SYLLABUS

(With effect from 2017-18 Academic year)

ಪಠ್ಯಕ್ರಮ

(ಶೈಕ್ಷಣಿಕವರ್ಷ 2017–18)

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.P S Puttaswamy
Dean (Academic)
Professor
Dept. of EE & 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 30. 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. Scheme of Teaching and Examination 2017- 18

Sl. No.	Course Code	Course Title	Teaching Dept.	Hours/ Week L:T:P:H	Credits	Examination Marks		
						CIE	SEE	Total Marks
1.	P17MA31	Mathematics-III	Maths	3:2:0:5	4	50	50	100
2.	P17IS32	Digital Design	IS&E	4:0:0:4	3	50	50	100
3.	P17IS33	Data Structures and Algorithms	IS&E	4:0:0:4	4	50	50	100
4.	P17IS34	Discrete Mathematics & Applications	IS&E	4:0:0:4	4	50	50	100
5.	P17IS35	Computer Organization and Architecture	IS&E	4:0:0:4	4	50	50	100
6.	P17IS36	Object Oriented Programming with C++ and Java	IS&E	3:0:2:5	4	50	50	100
7.	P17ISL37	Data Structures Lab	IS&E	0:1:2:3	1.5	50	50	100
8.	P17ISL38	Digital Design Lab	IS&E	0:1:2:3	1.5	50	50	100
9	P17HUDIP39	Comprehensive Communication Development (CCD)	HS&M	2:0:0:2	[2]	[50]	[50]	[100]
10	P17HU39	** Aptitude and Reasoning Development – BEGINNER (ARDB)	HS&M	2:0:0:2	0	(50)	--	--
11.	P17HUDIP310	*Indian Constitution ,Human Rights & Professional Ethics	Human& Science	2:0:0:0	0	--	---	---
12.	P17MADIP31	* Additional Maths-I	Maths	4:0:0:4	0	--	---	---
Total					26[28]	400 [450]	400 [450]	800 [900]

IV Semester B.E. Scheme of Teaching and Examination 2017- 18

Sl.No	Course Code	Course Title	Teaching Dept.	Hours/ Week L:T:P:H	Credits	Examination Marks		
						CIE	SEE	Total Marks
1.	P17MAES41	Mathematics-IV	Maths	3:2:0:5	4	50	50	100
2.	P17IS42	Operating System	IS&E	3:0:2:5	4	50	50	100
3.	P17IS43	Finite Automata and Formal Language	IS&E	4:0:0:4	4	50	50	100
4.	P17IS44	Analysis and Design of Algorithms	IS&E	4:0:0:4	3	50	50	100
5.	P17IS45	Software Engineering	IS&E	4:0:0:4	4	50	50	100
6.	P17IS46	Microprocessors	IS&E	4:0:0:4	4	50	50	100
7.	P17ISL47	Analysis and Design of Algorithm Lab	IS&E	0:1:2:3	1.5	50	50	100
8.	P17ISL48	Microprocessors Lab	IS&E	0:1:2:3	1.5	50	50	100
9.	P17HU49	Aptitude and Reasoning Development – INTERMEDIATE (ARDI)	HS&M	2:0:0:2	1	50	50	100
10.	P17EVDIP410	*Environmental Studies	ENV	2:0:0:2	0	--	--	--
11.	P17MADIP41	* Additional Maths-II	Maths	4:0:0:4	0	--	--	--
Total					27	450	450	900
***Additional Mathematics-II & environmental Studies: lateral Entry Students shall have to pass these mandatory learning courses before completion of VI-semester.**								

Course Title: Engineering Mathematics-III			
Course Code: P17MA31	**Semester**: III	L – T – P – H : 3– 2 – 0 – 5	**Credits**:4
Contact Period - Lecture: 52Hrs.; Exam: 3Hrs.		**Weightage**: CIE: 50 %; SEE: 50%	

Prerequisites: The student should have acquired the knowledge of Engineering Mathematics-I & II of I and II semester B.E.

Course Learning Objectives (CLOs):
The course P17MA31 aims to:

1. Describe the concepts of elementary numerical analysis such as forward/backward finite differences, central differences, interpolation and extrapolation formulae, techniques of numerical differentiation and integration.
2. Explain the nature of periodic functions Fourier series of general as well as even /odd functions valid in full range/half-range periods along with applications through practical harmonic analysis.
3. Learn modeling in terms of partial differential equations and also, learn different exact/analytical methods of solving with special emphasis on interpretation of the solution of one-dimensional wave, heat and Laplace equations with given initial and boundary conditions in the context of various engineering and technological applications.

Relevance of the course:

Engineering Mathematics-III deals with the Numerical methods to solve interpolation and extrapolation problems in engineering field.

In Fourier series analyze engineering problems arising in control theory and fluid flow phenomena using harmonic analysis

Analyze the engineering problems arising in signals and systems, digital signal processing using Fourier transform techniques.

Z-transforms & 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

Partial Differential Equations (PDE's), order, degree and formation of PDE's and, to solve PDE's by various methods of solution.

One – dimensional wave and heat equation and Laplace's equation and physical significance of their solutions to the problems selected from engineering field

Course Content

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, Fourier series- Euler's formula, Dirichlet's conditions. Fourier series of discontinuous functions, Fourier series of even and odd functions. Change of interval- Fourier series of functions of arbitrary period. Half–range Fourier series expansions, Fourier series in complex form, 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 Fourier cosine transforms, properties. Inverse infinite Fourier and inverse Fourier sine & cosine transforms –Illustrative examples.

Difference equations and Z-transforms: Definition of Z-transforms- standard Z-transforms, 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's identities.related problems. **10 Hrs**

UNIT-V

Partial differential equations (PDE's):

Formation of PDE's. Solution of non-homogeneous PDE by 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: Charpit'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

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

Course Outcomes

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

CO-1. Apply forward, backward difference formulae and central differences formulae in solving interpolation- extrapolation problems in engineering field.

CO-2. Apply Numerical differentiation and integration rules in solving engineering where the handling of numerical methods is inevitable.

CO-3. Recognize the importance of Fourier series & Fourier transforms, difference equations and Z-transforms in the field of signals and systems, communication and network theory signal and image processing, control theory, flow & heat transfer and theory of elasticity.

CO-4. Learn modelling in terms of partial differential equations and also, learn different exact/analytical methods of solving with special emphasis on interpretation of the solution.

