Syllabus

(With effect from 2015-2016 Academic year)

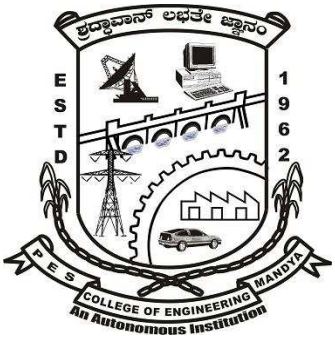
ಪಠ್ಯಕ್ರಮ

(ಶೈಕ್ಷಣಿಕವರ್ಷ 2015–16)

III & IV Semester

Bachelor Degree
in
Computer 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.

Sri. B.Dinesh Prabhu
Deputy Dean (Academic)
Associate Professor,
Dept. of Automobile Engg

Dr.P S Puttaswamy
Dean (Academic)
Professor,
Dept. of Electrical & Electronics Engg.

P.E.S.COLLEGE OF ENGINEERING, MANDYA-571401
(KARNATAKA)
(An Autonomous Institution under VTU, Belagavi)

Vision

"An institution of high repute, imparting quality education to develop innovative and Humane engineers"

Mission

"Committed to develop students potential through high quality teaching- learning processes and state of the art infrastructure"

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

About the Department

The Department of Computer Science and Engineering was established in 1983. The department offers B.E. program with an intake of 120 students, M.Tech. in Computer Science and Engineering with 18 students, M.Tech. in Computer Engineering with 24 students and also Ph.D. programme. Currently the strength of teaching faculty is 32 and that of non teaching staff is 14. The teacher - student ratio is 1:16. The department has a research centre under VTU and University of Mysore, with 2 research guides and 8 research students. During the last five years, the department has published 85 technical papers in international/national journals/conferences. So far, the department has organized four international and 8 national conferences. The department is equipped with all the required infrastructure, laboratories, class rooms, departmental library. The departments wish to achieve the mission of developing and nourishing computer science engineers through well-trained, committed and experienced faculty members. Faculty members of the departments are involved in research activities in different fields such as Image Processing, Pattern Recognition, Data Mining, Wireless Networks, Big Data Analytics and Computer Vision.

Vision

"*To develop globally competent computer professionals by exploring latest technologies through continuous learning, research and innovation."*

Mission

"To impart quality technical education in modern practices of Computer Science and Engineering through competent faculties, state of the art teaching–learning infrastructure and methodologies".

Department of Computer Science and Engineering

Programme Educational Objectives (PEOS) are

I. To provide students with a strong basis in the mathematical, scientific and engineering fundamentals to solve computer science engineering problems and to prepare them for employment, higher learning, R&D and consultancy.

II. To provide technical knowledge, skills and awareness of current technologies of computer science engineering and to develop an ability to design and provide novel engineering solutions for software/hardware problems through entrepreneurial skills.

III. To provide an exposure to emerging cutting edge technologies to work as teams on multidisciplinary projects *with* effective communication skills and leadership qualities.

IV. To provide an ability to function ethically and responsibly in a rapidly changing environment by applying innovative technologies that allows them to become effective professionals in Computer *Science to sustain a life-long career in related areas.*

Program outcomes (POs)

A graduate of the Computer Science and Engineering Program will demonstrate

a) An ability to apply knowledge of computing, mathematical foundations, computer science and engineering fundamentals. (Fundamental engineering analysis skills).
b) An ability to analyze a problem, and identify and formulate the computing requirements appropriate to its solution (Information retrieval skills).
c) An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs. (creative skills).
d) An ability to identify, formulate and solve computer science engineering problems for obtaining its solutions using the appropriate computing and engineering requirements (engineering problem solving skills).
e) An ability to use current techniques, skills, and modern engineering tools necessary for computing and engineering practice (practical engineering analysis skills).
f) The broad education necessary to understand the impact of engineering solutions in a global and societal context (engineering impact assessment skills).
g) A knowledge of contemporary issues (social awareness).
h) An understanding of professional, ethical, legal, security and social issues and responsibilities (professional integrity).
i) An ability to function effectively individually and on teams to accomplish a common goal (teamwork).
j) An ability to communicate effectively, both in writing and orally with a range of audiences (speaking / writing skills).
k) Recognition of the need for, and an ability to engage in continuing professional development and life-long learning (continuing education awareness).
l) An ability to acquire concepts to integrate computer science and engineering principles in the construction of software and hardware systems of varying complexity (software hardware interface).
m) An ability to recognize the importance of professional development by pursuing postgraduate studies or face competitive examinations that offer challenging and rewarding careers in computing (successful career and immediate employment).

