- Implementation of Push, Pop and display operations using arrays and pointers. Example of Stacks: Infix, Postfix, Prefix, Infix to postfix, prefix to postfix, evaluation of postfix.

Recursion: Definition, Writing Recursive Programs-Factorial Numbers, Fibonacci Numbers and Tower of Hanoi Problem.

Self study component: Some other Applications of stacks like check whether the given string is palindrome or not, reverse of a string. **10 Hrs**

UNIT – II

Linked Lists: Static Memory Allocation and Dynamic Memory Allocation, Basic operations on SLL, DLL, Circular SLL and Circular DLL: insertion, deletion and display.

Self study component: Implementation of SLL, DLL using Header nodes

1**0 Hrs**

UNIT – III

Applications of Linked Lists: Merging, Reversing, Searching, Addition of two polynomials using SLL. **Queues:** Definition, Representation, operations, implementation using arrays and linked lists. Different types of queues, Basic operations on - Linear queue, Circular queue,

Priority Queue and Double ended Queue (Using SLL), Applications of Queues.
Self study component: Implementation of Job scheduling algorithm using priority queue.

10 Hrs

UNIT – IV

Trees: Introduction-Definition, Tree Representation, Properties of Trees, Operations on Binary tree, Binary Search Tree [BST] - Definition, searching BST, Insertion to BST, Deletion from BST, Display BST; Tree and their Applications- Tree Traversal, General Expression as a tree, Evaluating an Expression Tree; Threaded Binary Trees-Threads, In order Traversal of a Threaded Binary Tree.
Self study component: Inserting a Node into a Threaded Binary Tree.

12 Hrs

UNIT – V

Sorting Techniques: Radix sort, Address Calculation sort, Binary tree sort, Heap sort.
Searching Techniques: sentinel search, probability search, ordered list search
Self study component: Interpolation Search **10 Hrs**

Text Book:

1. "Data Structures using C and C++ ", Yedidyah Langsam and Moshe J. Augenstein and Aaron M.Tenanbaum , 2nd Edition , PHI.
2. "Data Structures – A pseudo code Approach with C ", Richard F Gilberg and Behrouz A forouzan, 2nd Edition .

Reference Books:

1. "Fundamentals of Data Structures in C ", Horowitz, Sahani, Anderson-Freed , Second Edition, University Press .

Course Outcomes

After learning all the units of the course, the student is able to

1. Understand primitive and derived data structure and Understand Abstract data types, Stacks and recursion.
2. Develop and implement linked list.
3. Develop programs to implement different queues.
4. Understand and create trees.
5. Design an algorithm to Sorting Techniques and Searching techniques.

Course Articulation Matrix (CAM)

Course Outcomes	**Program Outcomes (PO's)**												**PSO's**		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO 1	1	3	3		2				2				1	2	
CO 2	1	3	2		2				2					2	2
CO 3	1	3	2		2				2				1	2	
CO 4	2	3	2		2				2				2	2	2
CO 5	2	3	2		2				2				2		

Course title: Discrete Mathematics and Applications			
Course code: P18IS34	**Semester: III**	**L-T-P-H : 3-0-0-4**	**Credit:3**
Contact Period: Lecture:52 Hrs	**Exam: 3 Hrs**	**Weightage: CIE: 50, SEE: 50**	

Course Learning Objectives (CLOs)

This course aims to

1. Prepare for a background in abstraction, notation, and critical thinking of mathematics directly related to problems of computer science.
2. Understand and apply logic, relations, functions, basic set theory, countability and counting arguments, proof techniques.
3. Understand and apply mathematical induction, combinatorics, discrete probability, sequence and recurrence, elementary number theory
4. Understand and apply graph theory and mathematical proof techniques.

Course Content

UNIT - I

Fundamentals of Logic: Truth Tables, Logic Equivalence – The Laws of Logic, Logical Implication – Rules of Inference. The Use of Quantifiers, Quantifiers, Definitions and the Proofs of Theorems.

Self Study: Basic Connectives, proofs of theorems. **10 Hrs**

UNIT - II

Properties of the Integers: Mathematical Induction, the Well Ordering Principle – Mathematical Induction, Recursive Definitions. The Division Algorithm: Prime numbers. Fundamental Permutations, Combinations – The Binomial Theorem, Combinations with Repetition

Self Study: Principles of Counting: The Rules of Sum and Product. **10 Hrs**

UNIT - III

Relations and Functions: Relations, Functions – Plain and One-to-One, Onto Functions. The Pigeon-hole Principle, Function Composition and Inverse Functions. Properties of Relations.

Computer Recognition – Zero-One Matrices and Directed Graphs, Partial Orders – Hasse Diagrams, Equivalence Relations and Partitions.

Self Study: Cartesian Product and Relations **11 Hrs**

UNIT - IV

The Principle of Inclusion and Exclusion: The Principle of Inclusion and Exclusion, Generalizations of the Principle, Derangements – Nothing is in its Right Place, Rook Polynomials. Arrangement with forbidden position.

Recurrence Relations: First Order Linear Recurrence Relation, The Second Order Linear Homogeneous Recurrence Relation with Constant Coefficients.

Self Study: Addition Principle **10 Hrs**

UNIT - V

Introduction to Graph Theory: Definitions and Examples, Sub graphs, Complements, and Graph Isomorphism, Vertex Degree, Euler Trails and Circuits, Planar graphs. Trees: Definitions, Properties, and Examples, Routed Trees, Trees and Sorting, Weighted Trees and Prefix Codes

Self Study: Graph coloring and chromatic polynomials

11 Hrs

Text Book:

1. Ralph P. Grimaldi: Discrete and Combinatorial Mathematics, 5th Ed, Pearson Ed. 2004

Reference Books:

1. Basavaraj S Anami and Venakanna S Madalli: Discrete Mathematics – A Concept based approach, Universities Press, 2016 2.
2. Kenneth H. Rosen: Discrete Mathematics and its Applications, 6th Edition, McGraw Hill, 2007.
3. Jayant Ganguly: A Treatise on Discrete Mathematical Structures, Sanguine-Pearson, 2010.
4. D.S. Malik and M.K. Sen: Discrete Mathematical Structures: Theory and Applications, Thomson, 2004. 5. Thomas Koshy: Discrete Mathematics with Applications, Elsevier, 2005, Reprint 2008.

Course Outcomes

After studying this course, students will be able to

1. Verify the correctness of an argument using propositional and predicate logic
2. Demonstrate the ability to solve problems using counting techniques and combinatorics in the context of discrete probability.
3. Solve problems involving recurrence relations.
4. Construct proofs using direct proof, proof by contraposition, proof by contradiction, proof by cases, and mathematical induction.
5. Ability to Explain and distinguish graphs and their properties.

Course Articulation Matrix (CAM)															
Course Outcomes	**Program Outcomes (PO's)**												**PSO's**		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO 1	3	3	1							3			2	2	2
CO 2	3	2	2							2			2	2	2
CO 3	3	2	2							2			2	2	2
CO 4	3	2	2							2			2	2	2
CO 5	3	2	2							2			2	2	2

Course title: Computer Organization and Architecture			
Course Code: P18IS35	**Semester: III**	**L-T-P-H: 3-0-0-4**	**Credit:3**
Contact Period: Lecture: Lecture:52Hr, Exam: 3 Hr		**Weightage: CIE:50%, SEE: 50%**	

Prerequisites:

1. Digital logic design
2. Basic mathematics
3. Fundamentals of computers

Course Learning Objectives

The student should be able to:

1. Understand the basic structure of a computer and execution of instructions.
2. Identify the major hardware, Input/output components of the system.
3. Understand the basic concepts of memory system, Memory replacement algorithms, and performance considerations
4. Analyze the execution of instructions /programs knowing the basic principles of computer architecture and assembly language.
5. Analyze and compare the algorithms used in arithmetic unit.

Relevance of the Course: This course provides detail of computer system's functional components, their characteristics, performance and interactions including system bus, different types of memory and input/output organization and CPU. This course also covers the architectural issues such as instruction set program and data types. The students are also introduced to the increasingly important area of parallel organization. This course also serves as a basic to develop hardware related projects.

Course Content

UNIT -I

Basic Structure of Computers: Computer Types, Functional Units, Basic Operational Concepts, Bus Structure, Performance,

Machine Instructions and Programs: Number, Arithmetic Operations, and Character Representation, Memory Locations and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Basic Input/Output Operations.