CO-5. Interpret the solution of one-dimensional wave, heat and Laplace equations with given initial and boundary conditions in the context of various engineering and technological applications.

CO-PO Mapping

Semester: 3 **Course code : P17MA31**	**Title : Engineering Mathematics –III**											
CO's	**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	1	**2**	-	-	-	-	-	-	-	-	-	-
CO-2	**2**	**2**	-	-	-	-	-	-	-	-	-	-
CO-3	**3**	**3**	-	-	-	-	-	-	-	-	-	-
CO-4	**2**	**3**	-	-	-	-	-	-	-	-	-	-
CO-5	**2**	**3**	-	-	-	-	-	-	-	-	-	-

Course title: Digital Design			
Course Code: P17IS32	**Semester:** III	**L-T-P-H** : 4-0-0-4	**Credit:**3
Contact Period: Lecture: 52 Hrs, Exam: 3 Hrs		**Weightage: CIE:**50%, SEE: 50%	

Prerequisites: Computer Concepts and C Programming

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. Analyse and design Asynchronous and Synchronous Sequential circuits.

Relevance of the course

This course assists the students in developing the knowledge and skills to design combinational and sequential digital systems using modern electronic design flow, and to apply relevant design considerations and constraints. Course initially gives the overview of concepts of Boolean algebra, Karnaugh maps, flip-flops, registers, and counters along with various logic families and comparison of their behavior and characteristics. It gives foundation in design and analysis of the operation of digital gates.

Course Content

Unit – I

Boolean Algebra and Combinational Networks & Simplification of Boolean Expressions:

Boolean Functions-Canonical Formulas- Manipulations of Boolean Formulas-Gates and Combinational Networks-Incomplete Boolean Functions and Don't Care Conditions-Additional Boolean operations and Gates Formulation of the simplification problem. Prime Implicants and Irredundant Disjunctive Expressions-Prime Implicates and Irredundant Conjunctive Expressions and 2, 3 and 4 variable K-Map and The Quine-Mccluskey Method-To find Prime Implicants using decimal method and binary method upto 5 variables.

Self study component: Boolean Algebra and Boolean Theorems **10 Hrs**

Unit – II

Arithmetic Circuits and Data Processing Circuits:

Binary Adders and Subtracters, Decimal Adders, Code converter, full Adder and Full Subtractor. Multiplexers and Demultiplexers, Decoders, Parity generator and checkers, Encoders.

Self study component: Magnitude Comparators. **10 Hrs**

Unit – III

Memory and Programmable logic, DAC Converters, VHDL Language:

Programmable Logic Devices- RAM, ROM, PROMs, PLA's, PAL. D/A Conversion and A/D Conversion: Variable, Resistor Networks, Binary Ladders, D/A Converters, D/A Accuracy and Resolution.

Self-study component: VHDL – Introduction to VHDL, describing data flow, behavioral, structural and mixed design style, Simulating design for arithmetic and combinational circuits. **11 Hrs**

Unit – IV

Flip-Flops and Registers:

Clock Waveform – Characteristics of ideal Clock Waveforms, synchronous operation, propagational delay time. The Basic Bistable Element-Latches-Timing Considerations-JK Master –Slave Flip-Flops, Pulse-Triggered Flip flops, Edge-Triggered Flip-Flops-Characteristic Equations. Conversions of Flip

Flop. Types of Registers, Serial In – Serial Out, Serial In – Parallel out, Parallel In – Serial Out, Parallel In – Parallel Out using JK or D Flip Flops. Applications of Shift Registers. Sequence detector and sequence generator.

Self study component: Ring counter, Johnson counter **11Hrs**

Unit – V

Asynchronous and Synchronous Counter:

Asynchronous counter – Up, Down, Up and Down Counter, Design of Synchronous up counter and down counter, decade counter. Counter design as a synthesis problem.

Design of Synchronous and Asynchronous Sequential Circuits:

Design of Synchronous Sequential Circuits- Model Selection, State Transition Diagram, State Synthesis Table, Design Equations and Circuit Diagram, State Reduction Technique. Analysis of Asynchronous Sequential Circuit, Problems with Asynchronous Sequential Circuits.

Self study component: Design of Asynchronous Sequential Circuit. **10 Hrs**

Text Book:

1. "Digital Principles & Design", Donald D Givone, 4th Reprint, Tata McGraw Hill.
2. "Digital Principles and Applications", Donald P Leach, Albert Paul Malvino and Goutham Saha, TMH, 7th Edition.

Reference Books:

1. "Fundamentals of Digital Logic Design with VHDL", Stephen Brown, Zvonko Vranesic, 2nd Edition, Tata McGraw Hill.
2. "Digital Systems Principles and Applications", Ronald J. Tocci, Neal S. Widmer, Gregory L. Moss, 10th Edition, Pearson Education.
3. "Digital Logic and Computer Design", M Morris Mano.

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. Analyse 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. Analyse 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: P17IS33	**Semester**: III	**L-T-P-H :** 4-0-0-4	**Credit:**4
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.-L4
2. Learn the different types of linked list and Design and implement operations on SLL, DLL, Circular SLL and Circular DLL using header nodes.-L6
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.-L6
4. Identify the different tree traversal techniques and Design and implement different tree traversal techniques using iteration and recursion. -L6
5. Learn the different sorting and searching techniques and Analyze the performance of the different sorting and searching techniques. -L4

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, Abstract Data Types-ADT for rational numbers, and ADT for varying length Character String, 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, Inorder Traversal of a Threaded Binary Tree.
Self study component: Inserting a Node into a Threaded Binary Tree. **12 Hrs**

Unit – V

Sorting Techniques: Insertion sort, Quick sort, Binary tree sort, Heap sort, Merge 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: P17IS34	**Semester**: 3	**L-T-P-H**:4-0-0-0	**Credit**:4
Contact Period: Lecture:52 Hrs	**Exam**: 3 Hrs	**Weightage**: CIE: 50, SEE: 50	

Course Learning Objectives (CLOs)

1 Learn and identify the fundamentals of logic and use of Quantifiers and understand the importance of induction principle
2 Learn and understand the basic concepts relations and functions and their representation
3 Develop the ability to identify different types of graphs and its properties
4 Learn and apply the concept of group theory
5 Apply the generalized principle of inclusion and exclusion theorem, concept of generating functions to solve the given problems of counting theory

Relevance of the Course: Is to develop the mathematical ability of the student in the application areas such as data structures, networks, computer graphics. The student gets introduced to applications in engineering, physical and life sciences, statistics and social sciences of converting the problem into a pictorial representation which makes the analysis easier. The student will be exposed to the idea that generates applications of mathematical expressions to the real time problems and develop ability to think in that direction.