These programme outcomes (POs) are achieved through an array of courses. To ensure the achievement of POs, the course learning outcomes (CLOs) are so formulated that they address these POs.

PES COLLEGE OF ENGINEERING, MANDYA

(An Autonomous Institution, Under VTU)

Scheme of Teaching and Examination for B.E. (3rd & 4th) Computer Science & Engineering

III Semester

Sl. No	Course code	Course Title	Teaching Dept.	Hrs/Week	Total Credit	Examination Marks		
				L :T: P:H		CIE	SEE	Total
1	P15MA31	Engineering Mathematics-III	Maths	3:2:0:5	4	50	50	100
2	P15CS32	Digital Logic Design	CS	3:2:0:5	4	50	50	100
3	P15CS33	Data Structures	CS	4:0:0:4	4	50	50	100
4	P15CS34	Discrete Mathematical Structures	CS	2:2:0:4	3	50	50	100
5	P15CS35	Object Oriented Programming with C++	CS	3:2:0:5	4	50	50	100
6	P15CS36	Computer Organization	CS	4:0:0:4	4	50	50	100
7	P15CSL37	Data Structures Lab	CS	0:1:2:3	1.5	50	50	100
8	P15CSL38	Digital Logic Design Lab	CS	0:1:2:3	1.5	50	50	100
9	P15HUDIP39	Comprehensive Communication Development (CCD)	HS&M	0:0:2:2	[2]	[50]	--	50
10	P15HU39	** Aptitude and Reasoning Development – BEGINNER (ARDB)	HS&M	2:0:0:2	0	[50]	--	100
11	P15MADIP31	*Additional Maths-I	Maths	2:0:0:2	0	--	--	--
12	P15HMDIP310	* Constitution of India & Professional Ethics	Human & Science	2:0:0:2	0	--	--	--
				Total	26[28]	400[450]	400[450]	800[900]

* Additional Mathematics-I & Constitution of India and Professional Ethics : Lateral entry students shall have to pass these mandatory learning courses before completion of VI semester

** ARDB : All students shall have to pass this mandatory learning courses before completion of VI semester

IV Semester

Sl. No	Course code	Course Title	Teaching Dept.	Hrs/Week	Total Credit	Examination Marks		
				L :T: P:H		CIE	SEE	Total
1	P15MA41	Engineering Mathematics-IV	Maths	3:2:0:5	4	50	50	100
2	P15CS42	Graph Theory & Combinatorics	CS	3:2:0:5	4	50	50	100
3	P15CS43	Theory of Computation	CS	4:0:0:4	4	50	50	100
4	P15CS44	Analysis and Design of Algorithms	CS	4:0:0:4	4	50	50	100
5	P15CS45	Unix System Programming	CS	3:2:0:5	4	50	50	100
6	P15CS46	Microprocessor	CS	4:0:0:4	3	50	50	100
7	P15CSL47	Analysis and Design of Algorithms Lab	CS	0:1:2:3	1.5	50	50	100
8	P15CSL48	Object oriented programming with C++ Lab	CS	0:1:2:3	1.5	50	50	100
9	P15HU49	** Aptitude and Reasoning Development – INTERMEDIATE (ARDI)	HS&M	2:0:0:2	1	50	50	100
10	P15MADIP41	*Additional Maths-II	Maths	2:0:0:2	0	--	--	--
11	P15EVDIP410	*Environmental Studies	ENV	2:0:0:2	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