Self-Study Component: Assembly Language **11 Hrs**

UNIT -II

Machine Instructions and Programs Cont'd): Stacks and Queues, Subroutines, Additional Instructions, Encoding of machine Instructions

Input/output Organization: Accessing I/O Devices, Interrupts-Interrupt hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Controlling Device Requests, Exceptions, Use of Interrupts in OS, DMA-Bus Arbitration, Buses-Synchronous bus and asynchronous bus.

Self-Study Component: Interface Circuit **10 Hrs**

UNIT -III

Basic Processing Unit: Fundamental Concepts, Execution of complete instruction, multiple bus organization, Hardwired Control, Micro programmed control.

Self-Study Component: Micro programmed control- Emulation **11 Hrs**

UNIT- IV

The Memory System: Basic Concepts, Semiconductor RAM Memories, Speed, Size and Cost, Cache Memories, Performance Considerations, Virtual Memories, Memory Management Requirements

Self-Study Component: Read-Only Memories **10 Hrs**

UNIT-V

Arithmetic: Multiplication of Positive Numbers, Signed-Operand Multiplication, Fast Multiplication, Integer Division.

Introduction to Multi -core Architecture: Motivation for Concurrency in software, parallel computing platforms-Parallel computing in Microprocessors, Differentiating Multi-core Architectures from Hyper-Threading Technology, Multithreading on single core versus Multi-core platforms, Understanding performance - Amdhal's Law.

Self-Study Component: Floating-Point Numbers and Operations **10 Hrs**

Text Book:

1. Computer Organization and Embedded Systems, Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Naraig Manjikian ,5th Edition, McGraw Hill, 2017.
2. Multi-Core Programming, Shameem Akther and Jason Roberts, Intel Press.

Reference Book:

1. Computer Organization & Architecture, William Stallings, 9th Edition, PHI, 2013.
2. Computer Systems Design and Architecture, Vincent P. Heuring & Harry F. Jordan, 2nd Ed. Pearson Education

Course Outcomes

After studying this course, students will be able to

1. Analyze program execution.
2. Explain the basic input/output operations.
3. Develop the control sequence for a given instruction.
4. Design the memory system using various techniques.
5. Analyze different algorithms for performing arithmetic operations and understand need for multithread.

Prerequisites: Computer concepts and C Programming

Course Articulation Matrix (CAM)															
Course Outcomes	**Program Outcomes (PO's)**											**PSO's**		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO 1	3	2	2										2		1
CO 2	2	3	3	1									2		2
CO 3	2	2	3	2									2		2
CO 4	3	3	2	1									2		2
CO 5	1	2	2	1									2		2

Course Title : Object Oriented Programming with Java (FC-II)			
Course Code : P18IS36	**Semester : III**	**L:T:P:H : 2:2:0:4**	**Credits: 3**
Contact Period: Lecture: 52 Hr, Exam: 3 Hr		**Weightage: CIE:50; SEE:50**	

Prerequisites: Computer concepts and C Programming

Course Learning Objectives (CLOs)

This course aims to

1. Explain the significance of object oriented concepts
2. Describe the concept of class, objects and methods in Java
3. Apply the concepts of inheritance and interfaces in Java
4. Illustrate usage of packages, string handling and exception handling in Java
5. Understand and explain need for multi threading, generics and file handling in Java

Course Content

UNIT-I

Object Oriented Concepts : Fundamentals of Object Oriented programming - Object oriented paradigm, basics concepts of object oriented programming, benefits of object oriented programming, applications of object oriented programming.

Program Structure in Java: Brief introduction to data types, scope of variable identifier, literal constants, symbolic constants, user input to programs, formatted output, operators and control statements in java, Writing simple Java programs.

Arrays Introduction, Declaration and Initialization of Arrays, Accessing Elements of Arrays, Operations on Array Elements, Assigning Array to Another Array, Dynamic Change of Array Size, Class Arrays, Two-dimensional Arrays, Three-dimensional Arrays

Self-Study Component: Type casting in Java

10 Hrs

UNIT-II

Classes and Objects Introduction, Class Declaration and Modifiers, Class Modifiers, Declaration of Class Objects, Assigning One Object to Another ,Access Control for Class Members, Accessing Private Members of Class, Constructor Methods for Class, Nested Classes, Final Class and Methods , Passing Arguments by Value and by Reference, Keyword this.

Methods- Introduction, Defining Methods, Overloaded Methods, Overloaded Constructor Methods, Class Objects as Parameters in Methods, Access Control, Nesting of Methods, Overriding Methods, Attributes Final and Static.

Self-Study Component: Lambda Expression **11 Hrs**

UNIT-III

Inheritance Introduction, Process of Inheritance, Types of Inheritances, Universal Super Class, Inhibiting inheritance of Class Using Final, Access Control and Inheritance, Multilevel Inheritance, Application of Keyword Super, Constructor Method and Inheritance, Method Overriding, Abstract Classes, Interfaces and Inheritance.

Interfaces Introduction, Similarities between Interface and Class, Declaration of Interface, Implementation of interface, Multiple Interfaces.

Self-Study Component: Dynamic Method Dispatch. **10 Hrs**

UNIT-IV

Packages Introduction, Defining, Importing Packages and Classes into Programs, Path and Class Path, Access Control, Java. lang Package and its Classes, Class object

String Handling in Java Introduction, Interface Char Sequence, Class String, Methods for Extracting Characters from Strings, Methods for Comparison of Strings, Methods for Modifying Strings, Methods for Searching Strings, Class String Buffer

Exception Handling- Introduction, Hierarchy of Standard Exception Classes, Keywords throws and throw, try, catch, and finally Blocks, Multiple Catch Clauses, Custom exceptions.

Self-Study Component: Class String Builder. **11 Hrs**

UNIT-V

Multithreaded Programming Introduction, Need for Multiple Threads, Thread Class, Main Thread, Creation of New Thread by Extending Thread Class, Creation of New Threads by Implementing Runnable, Creation of Multiple Threads by Runnable

Generics Introduction, Generics and Primitive Types, Declaration of Generic Class and Constructor, Generic Method

Input out streams and File operations Introduction, Class File, File Input Stream, File Output Stream

Self-Study Component: Thread States and Priority **10 Hrs**

Text Books:

1. E Balagurusamy Programming With Java : A Primer 5th Edition Tata McGraw Hill.
2. Java: One Step Ahead by Anita Seth , B.L. Juneja ,OXFORD University press - First Edition 2017.

Reference Book:

1. Herbert Schildt, Java The Complete Reference, 9th Edition, Tata McGraw Hill.

Course outcomes

After learning all the units of the course, the student is able to

1. Explain the object-oriented concepts and apply Java features to develop simple Java programs.
2. Understand the concepts of classes, objects and methods.
3. Demonstrate the usage of Inheritance and Interfaces.
4. Implement the concepts involving Packages, String handling and Exception Handling
5. Apply the concepts of multi-threading, generics and files in java

Course Articulation Matrix (CAM)

Course Outcomes	Program Outcomes (PO's)												PSO's		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO 1	2	2											2		
CO 2	2	3	2										2		
CO 3	2	2	2										2		
CO 4	2	2	2										2		
CO 5	2	2	2										2		

Course title: Data Structure Lab			
Course Code: P18ISL37	**Semester: III**	**L-T-P-H: 0 -0–0-3**	**Credit:1.5**
Contact Period: Lecture: 39 Hr, Exam: 3 Hr		**Weightage: CIE:50%, SEE: 50%**	

Prerequisites: Computer Concepts and C Programming.

Course Learning Objectives (CLOs)

This course aims to

1. Apply different concepts of data structures to solve real time problems.
2. Distinguish between iterative method and recursive method.
3. Apply the concept of recursion, stack, queues and Linked list to solve various applications.
4. Solve non-linear data structures, such as binary tree.
5. Implement different sorting and searching techniques.