Course contents

UNIT-I

Fundamentals of Logic: Logic Equivalence, the Laws of Logic, Logical Implication **-** Rules of Inference. **Quantifiers** and their uses: Quantifiers, Definitions and the Proofs of Theorems (Direct and indirect methods). **Properties of Integers**: Mathematical Induction, The Well Ordering Principle-Mathematical Induction in the Alternative form

Self-Study Component: Principles of counting **12 Hrs**

UNIT-II

Relations and Functions: Cartesian Products and Relations, Functions. Plain and One-to-One, Onto Functions – Stirling’s Numbers of the Second Kind, The Pigeon-hole Principle, Function Composition and Inverse Functions. Special functions-characteristic function, Permutation function, Hashing function Properties of Relations Computer Recognition: Zero-One Matrices and Directed Graphs, Partial Orders - Hasse Diagrams

Self-Study Component: Recurrence Relation **10 Hrs**

UNIT-III

Equivalence Relations and Partitions- Partitions induced by Equivalence relations. Topological sorting algorithm, totally ordered sets. Extremal elements, Lattices. **Groups**: Definitions, Elementary Properties, Homomorphism’s, Isomorphisms, and Cyclic Groups, Cosets, and Lagrange's Theorem.

SELF STUDY: coding Theory **10 Hrs**

UNIT- IV

Introduction to Graph Theory: Definitions and examples, finite and infinite graphs, sub graphs, Operations on graphs, complements, and Graph isomorphism. Applications: Vertex degree, Euler Trails and circuits, complements, Hamilton paths and cycles. Application of Graphs-Konigsberg Bridge problem, travelling salesmen problem, Utility problem, seating arrangement problem.

Self-Study Component: Planar Graphs **10Hrs**

UNIT- V

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. Generating functions: Introductory examples, Definition and examples– calculation techniques, partitions of integers

Self-Study Component: Matrix Representation of Graphs **10 Hrs**

Text Books :

1. Discrete and Combinatorial Mathematics, Ralph.P. Grimaldi& B.V. Ramana, 5thEdition, PHI/Pearson education
2. “Discrete Mathematical structures”, Dr D. S. Chandrashekariah, Prism.

Reference Books:

1. “Discrete Mathematics and its Applications ”, Kenneth H. Rosen, 6th Edition, McGraw Hill, 2007.
2. “Discrete Mathematical Structures: Theory and Applications ”, D.S. Malik and M.K. Sen, Thomson.
3. Discrete Mathematical structures”, Kolman Busby Ross , 5th edition , PHI.

Course Outcomes

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

1. Identify the quantifiers and their uses and learn the fundamentals of logic theory.-L3
2. Solve the problems using the concepts of relations and functions and Identify the different ways of representing relations-L3
3. Apply the concepts of group theory and coding theory to solve the given problem.-L5
4. Identify different parameters of graphs and its applications –L1
5. Apply and understand the principle of inclusion and exclusion, generating functions to solve the given problem. –L3

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	1	1	1	1	2	3	1	1	1	1	1		3	
CO 2	3	1	1	1	1	1	1	1	1	1	1	1	2	3	
CO 3	3	1	1	1	1	1	1	1	1	1	1	1	1	3	
CO 4	3	1	2	3	3	1	2	2	1	1	1	1	1	3	
CO 5	3	1	1	1	3	1	1	2	2	1	1	1			

Course title: Computer Organization & Architecture			
Course Code: P17IS35	**Semester:** III	**L-T-P-H:** 4-0-0-4	**Credit:**4
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 1

Basic Structure of Computers: Computer Types, Functional Units, Basic Operational Concepts, Number Representation and Arithmetic Operations, Character Representation, Performance **Instruction Set Architecture:** Memory Locations and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes

Self-Study Component: Assembly Language, Stacks. **11 Hrs**

UNIT II

Instruction Set Architecture (Cont'd): Subroutines, Additional Instructions, Dealing with 32-Bit Immediate Values **Basic Input/output**: Accessing I/O Devices, Interrupts. **Software:** Interaction between Assembly language and C Program, The Operating System.

Self-Study Component: CISC Instruction Sets, RISC and CISC Styles **10 Hrs**

UNIT III

Basic Processing Unit: Fundamental Concepts, Instruction Execution, Hardware Components, Instruction Fetch and Execution Steps, Control Signals, Hardwired Control. **Input/output Organization:** Bus Structure, Bus Operation, Arbitration, Interface Circuits

Self-Study Component: Interconnection Standards **11 Hrs**

UNIT IV

The Memory System: Basic Concepts, Semiconductor RAM Memories, Direct Memory Access, Memory Hierarchy, Cache Memories, Virtual Memory, Memory Management Requirements
Self-Study Component: Read-Only Memories, Performance Considerations. **10 Hrs**

UNIT V

Arithmetic: Multiplication of Unsigned Numbers, Multiplication of Signed Numbers, 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, Decimal-to-Binary Conversion.**10 Hrs**

Text Book:

1. Computer Organization and Embedded Systems, Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Naraig Manjikian ,6th Edition, McGraw Hill, 2014.
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:

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.

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 C++ and Java			
Course Code: P17IS36	**Semester** : III	**L- T- P-H** : 3-0-2-5	**Credits:**4
Contact period : 52 Hrs, Exam: 3 Hrs	**Weightage:** CIE: 50%; SEE: 50%		

Prerequisites: Computer Concepts and C Programming

Course Learning Objectives (CLOs)

This course aims to

1. Explain the need of using Object Oriented Programming in the real world applications and write C++ programs using classes and objects
2. Write C++ programs for automatic initialization of objects and destroy objects that are no longer required and discuss the mechanism of deriving new class from older classes through inheritance.
3. Develop methods to select appropriate member functions during run time and write C++ programs by overloading the given operators.
4. Write C++ programs to demonstrate the use of stream handling, templates and exception handling.
5. Implement features of Java programming to solve real world problems.