* Common to B.E. (AU,CV, ME and I&PE) | ** Common to B.E. (CS, EC, E&E and IS&E)

Evaluation Scheme							
Scheme	**Weightage**	**Marks**	**Event Break Up**				
CIE	50%	50	**Test I**	**Test II**	**Quiz I**	**Quiz II**	**Assignment**
			35	35	05	05	10
SEE	50%	50	**Questions to Set: 10**		**Questions to Answer: 5**		

Course Title: Engineering Mathematics-III			
Course Code: P15MA31	**Semester:** III	**L – T – P – H :** 3– 2 – 0 – 5	**Credits:** 04
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 P15MA31 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.

Central differences: Gauss Forward and Backward difference formulae, Stirling's, and Bessel's formulae (All formulae without proof) – Illustrative problems. **10 Hrs**

UNIT-II

Numerical differentiation using Newton's forward and backward interpolation formulae, Newton's divided difference formula and Stirling'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 $(\frac{1}{3})^{rd}$ rule, Simpson's $(\frac{3}{8})^{th}$ rule, Boole's rule and Weddle's rule (All rules without proof)-Illustrative problems. **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. **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 – problems. Convolution theorem, Parseval's identities for Fourier transforms (statements only).

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

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. **11 Hrs**

Text Books:

1. B.S. Grewal: Higher Engineering Mathematics, 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. Peter V O' Neil – Advanced Engineering Mathematics, Thomson Brooks/Cole ,5th edition,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 forward, backward difference formulae and central differences formulae in solving interpolation- extrapolation problems in engineering field.
2. Apply Numerical differentiation and integration rules in solving engineering where the handling of numerical methods is inevitable.
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.
4. 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.
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.

Course Title : Digital Logic Design			
Course Code : P15CS32	**Semester :** III	**L :T:P:H** : 3:2:0:5	**Credits: 4**
Contact Period: Lecture: 52 Hr, Exam: 3 Hr		**Weightage**: CIE:50; SEE:50	

Prerequisites:

1. Basic Knowledge of Number System
2. Electronic Circuits
3. Programming Knowledge

Course Learning Objectives (CLO's)

The student should be able to,

1. **Describe** the Boolean Laws and Theorems and minimize Boolean expressions using different techniques.
2. **Design** data processing circuit and arithmetic circuit.
3. **Analyze** the working of flip-flops and **Use** the flip-flops to design registers and counters.
4. **Design** the asynchronous and synchronous counter for any modulus and
5. **Convert** digital data to analog data and vice-versa and **write** VHDL code for logic circuits.

Relevance of the course:

1. This course presents the design skills and theoretical knowledge needed to design, simulate and build combinational logic circuits and basic sequential circuits.
2. The student will also understand the basic internal working of CPU and ins interface with memory and I/O systems.
3. This course gives students sufficient preparation for the microprocessor system course.

Course Content

Unit-1

Digital Logic and Combinational Logic Circuits:

Overview of Basic Gates and Universal Logic Gates, AND-OR –Invert Gates, Positive and Negative Logic, Boolean Laws and Theorems, Sum-of-products Method, Truth table to Karnaugh Map, Pairs, Quads, and Octets, Karnaugh Simplification, Don't Care Conditions, Product-of-Sum Method, Product-of-sum Simplification, Simplification by Quine-McClusky Method, Simplification by VEM Technique. 11 Hours

Unit-2

Data Processing Circuits and Arithmetic Circuits:

Multiplexers, Demultiplexers, Decoders, BCD-to-Decimal Decoders, Seven-segment Decoders, Encoders, Ex-OR gates, Parity Generators and Checkers, Magnitude Comparators, Design of code converters, Half Adder, Full Adder, Half Subtractor, Full Subtractor, Fast Adder, Adder- Subtractor 10 Hours

Unit-3

Memory Devices: Read-only memory(ROM), PROM, EPROM, EEPROM , Programmable Array Logic (PAL), Programmable Logic Array (PLA).