Course Content

1. Write the C programs using Recursion.
2. Write a C program to implement the stack of integers. The program should print appropriate messages for stack overflow, stack underflow and stack empty.
3. Write a C program to convert and print a given valid parenthesized infix arithmetic expression to postfix expression.
4. Write a C program to evaluate a valid suffix/postfix expression using stack.
5. Write a C program using dynamic variables and pointers, to implement a singly linked list.
6. Write a C program to implement the applications of single linked list.
7. Write a C program to implement Queue of integers using an array.
8. Write a C program to implement circular Queue of integers using an array.
9. Write a C program using dynamic variables and pointers to construct a queue of integers using singly linked list.
10. Write a C program to simulate the working of a Double Ended Queue of integers using an array.
11. Write a C program to design a priority queue.
12. Write a C program to demonstrate the working of binary search tree.
13. Write a C program to sort the given list of N numbers.
14. Write a program to search an element in a given list of N numbers.

Course Outcomes

1. Implement the programs on stack.
2. Implement the programs on linked list
3. Implement the programs on queues.
4. Implement the programs on trees
5. Implement the programs on different sorting and searching techniques.

Course Articulation Matrix (CAM)															
Course Outcomes	**Program Outcomes (PO's)**												**PSO's**		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO 1	2	2	2		2				2					3	2
CO 2	2	2	2		2				2					3	2
CO 3	2	2	2		2				2					3	2
CO 4	2	2	2		2				2					3	2
CO 5	2	2	2		2				2					3	2

Course title: Digital Design Lab			
Course Code: P18ISL38	**Semester: III**	**L-T-P-H: 0-0-3- 3**	**Credit:1.5**
Contact Period: Lecture: 39 Hr, Exam: 3 Hr		**Weightage: CIE:50%, SEE: 50%**	

Course Learning Objectives (CLOs)

This course aims to

1. Design and implement different combinational circuits.
2. Design and implement different sequential circuits.
3. Design and implement D/A converter.

Course Content

PART A

1. Design a circuit for Full Adder.
2. Design a circuit for code Conversion.
3. Application of MUX/DEMUX.
4. Application of Decoder.
5. Design of 3 bit Synchronous Counter.
6. Design of Asynchronous Counter.
7. Design of Ring Counter/Johnson Counter.
8. Design of Sequence generator/detector.
9. Digital to analog Converter.

PART B

1. Write the Verilog/VHDL code for a Full Adder. Simulate and verify it's working.
2. Write the Verilog/VHDL code for 8:1 MUX. Simulate and verify its working.
3. Write the Verilog/VHDL code for a 3:8 decoder. Simulate and verify it's working.
4. Write the Verilog/VHDL code for a Flip-Flop with positive-edge triggering. Simulate and verify its working.
5. Write the Verilog/VHDL code for a mod-8 up counter. Simulate and verify it's working.
6. Write the Verilog/VHDL code for a Ring Counter. Simulate and verify it's working.
7. Write the Verilog/VHDL code for a Johnson Counter. Simulate and verify it's working.

Note: In SEE, student has to pick a lot for question that contains subsections from PART A and PART B and has to execute both the subsections compulsorily.

Course Outcomes

1. Design and implement different combinational circuits and sequential circuits
2. Design and implement 4-bit D/A converter.
3. Simulate and verify the working of a logic circuit by writing the Verilog/VHDL code using tool such as Xilinx ISE 7

Course Articulation Matrix (CAM)

Course Outcomes	Program Outcomes (PO's)												PSO's		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO 1	2	2	2		2										2
CO 2	2	2	2		2										2
CO 3	2	2	2		2										2

Course Title: Aptitude and Reasoning Development - BEGINNER. (ARDB)			
Course Code : P18HU39	Semester : III	L-T-P-H : 2-0-0-2	Credits: NA
Contact Period: Lecture: 32 Hr. Exam: 3 Hr		Weightage :CIE:100% - [P/NP]	

Prerequisites: Basics of mathematics.

Course Learning Objectives (CLOs)

This course aims to

1. Solve the mathematical calculations easily and quickly using the methods of vedic mathematics.
2. Illustrate different examples to learn about percentages effectively.
3. Compare the different types of series.
4. Explain the logic behind solving problems under series such as A.P.,G.P.,H.P.
5. Explain divisibility rules, properties of different types of numbers.
6. Explain methods to find the number of factors and sum of factors.
7. Analyze the concept of power cycle, and find last digit and last two digits.
8. Solve problems involving simple equations and inequalities.
9. Explain Componendo, Dividendo, Invertendo, Alternendo and other terms related to ratio and proportion.
10. Explain the concepts behind the logical reasoning modules such as arrangement, blood relations and directions

Relevance of the course:

3rd Semester is considered as the right time to build a base to a student’s analytical and logical ability. This course connects the basics of maths learnt in school into the present problem solving techniques. It creates an awareness towards the importance and significance of an individual’s logical abilities.

Course Content

UNIT – I

Sharpen your axe!!

Vedic mathematics:

Viniculum and de- viniculum, subtractions using viniculum .Nikhilum multiplication: For numbers close to base values, multiplication of any two digit numbers or three digits number using criss cross method. Finding the square, square root, cubes, cube root of two digit and three digit numbers quickly. Approximation in multiplication and division. Checking the answer using digital sum method

SSC- Get hands on multiplication tables, increasing the speed in basic arithmetic operations. Classification of numbers.

Percentage calculations and ratio comparison:

Percentage calculations: Percentage rule for calculating, percentage values through additions, percentage– fraction table, approximation in calculating percentages. Application based problems

Ratio comparison: calculations method for ratio compressions: 1. the cross multiplication method, 2. percentage value compression method 3. Numerator and denominator percentage change method. Method for calculating the value of percentage change in the ratio. Application based problems.

SSC- Thorough with fractions and decimal values. Applications of tabulated fractions. Product of means and extremes. **8 Hrs**

UNIT – II

Analytical Reasoning 1: series

Number series: Standard patterns of number series, pure series: perfect square, square cube, prime, combination of this series. Difference series, ratio series, mixed series, geometric series, two-tier arithmetic series, three-tier arithmetic series, and change in the order for difference series, change in the order for ratio series, sample company questions.

Letter series: Alphabet and Alphanumeric series, finding the missing term based on logic learnt in number series module, continuous pattern series, and correspondence series. Sample company questions.

Picture series: image analysis, addition deletion rotation or modification of lines or shapes. Understanding the symmetry of the image. Mirror image analysis. Sample company questions.

SSC- Basic knowledge of letter positions, Different number series for example – even, odd, prime, composite etc. **6 Hrs**

UNIT – III

Number system:

Introduction, **Integers:** Remainder zero concept, Odd and Even Integers, Negative and positive integers, power number a^x, properties of a perfect square number. **Prime number:** General method to identify the prime number, properties of prime numbers. Euler's number. **Factorial number:** Wilson's theorem, important results on factorial. **Divisor:** number of divisors, sum of divisors, number expressed as the product of two factors.

Divisibility rules: divisibility of a whole number by a whole number, divisibility of an expression by an expression. **Modulus concept:** divisibility rules in modulus, rules of operations in modulus. **Finding one remainder:** One divisor, remainder of (a^n– b^n), remainder for more than one divisor.

UNIT digit: Concept of power cycle, finding last two digits**.** Number of trailing zeroes.

SSC-Basic arithmetic operations, knowledge about quotient and remainders, multiples and factors.

6 Hrs

UNIT – IV

Simple equations, Ratio Proportions and Variations:

Simple equations: Linear equations-Linear equations in one variable, linear equation in two variables, Different methods of solving linear equations in two variables– Method of elimination, Method of substitution, Method of cross multiplication. Format of equations that can be converted to linear equations, Linear equations of three variables, Inequalities and its properties. Advanced problems on Simple equations. Age problems.

Ratio Proportions and Variations: Understanding the meaning and difference between ratio, proportion and variation. Properties of ratio, Comparison of more than two quantities, Proportion, Properties of proportion - Componendo, Dividendo, Invertendo, Alternendo. Continued proportion, Mean proportion. Variation - Direct variation, indirect variation, Joint variation, Short cut methods to solve problems on variation.

SSC-Knowledge about factors, types of factors. Splitting the middle term rule, formula rule.

6 Hrs

UNIT – V

Building the fundamentals of logical reasoning:

Arrangement:

Approach to tackle questions, Different types of arrangement– Linear arrangement, Circular arrangement. Selection, Double line map. Possible ways of arrangement– Words or numbers, left side only, right side only, left right alternate, increasing or decreasing order, interchange vs push, Strategy for solutions– some tips for quick answers, general strategy.

Directions:

Basics. Pythagorean Theorem, Pythagorean triplets, solving problems for practice.