Relevance of the Course:

This course is one of the foundation courses in Information Science and Engineering program, which helps students to understand the importance of Object Oriented Programming (OOP) and one of the most important concepts in modern programming is OOP. It is important to learn OOP because you can assign an object of real life a variable in your program, give all the details about that object and define how you want to deal with them. For example, you might want to define a new way of subtracting/adding on some object of real life.

Course Content

UNIT - I

Introduction: A review of structures, Procedure Oriented programming systems, OOPS, Comparison of C++ with C, Console input/ Output in C++, variables in C++, Reference variable in C++, function prototyping, function overloading, Default values for formal arguments of functions, Inline functions.Class and Objects: Introduction to Classes and Objects, Member functions and Member data, Objects and functions, Objects and arrays, Namespace.
Self study component: Nested Inner Classes **10 Hrs**

UNIT - II

Dynamic Memory Management: Introduction, Dynamic Memory Allocation, Dynamic Memory Deallocation. Constructors and Destructors: Constructors, Destructors Inheritance: Introduction to Inheritance, Base Class and Derived class Pointers, Function Overriding, Base Class Initialization, The Protected Access Specifier, Deriving by Different Access Specifiers, Different Kinds of Inheritance.
Self study component: set_new_handler( ) Function, Philosophy of OOPS, Order of Invocation of Constructors and Destructors. **10 Hrs**

UNIT - III

Virtual Functions: The Need for Virtual Functions, Virtual Functions, the Mechanism of Virtual Functions, Pure Virtual Functions, Virtual Destructors and Virtual Constructors. Operator Overloading: Operator Overloading, Overloading the Various Operators, Overloading the Increment and the Decrement Operators (Prefix and Postfix), Overloading the Unary Minus and the Unary Plus Operator, Overloading the Arithmetic Operators. Overloading the Relational Operators, Overloading the Assignment Operator, Overloading the Insertion and Extraction Operators.
Self study component: Overloading Some Special Operators ([ ], ( ),–>). **10 Hrs**

UNIT - IV

Stream Handling: Streams, The Class Hierarchy of Handling Streams, Opening and Closing Files, Files as Objects of the fstream Class, File Pointer, Random Access to Files Templates: Introduction, Function Templates, Class Templates Exception Handling: Introduction, C-Style Handling of Error generating Codes, C++ Style Solution – the try/throw/catch Construct, Limitation of Exception Handling.

Self study component: Standard Template Library, Handling Derived-Class Exceptions **10 Hrs**

UNIT - V

Introduction to Java: Java AS A PROGRAMMING TOOL, Features of Java, Two control statement, using blocks of code, Lexical Issues, The java class Libraries, Data Types, Variables & Arrays: the primitive types, integers, floating-point types, Characters, Booleans, Arrays operators and arithmetic operators, the bitwise operators, relational operators, Boolean logical operators, the assignment operators, operator precedence, using parentheses, control statements: java's selection statements, iteration statements, jump statements

Classes & Inheritance: Class Fundaments, Declaring Objects, Assigning Object Reference Variables, introducing methods, constructors, the finalize() Method, A stack class, overloading methods, using objects as parameters, argument passing, returning objects, recursion, introducing access control, understanding static, introducing final, Arrays revisited, introducing nested & inner classes, exploring the string class, using command-line arguments, Inheritance: using super, creating a multilevel Hierarchy, when constructors are called.

Self study component: Method overriding, dynamic method dispatch, using abstract classes, using final with inheritance. **12 Hrs**

Text Books:

1. "Object-Oriented Programming with C++ 2/e", Sourav Sahay, Oxford University Press, 2012.
2. "The Complete Reference JAVA, J2SE", Herbert Schildt, 6th Edition, TMH, 2010.

Reference Books:

1. "C++ Primer", Stanley B. Lippman, Josee Lajoie, Barbara E. Moo, 5th Edition, Addison Wesley, 2012.
2. "The Complete Reference C++", Herbert Schildt, 4th Edition, TMH.
3. "Programming with Java", Balaguruswamy E, 4th Edition, 2010

Course Outcomes

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

1. Develop application programs using non-object oriented features of C++.
2. Demonstrate automatic initialization/de-initialization of objects and Inheritance using C++.
3. Apply the concepts of virtual functions and operator overloading for a given problem.
4. Implement the concepts of stream handling, templates and exception handling.
5. Develop programs using object oriented programming language like Java for a given scenario.

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		1								1		
CO 2	1		2		1										
CO 3	2		2		1								1		
CO 4	1		2		1								1		
CO 5	2		2		1								1	1	

Course title: Data Structure Lab			
Course Code: P17ISL37	**Semester:** III	**L-T-P-H:** 0 -1–2-3	**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: P17ISL38	**Semester:** III	**L-T-P-H:** 0-1–2- 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 : P17HU39	**Semester :** III	**L - T - P :** 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. Analyse 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
Self-study Component- 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.
Self-study Component- 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, 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, 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.

Self-study Component- 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.

Self-study Component-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.

Self-study Component-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.
Self-study Component-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. L2
2. Give examples for AP, GP and HP and differentiate between them. L1
3. Apply divisibility rules , power cycle method and evaluate the significance of the number system module. L2
4. Point out the errors in the problems concerning inequalities and solve simple equations and problems based on ratio, proportion and variation. L5
5. Solve the problems based on blood relations, directions and arrangement. L4

Course Articulation Matrix															
Course Outcomes		Program Outcomes												PSO	
		1	2	3	4	5	6	7	8	9	10	11	12	01	02
CO1	Solve mathematical calculations in less duration compared to the conventional method.	_	3	1	2	3	_	_	_	_	_	_	_	_	_
CO2	Give examples for AP, GP and HP and differentiate between them.	2	1	_	_	_	_	_	_	1	_	_	_	_	_
CO3	Apply divisibility rules, power cycle method and evaluate the significance of the number system module.	_	3	3	_	_	_	_	_	_	_	_	_	_	_
CO4	Point out the errors in the problems concerning inequalities and solve simple equations and problems based on ratio, proportion and variation.	_	_	_	_	_	_	_	2	_	2	2	_	_	_
CO5	Solve the problems based on blood relations	_	3	_	_	_	_	_	_	_	_	1	_	_	_

Course Title: Additional Mathematics-I			
Course Code : P1MADIP31	**Semester :** III	**L - T - P :** 4 - 0 - 0	**Credits:** NA
Contact Period: Lecture: 52 Hr**, Exam: --**		**Weightage :CIE:100% -** [P/NP]	

((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, De-Moivre's theorem (without proof). Roots of complex number - Simple problems.**Vector Algebra**: Scalar and vectors. Vectors addition and subtraction. Multiplication of vectors(Dot and Cross products). Scalar and vector triple products-simple problems. **12Hrs**

UNIT -II

Differential Calculus: Review of successive differentiation. Formulae for nth derivatives of standard functions- Liebnitz's theorem (without proof). 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. Application to Jacobians, errors & approximations.