Flip-Flops and Simple Flip-Flop Applications: The Basic Bistable Elements, Latches, Timing Considerations, Master-Slave Flip-Flops Pulse-triggered Flip-Flops, Edge –Triggered Flip-Flops, Characteristics Equations, conversion of flip-flops. 10 Hours

Unit-4

Registers: types of registers, serial in serial out , serial in parallel out, parallel in serial out, parallel in parallel out, Application of shift registers: Ring counter, Johnson counter, sequence detector and sequence generator.

Asynchronous and synchronous counter: Asynchronous counter- up, down, up and down counter, design of synchronous up counter and down counter, decade counter, counter design as a synthesis problem. 10 Hours

Unit-5

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

VHDL Programming: Introduction to VHDL, Describing data flow, Behavioral, Structural and Mixed design style, Simulating design for Arithmetic and Combinational circuits.

11 Hours

Text Books:

1. Digital Principles and Applications: Donald P Leach, Albert Paul Malvino & Goutham Saha, TMH, 8th Edition, 2014.
2. A Verilog HDL Primer, 2nd Edition, J . Bhaskar, BS Publications

Reference Books:

1. Digital Principles & Design by Donald D Givone, 4th Reprint, Tata McGraw Hill 2009.
2. Fundamentals of Digital Logic with Verilog Design, Stephen Brown, ZVonkoVranesic, TMH, 2006

Course Outcomes :

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

1. **Understand** Basic gates and Universal gates.
2. **Design** the combinational logic circuits with minimum gates.
3. **Apply** the logic to design memory circuits.
4. **Design** shift registers and counters using flip-flops.
5. **Understand** and **Write** the VHDL code for all logic circuits

Course Title : Data Structures			
Course Code : P15CS33	**Semester : III**	**L :T:P:H : 3:2:0:5**	**Credits: 4**
Contact Period: Lecture: 52 Hr, Exam: 3 Hr		**Weightage: CIE:50; SEE:50**	

Prerequisites: Subject requires student to know about

1. Basic C programming skills
2. Basics of computers

Course Learning Objectives (CLOs)

-

This course aims to

1**. Understand** Abstract data types, Stacks and recursion
2. **Learn** the different types of linked list
3. **Learn** the Basic operations on - Linear queue, Circular queue, Priority Queue and Double ended Queue .
4. **Apply** the différent tree traversal techniques
5. **Learn** the different sorting and searching techniques.

Relevance of Course:

1. Data structures are used in almost every program or software system.
2. Specific data structures are essential ingredients of many efficient algorithms, and make possible the management of huge amounts of data, such as large integrated collection of databases.
3. Some programming languages emphasize data structures, rather than algorithms, as the key organizing factor in software design.
4. They help you write efficient code and solve problems in optimal or near-optimal ways. Without them, you will be reinventing the wheel - not always successfully.
5. Also, they help you structure your code, so that it can be maintained more easily by encouraging a better design / implementation.

Course Content

Unit-1

Introduction to data structures-Definition, Abstract Data Types-ADT for rational numbers, 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. 10 Hours

Unit-2

Linked Lists

Static Memory Allocation and Dynamic Memory Allocation, Basic operations on SLL, DLL, Circular SLL and Circular DLL: insertion, deletion and display. Implementation of SLL with

Header nodes. 10 Hours

Unit-3

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. 10 Hours

Unit-4

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, Inserting a Node into a Threaded Binary Tree. 12 Hours

Unit-5

Sorting Techniques

Insertion sort, Quick sort, Binary tree sort, Heap sort, Merge sort.

Searching Techniques

sentinel search, probability search, ordered list search (Text Book-2) 10 Hours

Text Book :

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

Reference Book :

1. Fundamentals of Data Structures in C - Horowitz, Sahani, Anderson-Freed, Second Edition, University Press, 2nd Edition. Understand primitive and derived data structure.