Blood relations:

Some typical relations that we come across, family tree, structuring the given problem step by step. Suggested methods– Backtracking, drawing family tree. Problems on blood relations and professions.

SSC-Basic knowledge of directions, Pythagoras theorem. Logical reasoning skills, Relations, Family tree. **6 Hrs**

Reference Books:

1. The Trachtenberg speed system of basic mathematics, published by Rupa publications.
2. CAT Mathematics by AbhijithGuha. Published by PHI learning private limited.
3. Quantitative aptitude by Dr. R. S Agarwal published by S.Chand private limited.
4. Verbal reasoning by Dr. R. S Agarwal , published by S. Chand private limited.
5. Quantitative aptitude for CAT by Arun Sharma, published by McGraw Hill publication.
6. Analytical reasoning by M.K Pandey BSC PUBLISHING.CO.PVT.LTD

Course Outcomes

After learning all the UNITs of the course, the student is able to:

1. Solve mathematical calculations in less duration compared to the conventional method.
2. Give examples for AP, GP and HP and differentiate between them.
3. Apply divisibility rules, power cycle method and evaluate the significance of the number system module.
4. Point out the errors in the problems concerning inequalities and solve simple equations and problems based on ratio, proportion and variation.
5. Solve the problems based on blood relations, directions and arrangement.

Course Title : Additional Mathematics -I			
Course Code : **P18MADIP31**	Semester : III	L:T:P:H : 4:0:0:4	Credits: 0
Contact Period: Lecture: 52 Hr.		Weightage: CIE (max:50)	

((Mandatory Learning Course: **Common to All Branches)**

(A Bridge course for Diploma qualified students of III Sem. B. E.)

UNIT -I

Complex Trigonometry: Complex Numbers: Definitions & properties. Modulus and amplitude of a complex number, Argand's diagram,

Vector Algebra: Scalar and vectors. Vectors addition and subtraction. Multiplication of vectors (Dot and Cross products). Scalar and vector triple products-simple problems.

Self-study components: De-Moivre's theorem (without proof). Roots of complex number – Simple problems. **12 Hrs**

UNIT -II

Differential Calculus: Polar curves –angle between the radius vector and the tangent pedal equation- Problems. Maclaurin's series expansions- Illustrative examples.

Partial Differentiation: Euler's theorem for homogeneous functions of two variables. Total derivatives-differentiation of composite and implicit function.

Self-study components: Review of successive differentiation. Formulae for n^{th} derivatives of standard functions- Liebnitz's theorem (without proof). Application to Jacobians, errors & approximations. **10 Hrs**

UNIT -III

Integral Calculus: Statement of reduction formulae for $sin^{n}x$, $cos^{n}x$, *and* $sin^{m}xcos^{m}x$ and evaluation of these with standard limits-Examples. Applications of integration to area, length of a given curve, volume and surface area of solids of revolution.

Self-study components: Differentiation under integral sign (Integrals with constants limits)-Simple problems. **10 Hrs**

UNIT-IV

Vector Differentiation: Differentiation of vector functions. Velocity and acceleration of a particle moving on a space curve. Scalar and vector point functions. Gradient, Divergence, Curl and Laplacian (Definitions only).

Self-study components: Solenoidal and irrotational vector fields-Problems. **10 Hrs**

UNIT-V

Ordinary differential equations (ODE's): Introduction-solutions of first order and first degree differential equations: homogeneous**,** exact, linear differential equations of order one and equations reducible to above types

Self-study components: Applications of first order and first degree ODE's - Orthogonal trajectories of cartesian and polar curves. Newton's law of cooling, R-L circuits- Simple illustrative examples from engineering field. **10 Hrs**

Text Book:

1. B.S. Grewal: Higher Engineering Mathematics, Khanna Publishers, New Delhi, 42nd Ed. 2012.

References:

1.zE. Kreyszig: Advanced Engineering Mathematics, John Wiley & Sons, 6th Ed., 2007.

2. N.P.Bali and Manish Goyal: Engineering Mathematics, Laxmi Publishers, 7th Ed., 2007.

Course Title: COMPLEX ANALYSIS, STATISTICS, PROBABILITY AND NUMERICAL TECHNIQUES			
Course Code: P18MA41	Semester: IV	L:T:P:H: 4:0:0:4	Credits: 4-0-0
Contact Period-Lecturer: 52Hrs. Exam: 3Hrs		Weightage:CIE:50%; SEE:50%	

COMPLEX ANALYSIS, STATISTICS, PROBABILITY AND NUMERICAL TECHNIQUES

(Common to all Branches)

UNIT-I

Numerical Methods-II: Solution of algebraic and transcendental equations: Bisection method, Regula-Falsi & Newton-Raphson method. Fixed point iteration method: Aitken's Δ^2- process - Illustrative examples only. Numerical solution of ordinary differential equations (ODE's): Numerical solutions of ODE's of first order and first degree – Introduction. Taylor's series method. Modified Euler's method, Runge - Kutta method of fourth order, Milne's and Adams-Bashforth predictor & corrector methods (All formulae without proof). Illustrative examples only.

Self-Study Component: Solution of second order ordinary differential equations using Runge-Kutta methods. Solution of first order simultaneous differential equations.

10 Hrs.

UNIT-II

Numerical methods for system of linear equations- Gauss-Jacobi and Gauss- Seidel iterative methods. Relaxation method**.** Determination of largest Eigen value and corresponding Eigen vector by power method.Series solutions of ODE's and special functions:Power series solution of a second order ODE, Series solution-Frobenius method. Series solution of Bessel's differential equation leading to $J_n(x)$. Expansions for $J_{\frac{1}{2}}(x)$ *and* $J_{-\frac{1}{2}}(x)$.. Series solutions of Legendre's differential equation leading to $P_n(x)$-Legendre's polynomials. Rodrigue's formula (No Proof) - simple illustrative examples.

Self-Study Component: Basics of Series solutions of ODE's; analytic, singular point and basic recurrence relations. **10 Hrs.**

UNIT-III

Complex Analysis: Introduction to functions of complex variables. Definitions- limit, continuity and differentiability. Analytic functions. Cauchy - Riemann equations in Cartesian and polar forms (no proof). Construction of analytic function using Milne-Thomson method. Harmonic functions – Problems. Applications of analytic function to flow problems. Conformal transformation – Definitions and Discussion of Transformations: w=z^2, w =e^z and w= $z + \frac{1}{z}\,(z \neq 0)$ and related problems. Bilinear transformation.

Self-Study Component: Derivation of Cauchy- Riemann equation in Cartesian and polar form.

11 Hrs.

UNIT-IV

Complex integration: complex line integrals. Cauchy theorem, Cauchy integral formula. Taylor's and Laurent's series (Statements only) and illustrative examples. Singularities, poles and residues. Cauchy residue theorem (statement only). Illustrative examples. Statistics: Brief review of measures of central tendency and dispersion. Moments, skewness and kurtosis. Curve fitting – least square method: $y = a + bx; y = ax^b, y = ab^x$ *and* $y = ax^2 + bx + c.$ Correlation and **regression**.

Self-Study Component: Derivation of Cauchy theorem, Cauchy integral formula and Cauchy's residue theorem.Fit an equation of the **curves** of the type : $y = ae^{bx}$, **11 Hrs.**

UNIT – V

ProbabilityTheory: Brief review of elementary probability theory. Random variables (discrete and continuous)-Introduction to probability distributions- probability mass/density functions and cumulative probability density functions – Illustrative examples. Discrete probability distributions-Binomial and Poisson's distributions; Continuous probability distributions - exponential and normal distributions. (No derivation of mean and variance). Illustrative examples from engineering and industrial fields.

Joint probability distributions and Markov chains: Concept of joint probability. Joint probability Distributions of discrete random variables. Expectation, Covariance, correlation coefficient – illustrative Examples. Continuous joint probability distributions. Probability vectors, stochastic matrices. Fixed pointand regular stochastic matrices.

Self-Study Component: Basic definitions of probability and problems up to Bayes' theorem.