10 Hrs

UNIT -III

Integral Calculus: Statement of reduction formulae for $sin^n x,\ cos^n x,\ and\ sin^m x cos^m x$ and evaluation of these with standard limits-Examples. Differentiation under integral sign(Integrals with constants limits)-Simple problems. Applications of integration to area, length of a given curve, volume and surface area of solids of revolution. **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). 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. 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.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.

<table>
<tr><td colspan="4"><b>Course Title</b>: Engineering Mathematics-IV
<b>(Common to E&C, E&E, CS&E and IS&E Branches)</b></td></tr>
<tr><td><b>Course Code:</b> P17MAES41</td><td><b>Semester:</b> IV</td><td><b>L – T – P – H :</b> 3 –2 – 0 – 5</td><td><b>Credits:</b>4</td></tr>
<tr><td colspan="2"><b>Contact Period - Lecture:</b> 52Hrs<b>.; Exam:</b> 3Hrs<b>.</b></td><td colspan="2"><b>Weightage: CIE:</b> 50%<b>; SEE:</b> 50%</td></tr>
</table>

Prerequisites: The student should have acquired the knowledge of Engineering Mathematics-I, II and III of I, II and III semester B.E.

Course Learning Objectives (CLOs):

This Course aims to;

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.
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.
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
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.
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

Relevance of the Course:

Engineering Mathematics-IV deals with solving algebraic, transcendental and ordinary differential equations arising in various engineering flow and design data problems.

In linear algebra deals with vector space, subspace, Rank and nullity, linear Transformation related to engineering problem

Complex Analysis. Here we understand the basics of complex variable, analyticity and potential fields through complex potential and conformal transformations interpret the solution in fluid flow and electromagnetic problems.

The process of complex integration and series representation of functions of complex variables in field theory and other Engineering applications.

In Statistics interpretation and analyzing the data, fitting of curves of best fit for experimental data arising in engineering calculations and analyze the same by expressing in the form of regression lines.

Probability distributions and use them in analyzing and solving engineering problems associated with probability models

The concept of joint probability of two random variables and apply the knowledge of joint probability distribution in interpreting data through statistical measure. And, analyze the notion of higher transition probabilities, the Markov chain and queuing models arising in engineering problems

Understand series solution of ODE’s and special functions in engineering fields.

Course Content

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 first degree – Introduction. Taylor's series method. Modified Euler's method, Runge - Kutta method of IV order, Milne's and Adams predictor & corrector methods (All formulae without proof).

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

Linear Algebra-II: Introduction to vector spaces – subspaces, Linear combination of vectors, linearly independent/dependent sets; Bases and dimension. Linear transformation - Rank nullity theorem (Statement only). Change of basis. Matrix of linear transformations. (No proof for theorems/properties) - Illustrative examples from engineering field. **Numerical methods for system of linear equations-** Gauss-Jacobi and Gauss- Seidel iterative methods. Relaxation methods**.** Determination of largest eigen value and corresponding eigen vector by power method.

Self-Study Component: Ramanujan's Method to find the smallest root of a polynomial**.** **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 problems on properties of analytic functions (No proof). Construction of analytic function: Milne-Thomson method. Conformal transformation–Definitions. Discussion of transformations: $w=z^2$, $w=e^z$, w = $z+{}^1\!/_z\,(z \neq 0)$. Bilinear transformations. **Complex integration:** complex line integrals. Cauchy theorem, Cauchy integral formula. Taylor's and Laurent's series (Statements only). Singularities, poles and residues. Cauchy residue theorem (statement only). Simple illustrative examples.

Self-Study Component: Derivation of Cauchy- Riemann equation in Cartesian and polar form. Derivation of Cauchy theorem, Cauchy integral formula and Cauchy's residue theorem. **11 Hrs**

UNIT-IV

Statistics: Brief review of measures of central tendency and dispersion. Moments, skewness and kurtosis. Curve fitting – least square method $: y = a + bx;\ y = ax^b, y = ab^x \text{ and } y = ax^2 + bx + c.$ Prof. Karl Pearson's coefficient of correlation and lines of regression.

Probability Theory: 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.

Self-Study Component: Basic definitions of probability and problems up to Bayes' theorem. To fit curves of the type : $y = ae^{bx}$, Derivation of Mean and SD of Binomial &Poisson distribution**.** **11 Hrs**

UNIT – V

Joint probability distributions and Markov chains:
Concept of joint probability. Joint probability distributions of discrete random variables. Expectation, covariance, correlation coefficient – simple examples. Probability vectors, stochastic matrices. Fixed point and regular stochastic matrices. **Series solutions of ODE's and special functions:** Power series solution of a second order ODE, Series solution-Frobenius method. Series solution leading to $J_n(x)$- Bessel's function of first kind. Expansions for $J_{1/2}(x) and\ J_{-1/2}(x)$. -simple related examples. Series solutions of Legendre's differential equation leading to $P_n(x)$-Legendre's polynomials. Rodrigues'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**

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 , 2007

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

Course Outcomes

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

1. Apply the familiarity of numerical methods for solving algebraic and transcendental equations and demonstrate single-step and multi-step numerical methods for solving ordinary differential equations and interpret the solution in engineering applications.
2. Describe the concept of vector space, subspace, basis, dimension and their practical utility in matrix of linear transformations required in the area of graphics, analysis of graphs, internet search, machine learning and scientific computing etc. And, understand the procedure of numerically solving large systems of linear algebraic equations and obtaining eigen value and eigen vector corresponding to a large eigen vector, with the aid of standard methods of numerical linear algebra.
3. Explain the concept of analyticity and potential fields through complex functional /potential, conformal transformations and interpret the solution in fluid flow and electromagnetic problems and describe the process of complex integration and learn series representation of a function of complex variables, residues and poles.