Course Outcomes

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

1. **Design** and Implement standard data structures like stack using recursion
2. **Design** and implement operations on linked list.
3. **Develop** programs to implement different queues.
4. **Design and** implement different tree traversal techniques using iteration and recursion..
5. **Implement** sorting and searching techniques

Course Title : Discrete Mathematical Structures			
Course Code : P15CS34	**Semester : III**	**L :T:P:H : 2:2:0:4**	**Credit: 3**
Contact Period: Lecture: 52 Hr, Exam: 3 Hr		**Weightage: CIE:50; SEE:50**	

Prerequisites: Logic Design

Course Learning Objectives :

1. Analyze and Solve problems using simple techniques of counting theory and set theory.-L1, L2
2. Learn and identify the fundamentals of logic and use of Quantifiers. –L1, L2, L3
3. Understand the importance of induction principle and pigeonhole principle in proving statements-.L2, L3, L4
4. Learn and understand the basic concepts relations and functions, and their representations.-L1, L2, L3
5. Learn the concepts of group theory and learn and apply coding theory concepts to code and decode a message.-L1, L5

Relevance of the Course: This course is to develop the mathematical ability of the student in the application areas such as set theory, counting theory, logic theory, coding theory, data structures, theory of computer languages and the analysis of algorithms. The student gets introduced to applications in engineering, physical and life sciences, statistics and social sciences. 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 1

Principles of counting: The rules of sum and product, Permutations, Combinations: The Binomial theorem- combinations with repetition.
Set Theory: Sets and subsets, set operations and the Laws of set theory, Counting and Venn Diagrams, A First Word on Probability. 10 Hours

Unit 2

Fundamentals of Logic: Basic Connectives and Truth Tables, 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) 10 Hours

Unit 3

Properties of Integers: Mathematical Induction, The Well Ordering Principle- Mathematical Induction in the Alternative form, Recursive Definitions.
Relations and Functions: Cartesian Products and Relations, Functions .Plain and One-to-One, Onto Functions – Stirling's Numbers of theSecond Kind, The Pigeon-hole Principle, Function Composition and Inverse Functions. Special functions-characteristic function, Permutation function, Hashing function. 10 Hours

Unit 4

Relations Revisited: Properties of Relations Computer Recognition : Zero-One Matrices and Directed Graphs, Partial Orders - Hasse Diagrams.

Equivalence Relations and Partitions- Partitions induced by Equivalence relations. Topological sorting algorithm, totally ordered sets. Extremal elements, Lattices. 12 Hours

Unit 5

Groups: Definitions, Elementary Properties, Homomorphism's, Isomorphisms, and Cyclic Groups, Cosets, and Lagrange's Theorem.

Coding Theory: Elements of Coding Theory, The Hamming Metric, The Parity Check, and Generator Matrices. Group Codes: Decoding with Coset Leaders. 10 Hours

Text Books :

1. Discrete and Combinatorial Mathematics, RalphP. Grimaldi &B.V. Ramana, 5th Edition, PHI/Pearson education. Chapter 8, 9, 10, 11, 12.
2. "Discrete Mathematical structures", Dr D. S. Chandrashekariah, Prism 2005.

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, 2004.
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. Understand the principles of counting and set theory. –L2
2. Identify the quantifiers and their uses and learn the fundamentals of logic theory.-L3
3. Apply the Mathematical induction principle and pigeonhole principle to solve the real time problems.-L5
4. Solve the problems using the concepts of relations and functions and Identify the different ways of representing relations.-L3
5. Apply the concepts of group theory and coding theory to solve the given problem.-L5