Derivation of Mean and SD of Binomial &Poisson distribution**.**

10 Hrs.

Text Books:

1. **Higher Engineering Mathematics:** B.S. Grewal, Khanna Publishers, New Delhi, 42nd Ed. 2012.
2. **Advanced Engineering Mathematics: -** E. Kreyszig, John Wiley & Sons, 10th Ed., 2011

References:

1. **T. Veerarajan:** Engineering Mathematics, Tata McGraw-Hill Pub., 2003.
2. **Introductory Methods of Numerical Analysis: -** S.S.Sastry, PHI, 3rd Ed.2000.
3. **Linear Algebra and its applications: -** David C.Lay, Pearson Education Ltd., 3rd Edition, 2003.
4. **Seymour Lipschutz:Probability:-,** Schaum's outline series, McGraw-Hill Pub., 2nd Ed, 2002.
5. **Higher Engineering Mathematics: -** B.V. RAMANA, McGraw Hill Education, 200

Course Articulation Matrix

Mapping of Course Outcomes (CO) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

Sem: 4	Course code : P18MA41	Title : Engineering Mathematics –IV											
CO's	Statement	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO-1	Solve algebraic, transcendental and ordinary differential equations arising in various engineering flow and design data problems, using numerical techniques along with physical interpretation of the solutions associated with initial/boundary conditions (UNIT-I)	2	2	-	-	-	-	-	-	-	-	-	-
CO-2	Learn logical thinking and analytical /geometrical skills in linear algebra through vector spaces, basis, dimension and linear transformations along with construction a matrix of linear transformations with respect change of Bases of same or different dimensions. Understand iterative methods in linear algebra such as Gauss-Jacobi, Gauss -Seidel, Relaxation and Power method and their practical utility in engineering fields(UNIT-II)	3	3	-	-	-	-	-	-	-	-	-	-
CO-3	Understand the basics of functions of complex variables, analytic functions, conformal and bilinear transformations, complex integration, line / surface / volume integrals and residue theorems with their scientific / engineering importance (UNIT-III)	3	3	-	-	-	-	-	-	-	-	-	-
CO-4	Apply the basic tools of statistics to understand curve fitting, moments, skewness, kurtosis, correlation and regression, for frequency distributions; explore the idea of probability, probability distributions, required in the analysis of engineering experiments (UNIT-IV)	2	2	-	-	-	-	-	-	-	-	-	-
CO-5	Apply the basic concepts of probability distributions to understand concept of joint probability and to find expectation covariance, correlation coefficient etc. and to understand probability vector, stochastic matrix etc. Obtain series solution of essential ODE's such as Bessel's and Legendre's differential equations and understand their scientific/engineering utility (UNIT-V)	3	3	-	-	-	-	-	-	-	-	-	-

Note: - Each unit contains two full questions of 20 marks each. Students are required to Answer any **five** full questions choosing at least **one** full question from each unit.

Course Title : Database Management System			
Course Code : P18IS42	**Semester : IV**	**L:T:P:H : 3:0:0:4**	**Credits: 3**
Contact Period: Lecture: 52 Hr, Exam: 3 Hr		**Weightage: CIE:50; SEE:50**	

Course Learning Objectives (CLOs)

This course aims to

1. State the importance of DBMS and explain how DBMS is better than traditional File Processing Systems.
2. Analyze the basic structure of Database and recognize the different views of the database.
3. Draw and Investigate Data Flow and Entity Relationship Diagrams.
4. Analyze and use Relational Data Model, while comparing with other data models.
5. Formulate data retrieval queries in SQL and the Relational Algebra and Calculus.
6. Understand and explain the terms like Transaction Processing and Concurrency Control.
7. Understand types of database failure and recovery

UNIT – I

Introductory concepts of DBMS: Introduction and example of DBMS, Characteristics of the database, Actors on the scene and Workers behind the scene in DBMS, Database System-Concepts and Architecture: Data models, Schemas and Instances, Three Schema architecture and Data Independence, Database language and interfaces, The Database System Environment.

Data modeling using the Entity Relationship Model: using high level conceptual data models for database design, Entity, Entity types, Entity sets, Attributes and keys, Relationship types, Relationship sets, Roles and structural constraints, an example Database Application, Refining the ER design for the company database.

Self Study: ER diagrams, naming conventions and design issues, Advantages of Using the DBMS Approach. **10 Hrs**

UNIT -II

Relational Data Model and Relational Database Constraints: Relational Model concepts, Relational Model constraints and Relational Database schemas, Update operations, Transactions, dealing with constraint violations. Basic SQL: Data Definition and Data Types; Specifying basic constraints in SQL, Basic Retrieval Queries in SQL, INSERT, DELETE, and UPDATE Statements in SQL, More Complex SQL Retrieval Queries, Schema Change Statements in SQL,

Self study: Specifying Constraints as Assertions and Actions as Triggers, Views (Virtual Tables) in SQL. **11 Hrs**

UNIT-III

Relational Algebra and Relational Calculus: Unary relational operations, Relational Algebra operations from set theory, Binary relational operations, Additional relational operations, example of queries in relational algebra. Database Design-1: Functional Dependencies; Normal Forms Based on Primary Keys; General Definitions of Second and Third Normal Forms;

Self study: Informal Design Guidelines for Relation Schemas **10 Hrs**

UNIT-IV

Database Design-2: Boyce-Codd Normal Form; Multi valued Dependencies and Fourth Normal Form; Join Dependencies and Fifth Normal Form.

NoSQL Data Management: introduction to NoSQL, why NoSQL? Characteristics of NoSQL, History of NoSQL, Types of NoSQL data models, schema-less databases, Materialized Views, Distribution Models, CAP Theorem, Sharding

Self study: Difference between SQL and NoSQL. **10 Hrs**

UNIT-V

Transaction processing concepts: Introduction to Transaction processing; Transactions and System concepts; Desirable properties of transactions; Characterizing Schedules based on Serializability. Concurrency control and recovery techniques: Two-phase locking techniques for concurrency control, concurrency control based on timestamp ordering; Database Recovery Techniques: NO-UNDO/REDO Recovery Based on Deferred Update, Recovery techniques based on immediate update.

Self study: Characterizing Schedules based on Recoverability, shadow paging, The ARIES Recovery Algorithm **11 Hrs**

Text Books:

1. Fundamentals of Database Systems – Elmasri and Navathe, 6th Edition, Addison-Wesley, 2011.
2. DT Editorial Services, "Black Book- Big Data (Covers Hadoop 2, MapReduce, Hive, Yarn, PIG, R, Data visualization)", Dream tech Press edition 2016. (Unit-4)

Reference Books:

1. Data Base System Concepts – Silberschatz, Korth and Sudharshan, 5th Edition, Mc-GrawHill, 2006
2. 2. An Introduction to Database Systems – C.J. Date, A. Kannan, S.Swamynatham, 8th Edition, Pearson Education, 2006.

Course Outcomes

After completing the course, the students will be able to

1. Understand and explore the needs and concepts of relational database management, non-relational database, transaction processing and related relational database facilities.
2. Apply the knowledge of logical database design principles to real time issues.
3. Analyze and design relational and document-based data model concepts.
4. Develop applications using Relational database and NoSQL database.

Course Articulation Matrix (CAM)															
Course Outcomes	**Program Outcomes (PO's)**												**PSO's**		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO 1	1	1							1	1		2	1		
CO 2		1	1	1	1		1		1	2		2	2		
CO 3	2	2	2	2	2		1		2	3	1	1	2		
CO 4	2	2	3	2	3			2	3	3	2	1	2		

Course title: Finite Automata and Formal language			
Course Code: P18IS43	**Semester: IV**	**L-T-P-H : 3-0-0-4**	**Credit:3**
Contact Period: Lecture: 52 Hrs, Exam: 3 Hrs		**Weightage: CIE:50%, SEE: 50%**	

Prerequisite: Knowledge in Discrete mathematics and in programming

Course Learning Objectives (CLOs)

This course aims to

1. Design regular expression and finite automata.
2. Design NFA-Λ, convert between finite automata and regular expressions for regular languages and apply the pumping lemma for regular languages to determine if a language is regular.
3. Design grammars for various languages and Demonstrate that grammar are ambiguous.
4. Design push-down automata from grammars and Design grammars from push-down automata and
5. Design Turing machines for a given language and consider other ways to characterize them.

Relevance of the Course: This course is one of the foundation courses in Information Science and Engineering program, It helps the student to design an abstract machine to accept any languages, Which will helps the student to understand the design process need to be followed for any problem.

Course Content

UNIT-I

Basic Mathematical Objects: Languages

Regular Expression and Finite Automata: Regular Languages and Regular Expressions, The memory required to recognize a language, Finite Automata, Distinguishing one string from another.