4. Apply the knowledge of statistics in interpretation the data, fitting of a linear and non-linear curves of best fit for experimental data arising in engineering calculations and analyze the same by expressing in the form of regression lines. And, Illustrate the concept of random variables (discrete/continuous) and related probability distributions and use them in analyzing and solving engineering problems associated with probability models
5. Define the concept of joint probability of two random variables and apply the knowledge of joint probability distribution in interpreting data through statistical measure. and, analyze the notion of higher transition probabilities, the Markov chain and queuing models arising in engineering problems for feasible random events.
   Obtain series solution of essential ODE's such as Bessel's and Legendre's differential equations and understand their scientific/engineering utility

CO-PO Mapping

Semester: 4 Course code : P17MAES41		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**	-	-	-	-	-	-	-	-	-	-

Course title: Operating System			
Course Code: P17IS42	**Semester:** IV	**L-T-P-H :** 3-0-2-5	**Credit:**4
Contact Period: Lecture: 52 Hrs, Exam: 3 Hrs		**Weightage:** CIE:50%, SEE: 50%	

Prerequisites: Operating system, computer networks

Course Learning Objectives (CLOs)

This course aims to:

1. Understand the basic principles of OS and its functionalities.
2. Describe CPU scheduling algorithms and process synchronization in OS.
3. Discuss deadlocks and memory management concepts in OS.
4. Summarize virtual memory management techniques and basic file system.
5. Understand file system structure and disk management strategies.

Relevance of the course:

An operating system is an essential part of any computer system. The purpose of this course is providing a clear understanding of the concepts that underlie operating systems. This course covers the classical internal algorithms and structures of operating systems, including CPU Scheduling, memory management, and device management, file systems, virtual memory, disk request scheduling, concurrent processes, deadlocks, security, and integrity. This course may be taken by other students with sufficient computer science background who have an interest in learning how an operating system works.

Course Content

UNIT – I

INTRODUCTION AND PROCESS CONCEPT: Operating System Structure – Operating System Operations – Process Management – Memory Management – Storage Management – System Structures: Operating System Services – User Operating System Interface – System Calls – Types of System Calls – System Programs. Process Scheduling – Operations on Processes – Inter-process Communication.

Self –Study Component: Protection and Security – Distributed Systems – Computing Environments

11 Hours

UNIT-II

PROCESS SCHEDULING AND SYNCHRONIZATION: Overview – Multithreading Models – Threading Issues Basic Concepts of process scheduling – Scheduling Criteria – Scheduling Algorithms – – Synchronization – The Critical-Section Problem – Peterson's Solution – Synchronization Hardware – Semaphores – Classic problems of Synchronization – Monitors.

Self-study Component: Multiple-Processor Scheduling **11 Hours**

UNIT-III

DEADLOCK AND MEMORY MANAGEMENT STRATEGIES:System Model – Deadlock Characterization – Methods for Handling Deadlock – Deadlock Prevention – Deadlock Avoidance – Deadlock Detection – Recovery from Deadlock. Swapping – Contiguous Memory Allocation – Paging –– Segmentation.

Self –Study - Component: Structure of the Page Table **10 Hours**

UNIT-IV

VIRTUAL MEMORY MANAGEMENT AND FILE SYSTEM: Demand Paging – Copy on Write – Page Replacement – Allocation of Frames – Thrashing File Concept – Access Methods – Directory Structure – File sharing.

Self- study-Component: Protection **10Hours**

UNIT-V

IMPLEMENTING FILE SYSTEMS AND SECONDARY STORAGE STRUCTURE: File System Structure – File System Implementation – Directory Implementation – Allocation Methods – Free-space Management. Disk Structure – Disk Scheduling – Disk Management – Swap-Space Management.

Self Study Component: Case Study: Linux OS, Android OS. **-10 Hours**

Text Books:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, “Operating System Principles”, John Wiley & Sons (Asia) Pvt. Ltd, Ninth Edition, 2013.
2. Andrew S. Tanenbaum, “Modern Operating Systems”, Third Edition Prentice Hall of India Pvt. Ltd, 2010 (Case Study Topic).

Reference Books:

1. Harvey M. Deitel, “Operating Systems”, Pearson Education Pvt. Ltd, Second Edition.
2. William Stallings, “Operating System”, Pearson Education, Sixth Edition, 2012.
3. Christopher Negus, Christine Bresnahan, “Linux Bible”, Wiley Pvt Ltd, Eighth Edition, 2013.
4. http://nptel.ac.in/courses/Webcoursecontents/IIScBANG/Operating%20Systems/New_index1.html.
5. www.thenewboston.com (for Android Application development)

Course Outcomes:

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

1. Outline the operating system concepts and its functionalities. (L2)
2. Implementation of various CPU scheduling algorithms and process synchronization using programming languages.(L3)
3. Identify deadlock Occurrence, deadlock recovery in various OS and outline memory management concepts.(L2)
4. Compare page replacement algorithms in OS and understand fundamental file concepts. (L4)
5. Discuss file system structure and implement disk scheduling algorithms.(L2)

Sem: IV Sem		**Course Code:** P17IS42					**Course Title**: Operating systems									
CO	**Statement**	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO1	Outline the operating system concepts and its functionalities.	2	1	1					1					2	1	1
CO2	Implementation of various CPU scheduling algorithms and process synchronization using programming languages	3	3	3					2	2				2	1	2
CO3	Identify deadlock Occurrence, deadlock recovery in various OS and outline memory management concepts	3	2	1					2	2				2	1	1
CO4	Compare page replacement algorithms in OS and understand fundamental file concepts.	2	2	1					2	2				2	1	1
CO5	Discuss file system structure and implement disk scheduling algorithms.	2	2	3					2	2				2	2	1

Course title: Finite Automata and Formal language			
Course Code: P17IS43	**Semester:** IV	**L-T-P-H :** 4-0-0-4	**Credit:**4
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 finite automata and Explain equivalence and minimization of finite automata.
2. Design regular expression for regular languages, 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 grammars from push-down automata and Design push-down automata from grammars.
5. Design Turing machines for simple languages and functions and Design problem reductions to determine the un-decidability of languages.