	Model Question Paper	Marks	CO's	Levels
	Unit-I			
1.a	Find the number of license plates created which contains two English alphabets followed by four digits i) with repetition ii) without repetition.	6	Co1	L2
b.	How many arrangements are there of all the letters in SOCIOLOGICAL such that (i) letters A and G are adjacent? (ii) are all the Vowels adjacent?	7	CO1	L2
c.	i) Determine the coefficient of $x^2y\,z^2$ in the expansion $(2x+y+3z)^{5.}$ ii) Find the number of positive integer solutions of the equation $x_1+x_2+x_3+x_4+x_5 = 32$ each is greater than or equal to zero.	6	CO1	L2
2.a	Define power set, subset, super set of A. For any three sets A, B, C Verify (A - C) - (B - C) = A- (B U C) = (A - B) – C	6	CO1	L2
b.	In a class of 31 students, a test of three questions was given and every student answered atleast one question, 6 students did not answer the first question, 7 failed to answer the second question and 8 did not answer the third question and 8 students answered all questions answered . Find the number of students who answered (i) exactly one question? (ii) atleast one question?	7	CO1	L3
c.	(i)If two integers are selected at random and without replacement from {1,2,…,99,100} .what is the probability that their sum is even. (ii) If a fair coin is tossed tour times what is the probability that two heads and two tails occur	7	CO1	L2
	UNIT - II			
3.a	Define tautology Is $(p \vee q) \rightarrow (p \rightarrow (p \wedge q))$ a tautology?(Justify your answer) using truth table and without using truth table.	6	CO2	L1,L2
b.	Define logical equivalence and using laws verify $(\neg p \square q) \square (p \square (p \square q)) \equiv (p \square q)$	6	CO2	L2
c.	Express symbolically and check the validity. It is not sunny this afternoon and is colder than yesterday. We will go for swimming if and only if it is sunny. If we do not go for swimming then we will take a trip. If we take a trip then we will be home by sunset. Therefore we will be home by sunset.	8	CO2	L3
4.a	Write the statements in the symbolic form with a specific universe for each (i) All students have greater than 80% attendance. (ii) Some students have enrolled in sports (iii) Some integers are divisible by 5 and are even	6	CO2	L2
b.	Define Rule of universal specification and generalization.	4	CO2	L2
c.	Expressing symbolically check the validity " No junior or senior has enrolled in sports. Raju has enrolled in sports. Therefore , Raju is not a senior."	6	CO2	L3
d.	Prove or disprove directly "The sum of any five consecutive integers is always divisible by 5"	4	CO2	L2
	Unit III			
5.a.	State mathematical Induction principle and Prove that $1.3+2.4+\ldots\ldots+n(n+2)= n(n+1)(2n+7) / 6$ for all integers $n\geq 1$.	6	CO3	L2
b.	(i) Write the given sequence in explicit form $a_1=8$, and $a_n=a_{n-1} + n$ for $n \geq 2$ (ii)Express the sequence recursively $a_n=3n+2$ for all $n \geq 1$	7	CO3	L2
c.	Define one-one fuctions, onto functions and find the number of one-one and	7	CO3	L1,

	onto functions from a set of m elements to a set of n elements.			L2
6.a.	Prove that any subset of size 6 from the set S={1,2,3,…,9}must contain two elements whose sum is 10.	6	CO3	L2
b.	Let f and g be two functions from R to R defined by f(x)=2x+1 and g(x) = x/3 , Find i) fog and gof (ii) $(gof)^{-1}$ and $f^{-1}o\ g^{-1}$	7	CO3	L2
c.	Define permutation function, Hashing function, characteristic function.	7	CO3	L1
	UNIT- IV			
7.a	Define an equivalence relation i)P.T. if R is a relation defined as "x+y =even" on Z ii) find the partition induced by R on A={1,3,5,6,8}.	8	CO4	L2
b	Define partially ordered set and draw the Hasse diagram of all positive divisors of 36.	8	CO4	L2
c	Write the relation matrix for the relations defined as(i) " x + y > 2 x" for the set given A={2,3,4,6,7}	4	CO4	L2
8.a.	Define least element, greatest element, minimal , maximal element of a relation R on A.	6	CO4	L1
b.	Draw four graphs that represent a lattice with valid reason	7	CO4	L2
c.	Define Supremum and infimum of a subset B of A where (A,R) is a poset.	7	CO4	L1
	UNIT - V			
9a	Define a group, Isomorphism between two groups.	6	CO5	L1
b	State and prove Lagrange's theorm	7	CO5	L1, L2
c	Prove that identity element and inverse of every element is unique in a group G	7	CO5	L2
10.a.	An encoding function $E:Z_2^2 \rightarrow Z_2^5$ is given by the generator matrix $G=\begin{bmatrix}1 & 0 & 1 & 1 & 0\\ 0 & 1 & 0 & 1 & 1\end{bmatrix}$ i) Determine all code words. What can be said about the error detection capability of this code? ii)'What about its error correction capability? iii) Find the associated parity check matrix H.	10	CO5	L3
b.	Write short notes on i)Encoding and Decoding of a message ii)Hamming metric iii)Generator matrix.	10	CO5	L2