Nondeterminism and Kleene's theorem: Nondeterministic Finite Automata

Self study: Unions, Intersections and Complements **10 Hrs**

UNIT-II

Nondeterminism and Kleene's theorem (Cont...): Nondeterministic Finite Automata with Λ-Transitions, Kleene's Theorem

Regular and Nonregular Languages: A criterion for regularity, Minimal Finite Automata, The Pumping Lemma for Regular Languages.

Self study: Decision Problems **11 Hrs**

UNIT-III

Context-Free Grammars: Examples and Definitions, More Examples, Regular Grammars, Derivation Trees and Ambiguity, Simplified Forms and Normal Forms.

Context-Free and Non- Context-Free Languages: The Pumping Lemma for Context-Free Languages, Intersections and Complements of Context-Free Languages.

Self study: An Unambiguous CFG for Algebraic Expressions **11 Hrs**

UNIT-IV

Pushdown Automata: Introduction, The definition of a pushdown Automata, Deterministic Pushdown Automata, A PDA corresponding to a given Context-Free Grammar, A Context-Free Grammar corresponding to a given PDA, Parsing.

Self study: Top Down Parsing **10 Hrs**

UNIT-V

Turing Machine: Definitions and Examples, Counting a Partial function with a Turing Machine, Combining Turing Machine, Variations of Turing Machine: Multitape TM's, Nondeterministic Turing Machine, Universal Turing Machine.

Self study: Recursively Enumerable and Recursive, A Non recursive languages and an Unsolvable Problem. **10 Hrs**

Text books:

1. John C Martin, Introduction to Languages and The Theory of Computations, 3rd Edition, Tata McGraw-Hill Education, 2016.

Referenced books:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman: Introduction to Automata Theory, Languages and Computation,3rd Edition, Pearson education.
2. Raymond Greenlaw, H.James Hoover: Fundamentals of the Theory of Computation, Principles and Practice, Morgan Kaufmann.
3. Daniel I.A. Cohen: Introduction to Computer Theory, 2nd Edition, John Wiley & Sons.
4. A. M. Padma Reddy: Finite Automata and Formal Languages, Pearson

Course Outcomes

After learning all the units of the course, the student is able to

1. Construct regular expression and finite automata
2. Analyze regular Language
3. Design context free grammars
4. Design push down automata
5. Design Turing machine

Course Articulation Matrix (CAM)															
Course Outcomes	Program Outcomes (PO's)												PSO's		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO 1	2	3	3	2									3		
CO 2	3	3	3	2									3		2
CO 3	3	3	3	3									3		
CO 4	3	3	3	2									3		
CO 5	3	3	3	2									3		

Course title: Design and Analysis of Algorithms			
Course Code: P18IS44	Semester: IV	L-T-P-H :3-0-0-4	Credit:3
Contact Period: Lecture: 52 Hrs, Exam: 3 Hrs		Weightage: CIE:50%, SEE: 50%	

Prerequisites: Students should have knowledge of Programming language and Data structures.
Students should know the usage of summation formulae, recurrences in mathematics.

Course Learning Objectives

1. Understand and use asymptotic notations to analyze the performance of algorithms.
2. Understand and analyze the design of algorithms using Brute force, Decrease & Conquer.
3. Understand and analyze the design of algorithms using Divide & Conquer, Transform & Conquer.
4. Understand and analyze the design of algorithms using Space and Time Tradeoffs, Dynamic Programming.
5. Understand and analyze the design of algorithms using Greedy technique, Backtracking, Branch & Bound techniques.

Course Contents

UNIT - I

Introduction: What is an Algorithm? Fundamentals of Algorithmic problem solving, Important Problem Types, Fundamentals Data Structures. Fundamentals of Analysis of Algorithm Efficiency: Analysis Framework, Asymptotic Notations and Basic Efficiency Classes, Mathematical Analysis of Non recursive and Recursive Algorithms, Example-Fibonacci Numbers.

Self-Study Component: Empirical Analysis of Algorithms. **10 Hrs**

UNIT - II

Brute Force: Selection Sort and Bubble sort, Sequential Search and Brute-Force String Matching, Exhaustive Search. Decrease and Conquer: Insertion Sort, Depth First Search, Breadth First Search, Topological Sorting.

Self-Study Component: Fake-coin problem. **10 Hrs**

UNIT - III

Divide and Conquer: Merge sort, Quick Sort, Binary Search, Binary tree traversals and related properties, Multiplication of Large integers and Strassen' Matrix Multiplication.

Transform and Conquer: Balanced Search Trees, Heaps and Heap sort.

Self-Study Component: Gaussian Elimination, presorting. **10 Hrs**

UNIT - IV

Space and Time Tradeoffs: Sorting by counting, Input Enhancement in String Matching, Hashing, B-Trees. Dynamic Programming: Computing a Binomial Coefficient, Warshall's and Floyd's Algorithms, the Knapsack Problem and Memory functions.

Self-Study component: B-Trees. **11 Hrs**

UNIT - V

Greedy Technique: Prim’s Algorithm, Kruskal’s Algorithm, Djikstra’s Algorithm, Huffman Trees. Limitations of Algorithm Power: P, NP and NP- Complete Problems. Coping with the Limitations of Algorithm Power: Backtracking: n-Queens Problem, Subset-Sum Problem, Branch and Bound: Knapsack Problem.

Self-Study component: Lower Bound Arguments, Decision trees. **11 Hrs**

Text Book:

1. Introduction to the Design & Analysis of Algorithms, Anany Levitin, 3rd Edition, Pearson education, 2011

Reference Books:

1. Computer Algorithms by Horowitz E., Sahini S.,Rajasekaran S., Galgotia Publications, 2001
2. Introduction to Algorithms, Thomas H., Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 2nd Edition, PHI, 2006

Course Outcomes

After learning all the units of the course, the student is able to

1. Use asymptotic notations to analyze the performance of algorithms.
2. Analyze the design of algorithms using Brute force, Decrease & Conquer.
3. Analyze the design of algorithms using Divide & Conquer, Transform & Conquer.
4. Analyze the design of algorithms using Space and Time Tradeoffs, Dynamic Programming.
5. Analyze the design of algorithms using Greedy technique, Backtracking, Branch & Bound techniques.

Course Articulation Matrix (CAM)															
Course Outcomes	**Program Outcomes (PO’s)**											**PSO’s**			
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO 1	3	2		1									1		
CO 2	2	2	2	1									1	1	1
CO 3	2	2	2	1									1	1	1
CO 4	2	2	2	1									1	1	1
CO 5	2	2	2	1									1	1	1

Course Title: Software Engineering			
Course Code: P18IS45	**Semester : IV**	**L- T – P - H : 3 – 0 – 0 - 4**	**Credit : 3**
Contact period : Lecture: 52 Hrs, Exam:3 hrs		**Weightage: CIE: 50;SEE:50**	

Course learning objectives

This course aims to

1. Study a body of knowledge relating to Software Engineering, Software reengineering, and maintenance.
2. Understand the principles of large scale software systems, and the processes that are used to build them.
3. Use tools and techniques for producing application software solutions from informal and semi-formal problem specifications;
4. Acquire and develop many valuable skills such as the ability to use computer aided software
5. Evaluate requirements for a software system
6. Apply the process of analysis and design using object oriented approach.
7. Communicate to others the progress of the system development and the contents of the design by means of reports and presentations.
8. Recognize current trends in the area of software engineering
9. Identify the processes, techniques and deliverables that are associated with requirement engineering including system requirement and system modeling
10. Identify the importance of testing in assuring the quality of software with an understanding of managing risks during the progress of the project.

Course contents

UNIT-I

Overview, and Requirements

Introduction: FAQ's about software engineering, Professional and ethical responsibility; software process models, process iteration, software specification, software design and implementation, software validation, software evaluation; Software Requirements: Functional and Non-functional requirements; User requirements; System requirements; the software requirements document; requirements engineering processes: feasibility studies, requirements elicitation and analysis, requirement validation and management; system models: context models, behavioral model, data models, object models, workbenches; software prototyping: prototyping in the software process, rapid prototyping techniques, user interface prototyping.

Self Study Component: CASE Tools. **12Hrs**

UNIT-II

Software Design

Architectural Design: system structuring, control models, modular decomposition, domain-specific architectures; object oriented design: Objects and Object Classes, An Object-Oriented design process.