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

Introduction to Finite Automata, Regular Expression

Introduction to Finite Automata: The central concepts of Automata theory; Deterministic finite automata; Nondeterministic finite automata. Application of finite automata; Finite automata with Epsilon transitions;

Self-Study Component: Equivalence and minimization of automata. **10 Hrs**

UNIT-II

Regular Expression, Regular Languages, Properties of Regular Languages: Regular expressions; Finite Automata and Regular Expressions; Applications of Regular Expressions. Regular languages; Proving languages not to be regular languages; Closure properties of regular languages;

Self-Study Component: Decision properties of regular languages. **10 Hrs**

UNIT-III

Context-Free Grammars And properties of Context-Free Languages: Context –free grammars; Parse trees; Applications; Ambiguity in grammars and Language**s,** Definitions of Normal forms for CFGs;

Self-Study Component: The pumping lemma for CFGs; Closure properties of CFLs. **12 Hrs**

UNIT-IV

Pushdown Automata Definition of the Pushdown automata; The languages of a PDA; Deterministic Pushdown Automata.

Self-Study Component: Equivalence of PDA’s and CFG’s; **10 Hrs**

UNIT-V

Introduction to Turing Machine, Un-decidability Problems that Computers cannot solve; The turning machine; Programming techniques for Turning Machines; Extensions to the basic Turning Machines; Turing Machine and Computers.

Self-Study Component: Un-decidable problem that is RE; Post's Correspondence problem. **10 Hrs**

Text books:

1. John E.. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman: Introduction to Automata Theory, Languages and Computation,3rd Edition, Pearson education.

Referenced books:

1. Raymond Greenlaw, H.James Hoover: Fundamentals of the Theory of Computation, Principles and Practice, Morgan Kaufmann.
2. John C Martin: Introduction to Languages and Automata Theory, 3rd Edition, Tata McGraw-Hill.
3. Daniel I.A. Cohen: Introduction to Computer Theory, 2nd Edition, John Wiley & Sons.
4. Thomas A. Sudkamp: An Introduction to the Theory of Computer Science, Languages and Machines, 3rd Edition, Pearson Education.

Course Outcomes

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

1. Construct finite automata
2. Analyze regular expression
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		

<table>
<tr><td colspan="4"><b>Course title:</b> Analysis and Design of Algorithms</td></tr>
<tr><td><b>Course Code:</b> P17IS44</td><td><b>Semester:</b> IV</td><td><b>L-T-P-H :</b> 4-0-0-4</td><td><b>Credit:</b>3</td></tr>
<tr><td colspan="2"><b>Contact Period:</b> Lecture: 52 Hrs, Exam: 3 Hrs</td><td colspan="2"><b>Weightage:</b> CIE:50%, SEE: 50%</td></tr>
</table>

Prerequisites: The Students are required to have familiarity with programming language and the following data structures such as Arrays, Linked List, Stack, Queue, Graphs, Trees, and Binary search trees, Heap and Priority Queues.

Course Learning Objectives:

- Ability to analyze time complexity of the algorithms.
- To find an algorithm to solve the problem (create) and prove that the algorithm solves the problem correctly (validate).
- To synthesize efficient algorithms in common engineering design situations.
- Implementation of various searching, sorting algorithmic techniques.
- Implementation of various algorithmic techniques like Greedy strategy, Divide and Conquer approach, Dynamic Programming and Backtracking.

Relevance of the course:

This course, part of the Information Science and Computer Science Essentials for Software Development Professional Certificate program, is an introduction to design and analysis of algorithms, and answers along the way these and many other interesting computational questions. You will learn about algorithms that operate on common data structures, for instance sorting and searching; advanced design and analysis techniques such as dynamic programming and greedy algorithms; advanced graph algorithms such as minimum spanning trees and shortest paths; NP-completeness theory; and approximation algorithms. After completing this course you will be able to design efficient and correct algorithms using sophisticated data structures for complex computational tasks.

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

Self Study Component: Closest pair and convex-hull problem using Brute force and Divide and Conquer Method. **10 Hrs**

UNIT - III

Decrease and Conquer: Insertion Sort, Depth First Search, Breadth First Search, Topological Sorting. **Transform and Conquer:** Gaussian Elimination, Balanced Search Trees, Heaps and Heap sort.

Self Study Component: Fake-coin problem, 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, 2nd Edition, Pearson Education, 2007

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

CO1: **Define** the various algorithm analysis methods.

CO2: **Estimate** the asymptotic time complexities of various recurrence relations**.**

CO3: **Choose** the appropriate method to solve the recurrence relations

CO4: **Distinguish** and **give examples** for the different types of algorithm development strategies.

CO5: **Describe** the use of Lower Bound Theory to improve the complexity of an algorithm

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													
CO 2		3	3										1		
CO 3	3	3	3	3									2		
CO 4	3	3	3	3	2						2		2		
CO 5	3		3	1							2				

Course Title: Software Engineering			
Course Code: P17IS45	**Semester :** IV	**L- T – P - H :** 4 – 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. 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

CO 1. Demonstrate an understanding of the principles and techniques of Software Engineering

CO2. Analyze the various steps involved in the design process and the different design approaches which include function-oriented design and object-oriented design

CO3. Understand the activities in project management, requirement engineering process and to identify the different types of system models

CO4. Apply the knowledge of design engineering in software development

CO5. 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: Microprocessors			
Course Code: P17IS46	**Semester:** IV	**L-T-P-H :** 4-0-0-4	**Credit:** 4
Contact Period: Lecture: 52 Hrs, Exam: 3 Hrs		**Weightage:** CIE:50%, SEE: 50%	

Prerequisites: Computer Organization.

Course Learning Objectives (CLOs)

This course aims to

1. Identify the various elements of 8086 microprocessor architecture, its bus organization including control signals.
2. Understand in-depth the hardware and software included in micro computer systems.
3. Analyze the concept considered are general in nature, the discussion is based on the particular microprocessor, the Intel 8086/8088 and its associated supporting devices and software.
4. Learn the programming concept of 8086 programming using (Microsoft assembler) MSAM Assembler.
5. Understand 32/64 bit architectures supporting pipelined and superscalar architectures.

Relevance of the course

This course deals with the systematic study of the Architecture and programming issues of 8086-microprocessor family and interfacing with other peripheral ICs and co-processor. This course introduces the assembly language programming of 8086 and basic concepts of microprocessor and helps students to develop the assembly language programming and real time applications of Microprocessor. The main aim of this course is to give the students basic knowledge of the microprocessors needed to develop the systems using it.

Course contents

UNIT- I

8086 Architecture and Instructions: CPU architecture, Internal Operation, Machine Language Instructions, addressing modes, Instruction formats, Assembler instruction format, Data transfer Instructions, Arithmetic instructions.