Course Title: Object Oriented Programming with C++.			
Course Code : P15CS35	**Semester** : III	**L :T:P:H :** 3:2:0:5	**Credits:** 4
Contact Period: Lecture: 52 Hr, Exam: 3 Hr		**Weightage:** CIE:50; SEE:50	

Prerequisites:

1. Basic knowledge in C++ or C constructs.

Course Learning Objectives (CLOs)

The students should be able to

1. **Describe** the benefits of object oriented programming.
2. **Illustrate** the concept of redefining the operators for user defined data types.
3. **Use the** concept of templates to reduce code size.
4. **Demonstrate** adeptness of object oriented programming in developing solutions to problems demonstrating usage of data abstraction, encapsulation, and inheritance.
5. **Identify** benefits of using virtual functions.

Relevance of the Course:

This course has been designed to help the students of computer Science and Engineering to write the *higher level of abstraction* for solving real-life problems.

By studying the features of OOPs you can combines the data structures and algorithms of a software entity inside the same box.

The fastest and safest way of developing a new application is to reuse existing codes, OOPs provide *Reusable code feature*.

Course Content

Unit 1

Basic Concepts of object oriented programming:

Objects, Classes, data abstraction and encapsulation Inheritance, polymorphism, dynamic binding, message passing. Benefits of OOP's and its application. Procedure oriented programming V/S object oriented programming (OOP).

Classes and Objects:

Creation, accessing class members, defining member functions, Inline function, function overloading, default arguments, friend function, static data members and member function, arrays of objects, object as function argument, returning objects from functions, const member function, pointer to object, namespace fundamentals. 11 Hours

Unit 2

Constructor and Destructor :

Types of constructors: Parameterized constructor, multiple constructors in a class, and constructors with default arguments, copy constructor, Dynamic constructor. Dynamic initialization of objects, Destructors.

Operator Overloading :

Need of operator overloading, overloading unary operators, overloading binary operators, binary operator overloading using friend function, instream / outstream operator overloading. 11 Hours

Unit 3

Templates:
Introduction, function templates, function templates with multiple parameters, class templates, class templates with multiple parameters, overloading of template functions, member function templates.

Exception handling:
Exception handling fundamentals, Exception handling options.

STL: overview, containers, vectors, lists, maps . 10 Hours

Unit 4

Inheritance:
Introduction, defining a derived classes, single inheritance, multilevel inheritance, multiple inheritance, hierarchical inheritance, hybrid inheritance, Virtual base classes, constructors in derived classes. 10 Hours

Unit 5

Virtual Functions and Polymorphism :
Virtual function, Calling a Virtual function through a base class reference, inheriting Virtual attribute and Virtual functions, Pure virtual functions, Early vs. late binding.

C++ I/O Stream Basics :
C++ streams, stream classes, Formatted I/O. 10 Hours

Text Books:
1. Object- oriented programming with C++,E Balguruswamy, Tata McGraw Hill, 2014.
2. Mastering C++, K R Venugopal, RajkumarBuyya, Tata McGraw Hill, 2nd Edition, Tata McGraw Hill, 2013.