Self Study Component: Design evolution. **10 Hrs**

UNIT-III

Critical System, Verification and Validation

Dependability: critical systems, availability and reliability, safety, security; critical system specification, verification and validation: Verification and Validation: Planning; Software inspections; Automated static analysis, clean room software development; software testing: defect testing, integration testing, object oriented testing, testing workbenches.

Self Study Component: V&V **10Hrs**

UNIT-IV

Management

Managing People: limits to thinking, group working, choosing and keeping people, the people capability maturity model; software cost estimation: productivity, estimation techniques, algorithmic cost modeling, project duration and staffing; quality management: quality assurance and standards, quality planning, quality control.

Self Study Component: Software Cost Estimation. **10 Hrs**

UNIT- V

Evolution

software change: program evolution dynamics, software maintenance, architectural evolution; software Re-engineering: source code translation, reverse engineering, program structure improvement, program modularization, data re-engineering.

Self Study Component: Reverse Engineering Process. **10 Hrs**

Text book:

1. Software Engineering– Ian Somerville, 8th Edition, Pearson Education, 2007.

Reference books:

1. Software Engineering: A Practitioners Approach **-** Roger S. Pressman, 7th Edition,McGraw-Hill, 2007.
2. Software Engineering Theory and Practice -Shari Lawrence Pfleeger, Joanne M. Atlee, 3rd Edition, Pearson Education, 2006.
3. Software Engineering Principles and Practice –Waman S Jawadekar, Tata McGraw Hill, 2004
4. Software Engineering – Pankaj Jalote, Tata Mc Graw Hill.

Course outcomes

After learning all the units of the course, the student is able to

1. Demonstrate an understanding of the principles and techniques of Software Engineering
2. Analyze the various steps involved in the design process and the different design approaches which include function-oriented design and object-oriented design
3. Understand the activities in project management, requirement engineering process and to identify the different types of system models
4. Apply the knowledge of design engineering in software development
5. Provide an understanding of the principles of software engineering in a broader system context and the notions of software engineering process and management.

Course Articulation Matrix (CAM)

Course Outcomes	**Program Outcomes (PO's)**												**PSO's**		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO 1	2	1	2			1	1						1	1	1
CO 2	2		2			1							1		1
CO 3	2	1									2				1
CO 4	2		2										1		1
CO 5	1	1						1			1		1		1

Course title: AVR Microcontroller (FC-II)			
Course Code: P18IS46	**Semester: IV**	**L-T-P-H : 3-0-0-4**	**Credit: 4**
Contact Period: Lecture: 52 Hrs, Exam: 3 Hrs		**Weightage: CIE:50%, SEE: 50%**	

Course learning objectives

This course aims to

1. Describe the architecture and organization of microprocessors and microcontrollers.
2. Explain main features of the AVR microcontrollers.
3. Examine the Arduino open source hardware and software systems.
4. Develop sketches to perform simple input and output operations.
5. Describe the AVR hardware and timer interrupts.
6. Describe diverse applications of AVR microcontrollers.

Course Content

UNIT-I

Microcontrollers and embedded processors: microcontroller versus microprocessors, criteria for choosing microcontroller. Overview of the AVR family, general purpose registers in AVR, the AVR data memory, instructions with data memory, AVR status register, AVR data format and directive, AVR assembly instruction format, the program counter and program ROM space in AVR. **11 Hrs**

Self-study components: Introduction to computing- numbering and coding system.

UNIT-II

Branch instruction and looping in AVR: looping in AVR, other conditional jumps, unconditional branch instruction. Call instructions and Stack: CALL, RCALL, and ICALL. I/O port programming in AVR, I/O Bit-manipulation programming. AVR time delay: time delay calculation for AVR. **10 Hrs**

Self-study components: AVR time delay: time delay calculation for AVR.

UNIT-III

Arithmetic instructions: Addition, subtraction, multiplication and division of unsigned numbers, signed number concepts and arithmetic operations. Logical and compare instructions. Rotate and Shift instructions and data serialization. **11 Hrs**

Self-study components: Data serialization.

UNIT-IV

BCD and ASCII conversion, Introducing to some more assembler directives, Register and Direct addressing mode, register and indirect addressing mode, look-up table and table processing,. Macros. Timer 0 programming. **10 Hrs**

Self-study components: look-up table and table processing.

UNIT-V

AVR programming in C: Data types and time delays in C, I/O programming in C, Logic operations in C, Data conversion programs in C, Data serialization in C, Memory allocations in C. Keypad interfacing: Interfacing the keypad to AVR. **10 Hrs**

Self-study components: Data serialization in C.

Text book:

1. The AVR microcontroller and embedded system using assembly and C by Muhammad Ali Mazidi, Sarmad Naimi, Sepehr Naimi.

Reference book:

1. programming and interfacing ATMEL's AVRs by Thomas grace.

Course outcomes

After learning all the units of the course, the student is able to

1. Compare and contrast Microprocessor and Microcontroller
2. Code simple AVR assembly language instructions.
3. Code assembly language to use the ports for input or output
4. Code c program for time delay, logical and arithmetic operations and fro data serialization.
5. Interfacing the keypad to the AVR using assembly and C

Course Articulation Matrix (CAM)															
Course Outcomes	**Program Outcomes (PO's)**												**PSO's**		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO 1	3	3	3	1					2		2	2		3	
CO 2	3	2	3	1					2		2	2		3	
CO 3	3	3	3		2				2		2			3	
CO 4	2	2	2						2		2			2	
CO 5	2	1	1											2	

Course title: Design and Analysis of Algorithms Lab			
Course Code: P18ISL47	Semester: IV	L-T-P-H : 0-0-3-3	Credit: 1.5
Contact Period: Lecture: 36 Hrs, Exam: 3 Hrs		Weightage: CIE:50%, SEE: 50%	

Prerequisites: Programming Knowledge of C/ Java**.**

Course Learning Objectives (CLOs)

This course aims to

1. Distinguish between the basic concepts of time and space complexity and various design strategies.
2. Apply the methodologies of Brute force and Divide and conquer and evaluate the complexity.
3. Solve a problem using Transform and conquer algorithms and evaluate its correctness.
4. Formulate the time-complexity analysis for Dynamic programming and greedy techniques.
5. Apply Analyze and Design Branch and Bound techniques.

Course Content

NOTE: Design and Implement programs for the following Algorithms using C or Java

1. Programs to implement Brute Force method.
2. Programs to implement Divide and Conquer method.
3. Programs to implement Decrease and conquer method.
4. Programs to implement Space and Time Tradeoffs.
5. Program to implement Dynamic Programming Method.
6. Program to implement Greedy method.

Course Outcomes

1. Implement the algorithms based on various design techniques.
2. Analyze the efficiency of various algorithms
3. Produce substantial written documentation

Course Articulation Matrix (CAM)															
Course Outcomes	Program Outcomes (PO's)											PSO's		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO 1	1	2	1	1										2	
CO 2	2	2	2	1										1	
CO 3	1	1	1	1										1	

Course Title : Java Programming Lab			
Course Code : P18ISL48	**Semester : IV**	**L:T:P:H : 0:0:3:3**	**Credits: 1.5**
Contact Period: Lecture: 39 Hr, Exam: 3 Hr	**Weightage: CIE:50; SEE:50**		

Prerequisites: Computer concepts and C Programming

Course Learning Objectives (CLOs)

This course aims to

1. Apply object oriented concepts to real world problems in Java
2. Apply the concepts of inheritance and interfaces in Java
3. Demonstrate usage of packages, string handling and exception handling in Java
4. Illustrate need for multi-threading, generics and file handling in Java

Course Content

1. Programs to implement basic features of Java
2. Programs to implement concepts of classes, objects and methods
3. Programs to implement inheritance and interface features
4. Programs to implement packages, string handling and exception handling
5. Programs to implement multi-threaded programming, generics and file handling

Course outcomes

1. Implement Java programs using object oriented concepts
2. Demonstrate features of Java involving Interfaces, Packages, String handling and Exception Handling
3. Apply the concepts of multi-threading, generics and files in Java

Course Articulation Matrix (CAM)

Course Outcomes	**Program Outcomes (PO's)**												**PSO's**		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO 1	2	2	2										2		
CO 2	2	2	2										2		
CO 3	2	2	2										2		

Course Title : Aptitude and Reasoning Development - Intermediate (ARDI)			
Course Code : P18HU49	Semester : IV	L - T - P : 2-0 - 0 - 2	Credits: 01
Contact Period: Lecture: 32 Hr. Exam: 3 Hr	Weightage: CIE:50%;SEE:50%		

Prerequisites: ARDB

Course Learning Objectives (CLOs)

This course aims to

1. Explain proportionality rule, average speed, relative speed and concepts in circular track.
2. Explain the application of time, speed distance in solving problems related to races, trains, boats and streams, and clocks.
3. Explain different methods to calculate number of smaller cubes, the date and the day of any year and the concepts of clocks.
4. Explain the methodology of strengthening or weakening the given statement.
5. Explain application of Venn diagrams in solving set theory problems.
6. Explains the concept of syllogism and provides the methodology to tackle the problems.
7. Describes all the important properties of triangle, polygons, circle and other geometrical figures and solve application based questions.
8. Describe the properties of cone, cylinder, sphere, cube and cuboid and solve the application based questions.
9. Differentiates between individual work and group work.
10. Integrates the concept of individual work in solving problems related to pipes and cisterns

Relevance of the course:

4th semester deals with more of quantitative aptitude. It is the intermediate level of aptitude which involves modules like Time speed distance. Time and work, set theory. This course also touches upon logical abilities through modules like cubes and Calendars.