Self study component: Instruction Execution Timing **10Hrs**

UNIT- II

Assembly language programming Instructions continued: Branch instructions- conditional branch instruction, unconditional branch instructions, Loop instructions, NOP and HLT instructions, Logical Instructions, Shift and Rotate Instructions, Directives and Operators- data definition and storage allocation, Structure, Records, Assigning name of expression, Segment definition, Program termination,

Self study component: Alignment directives, Value returning Attribute Operators **11Hrs**

UNIT- III

Modular Programming: Linking and Relocation – Segment Combination, Access to External Identifiers, Stacks, Procedures – Calls, Returns and Procedure Definitions, Saving and Restoring Register, Interrupts and Interrupt Routines, MSAM Macros.

Self study component: Parameter passing in Procedures and Macros, Difference between procedures and Macros **11Hrs**

UNIT- IV

Byte and String Manipulation: String Instructions, REP prefix, table translation. **I/O programming:** Fundamental I/O considerations, Programmed I/O, Block transfers and DMA**.**

Self study component: Interrupt I/O **10Hrs**

UNIT- V

System Bus Structure: Basic 8086/8088 configurations – Minimum mode, Maximum mode, System Bus Timing, Interrupt Priority Management – Interrupt System based on Single 8259A,

Self study component: Interrupt System Based on Multiple 8259As, Bus Standards**.** **10 Hrs**

Text Book:

1. **Microprocessor Systems: the 8086/8088 Family,** Glenn A.Gibson, Prentice-Hall of India, 2nd edition.

Reference Books:

1. **The Intel Microprocessors**, Barry.B.Brey, PHI Publication, 8th edition, 2009
2. **Microprocessor and Interfacing,** Douglas V.Hall, TMH, 2nd edition.
3. **The Intel Microprocessor Family: Hardware and Software Principles and Applications,** James L. Antonakos, Thomson.

Course Outcomes

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

1. Analyze the architecture of 8086 microprocessor
2. Apply 8086 instruction set for the given problems
3. Develop different modules & link them
4. Apply string instruction set and I/0 Interrupt in 8086 programming
5. Analyze min & max mode of 8086

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			3		2		2		1		1		1
CO 2	3	2			3		2		2		2				1
CO 3	3	2					2		2				1	1	2
CO 4	3	2					3		2		2		1	1	3
CO 5	3	3			1		2		2		1		1		3

Course title: Analysis and Design of Algorithms Lab			
Course Code: P17ISL47	**Semester:** IV	**L-T-P-H** : 0-1-2-3	**Credit:** 1.5
Contact Period: Lecture: 36 Hrs, Exam: 3 Hrs		**Weightage:** CIE:50%, SEE: 50%	

Prerequisites: Programming Knowledge of C/C++.

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 C++.

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.

Couse 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: Microprocessor Laboratory			
Course Code: P17ISL48	**Semester:** IV	**L-T-P-H :** 0-1-2-3	**Credit:** 1.5
Contact Period: Lecture: 36 Hrs, Exam: 3 Hrs		**Weightage:** CIE:50%, SEE: 50%	

Course Learning Objectives (CLOs)

This course aims to

1. Implement programs related to Searching, sorting and strings.
2. Implement programs using Macro, procedure and files.
3. Implement Data conversion, number generation.
4. Interface hardware device to 8086 processor

Course Content

Part - A

1. Program on searching and sorting,
2. Macros and Procedures (subroutines)
3. Data conversions
4. Strings
5. Recursion
6. File (Read and write)
7. Number Generation

Part - B
(Using Interfacing concepts)

1. Programs on logic controller,
2. Programs on7-segment display,
3. Programs on stepper motor interface,
4. Programs on 8X3 Keypad,
5. Programs on elevator interface.

Note: Out of the exercises executed in the regular laboratory classes, each students will be allotted one question from Part A and one question from Part B, by taking lots in the SEE which he has to execute individually.

Couse Outcomes

1. Implement programs related to searching, sorting and strings
2. Implement programs using macro, procedures and files.
3. Implement data conversion, number generation
4. Interface hardware device to 8086 processor

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	1										2		
CO 2	2	3	1										2		
CO 3	2	3	1										2		
CO 4	2	2	3		1								2		2

Course Title : Aptitude and Reasoning Development - INTERMEDIATE (ARDI)			
Course Code : P17HU49	**Semester :** IV	**L - T - P :** 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.

Self-study Component- 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.
Self-study Component- 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. L5
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. L5
3. Apply the concept of L.C.M in the module time and work to solve the problems with comprehension. L2
4. Analyze the concepts in Co-ordinate geometry by spatial visualization. L4
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. L4
6. Determine the solutions for complicated problems of set theory using the concept of venn diagram. L4

Course Articulation Matrix

Course Outcomes		**Program Outcomes**												**PSO**	
		1	2	3	4	5	6	7	8	9	10	11	12	01	02
CO1	Solve problems of higher difficulty level with ease in the following topics– Time, speed and distance and Geometry.	**3**	**2**	**2**	**-**	**1**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	-	-
CO2	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.	**-**	**2**	**2**	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	-	-
CO3	Apply the concept of L.C.M in the module time and work to solve the problems with comprehension.	**-**	**2**	**2**	**-**	**1**	**-**	**-**	**-**	**1**	**-**	**1**	**-**	-	-
CO4	Analyze the concepts in Co-ordinate geometry by spatial visualization.	**3**	**2**	**2**	**2**	**1**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	-	-
CO5	Interpret the logic in the statements of syllogism by critical thinking and apply venn diagram for the effective's ways of deriving at the conclusion.	**-**	**2**	**2**	**2**	**-**	**-**	**-**	**2**	**-**	**2**	**-**	**-**	-	-
CO6	Determine the solutions for complicated problems of set theory using the concept of venn diagram.	**-**	**2**	**2**	**2**	**-**	**-**	**-**	**-**	**2**	**-**	**1**	**-**	-	-

Course Title : Additional Mathematics-II			
Course Code : P17MADIP41	**Semester :** IV	**L - T - P : 4** - 0 - 4	**Credits:** NA
Contact Period: Lecture: 52 Hr**, Exam: --**	**Weightage: CIE:**50%; **SEE:**50%		

((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. 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. Solutions of initial value problems. Method of undetermined coefficients and variation of parameters. Solution of Cauchy's homogeneous linear equation and Legendre's linear differential equation. **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. 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. 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. Bayes's theorem-examples. **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