Course Content

UNIT – I

Time, Speed and Distance:

Concept of motion and mathematical representation of motion, The rule of proportionality, Conversion between kmph to m/s, Concept of average speed and its application in different scenarios, Relative speed– Importance, application and observation in day to day life, same direction and opposite direction, An application of allegation in Time speed and distance, Trains– Different scenarios. Boats and streams– resultant speed, upstream and downstream concept. Circular motion– Two or three bodies meeting at the starting point or anywhere in the track. Races– Concept of head start, solving problems under different constraints. Application of solving problems under Clocks.

SSC: Basic relation between the 3 different quantities. Conversions between different UNITs of measurement. Speed and velocity. **6 Hrs**

UNIT – II

Cubes, Clocks & Calendars:

Cubes: Number of faces, vertices and edges. Colored cubes. Number of colored faces and the formulae to find-out the same. Problems on cubes.

Clocks & Calendars: Minute spaces. Hour hand and minute hand. Angle between the hands. Relative speed. Faulty clocks. Time gained or lost by the clock. Odd days. Leap year. Ordinary year. Counting of odd days. Problems on clocks and calendars.

Self-study Component- Knowledge about shapes and dimensions, Area and volume. Leap year, number of days. Important dates. **8 Hrs**

UNIT – III

Set theory and Venn diagram: Set builder form, Tabular form, Venn diagram, Types of sets, Operation of sets using venn diagram, Important properties, Algebraic laws of sets, Maxima and minima in set operation, Venn diagram for four sets.

Syllogism: Meaning of syllogisms, Format of problems and standard qualifiers, Concept of distribution, Standard question pattern, Application of venn diagram to solve problems. **Logical Venn diagrams:** Analysis of the given problem and solve it.

Self-study Component- Basics about sets, operations using venn diagram. Basic applications.

6 Hrs

UNIT – IV

Geometry and Mensuration:

Theory, straight lines, triangles– theorems, area, lines inside triangle and geometric centre, Special property of an equilateral triangle, Application of Pythagoras theorem, Congruency and similarity of triangles, Basic proportionality theorem, Polygons, Quadrilaterals, Trapezium, Parallelogram, Rectangle, Rhombus, Square, Division of polygons, Circumscribed and Inscribed polygons, Concyclic points concept, Cyclic quadrilateral, Circle– Radius, Area and perimeter, Arc, Chord, Sector, Segment, Tangent, Secant, Area of common region Solid figures– Introduction, Classification of a solid, Net of a solid, Cuboid, Cube, Right cylinder, Pyramid– right pyramid, triangular pyramid, Cone– frustum of a cone, Sphere, Combination of solid.

Co-ordinate geometry:

Cartesian coordinate geometry– rectangular coordinate axis, distance formula, Section formula, Area of a triangle, Centre of gravity or Centroid of a triangle, In-centre of a triangle, Circumcentre of a triangle, Orthocentre of a triangle, Collinearity of three points, Slope of a line, Different forms of equations of a straight line, Perpendicularity and parallelism, Length of perpendicular.

Self-study Component-Basics of geometry, formula, dimensions, shapes. Different types of lines. Example – parallel, intersecting etc… **8 Hrs**

UNIT – V

Time and Work:

Relationship between time and work. Importance of efficiency, Conventional method of solving problems, L.C.M method, Negative work, The specific case of building a wall, Group work, Constant product rule, When work is not constant, Pipes and cistern– Similarity of logic.

SSC: LCM methods, basic arithmetic. Fractions and efficiency. **4 Hrs**

Reference Books:

1. The Trachtenberg speed system of basic mathematics, published by Rupa publications.
2. CAT Mathematics by AbhijithGuha. published by PHI learning private limited.
3. Quantitative aptitude by Dr. R. S Agarwal, published by S.Chand private limited.
4. Verbal reasoning by Dr. R. S Agarwal , published by S. Chand private limited.
5. Quantitative aptitude for CAT by Arun Sharma, published by McGraw Hill publication.
6. Analytical reasoning by M.K Pandey BSC PUBLISHING.CO.PVT.LTD

Course Outcomes (CO)

After learning all the UNITs of the course, the student is able to:

1. Solve problems of higher difficulty level with ease in the following topics– Time, speed and distance and Geometry.
2. Analyze the number of colored faces in a cube when it is cut into different number of pieces and solve the problems under clocks and calendars.
3. Apply the concept of L.C.M in the module time and work to solve the problems with comprehension.
4. Analyze the concepts in Co-ordinate geometry by spatial visualization.
5. Interpret the logic in the statements of syllogism by critical thinking and apply venn diagram for the effectives ways of deriving at the conclusion.
6. Determine the solutions for complicated problems of set theory using the concept of venn diagram.

Course Title : Additional Mathematics-II			
Course Code : P18MADIP41	Semester : IV	L :T:P:H : 4:0:0:4	Credits: 0
Contact Period: Lecture: 52 Hr.		Weightage: CIE:(Max 50 marks)	

(Mandatory Learning Course: **Common to All Branches)**

(A Bridge course for Diploma qualified students of IV Sem. B. E.)

UNIT –I

Linear Algebra: Introduction - Rank of matrix by elementary row operations - Echelon form of a matrix. Consistency of system of linear equations - Gauss elimination method. Gauss-Jordan and LU decomposition methods. Eigen values and eigen vectors of a square matrix..

Self-study Components: Application of Cayley-Hamilton theorem (without proof) to compute the inverse of a matrix-Examples

10 Hrs

UNIT –II

Higher order ODE's: Linear differential equations of second and higher order equations with constant coefficients. Homogeneous /non-homogeneous equations. Inverse differential operators. and variation of parameters. Solution of Cauchy's homogeneous linear equation and Legendre's linear differential equation.

Self-study Components: Method of undetermined coefficients **14 Hrs**

UNIT –III

Multiple Integrals: Double and triple integrals-region of integration. Evaluation of double integrals by change of order of integration.

Vector Integration: Vector Integration: Integration of vector functions. Concept of a line integrals, surface and volume integrals. Green's, Stokes's and Gauss theorems (without proof) problems.

Self-study Components: Orthogonal curvilinear coordinates. **10 Hrs**

UNIT –IV

Laplace transforms: Laplace transforms of elementary functions. Transforms of derivatives and integrals**,** transforms of periodic function and unit step function-Problems only. Inverse Laplace transforms: Definition of inverse Laplace transforms. Evaluation of Inverse transforms by standard methods.

Self-study Components: Application to solutions of linear differential equations and simultaneous differential equations. **12 Hrs**

UNIT –V

Probability: Introduction. Sample space and events. Axioms of probability. Addition and multiplication theorems. Conditional probability – illustrative examples.

Self-study Components: State and prove Bayes's theorem. **06 Hrs**

Text Book:

1. B.S. Grewal: Higher Engineering Mathematics, Khanna Publishers, New Delhi, 42nd Ed. 2012.

References:

1.E. Kreyszig: Advanced Engineering Mathematics, John Wiley & Sons, 6th Ed., 2007

2.N.P.Bali and Manish Goyal: Engineering Mathematics, Laxmi Publishers,7th Ed., 2007.