Reference Books:
1. The Complete Reference C++, Herbert Scheldt, 4th Edition, Tata McGraw Hill, 2012.
2. C++ Primer, Stanley B.Lippman, JoseeLajoie, 5th Edition, Pearson Education, 2009.

Course Outcomes:
The students should be able to
1. **Apply** the concepts of data abstraction and data encapsulation.
2. **Demonstrate** the concept of redefining the operators for user defined data types.
3. **Develop** the application using templates to reduce the code size.
4. **Identify and apply** the different inheritance in the given problem**.**
5. **Apply** multiple forms and I/O streams.

Model Question Paper:

Q. No	Questions	Marks	CO's	Levels
	Unit - 1			
1 a)	List and Explain the different features of OOP.	7	CO1	L1, L2
b)	Write a program illustrating class declaration, definition and accessing the members.	7	CO1	L6
c)	Write application of the scope resolution operator in C++?	6	CO1	L3
	OR			
2 a)	When will you make a function inline? Why? Write inline function to find the greatest of two numbers	6	CO1	L3
b)	Illustrate friend functions and friend classes? What are the merits and demerits of using friend functions?	8	CO1	L3
c)	Demonstrate the use of default arguments	6	CO1	L3
	Unit - 2			
3a)	Define Constructors and destructors? Explain the types of constructors.	9	CO2	L2,L2
b)	Write a program to access members of a student class using a pointer to object members.	6	CO2	L3
c)	List the limitations of overloading unary operator. How are they overcome?	5	CO2	L3
	OR			
4 a)	List the rules to overload binary operator.	7	CO2	L3
b)	The effect of a default argument can be alternatively achieved by overloading. Discuss with an example.	6	CO2	L2
c)	How many arguments are required in the definition of an overloaded unary operators / Illustrate with an example?	7	CO2	L3
	Unit - 3			
5a)	Define template. Explain the use of writing a template.	5	CO3	L1,L2
b)	Implement stack using templates.	10	CO3	L6
c)	Distinguish between lists and vectors, sets and maps.	5	CO3	L3
	OR			
6 a)	Define exception? How is an exception handled in C++?	8	CO3	
b)	Define STL and list its components? How STL algorithms are different from the conventional	6	CO3	L1,L2
c)	Identify the suitable containers for the following application i) insertion at the back of a container. Ii) Frequent insertion and deletion at both the ends of a container. iii) Frequent insertion and deletion in the middle of a container.	6	CO3	L3
	Unit - 4			
7 a)	Describe the syntax of different types of inheritance.	6	CO4	L2
b)	List the differences between inheriting a class with	6	CO4	L2

	public and private visibility mode?			
c)	Consider an example of declaring the examination result. Design threeclasses: **Student**, **Exam** and **Result**. The Student class has data members such as those representing roll no., name, etc. Create the class exam by inheriting the Student class. The Exam class adds data members representing the marks scored in six subjects. Derive the Result from the Exam class and it has its own members such as total marks. Write an interactive program to model this relationship. What type of inheritance does this model belongs to?	8	CO4	L6
OR				
8 a)	Use all types of inheritances to build class hierarchies.	7	CO4	L3
b)	With an example illustrate how base class member functions can be invoked in a derived class if the derived class also has a member function with the same name.	7	CO4	L1,L2
c)	Describe how an object of a class that contains objects of other classes created?	6	CO4	L2
Unit - 5				
9 a)	Define virtual functions. Describe the importance of virtual functions.	8	CO5	L1,L2
b)	List and Explain the various methods of performing formatted stream I/O operations.	6	CO5	L1,L2
c)	Differentiate between early and late binding by giving an example to each.	6	CO5	L3
OR				
10 a)	List the rules that must be satisfied while creating virtual functions	7	CO5	L2
b)	Write the output of the following statements? i)cout<<65 ; ii) cout.put(65); iii) cout.out('A');	6	CO5	L3
c)	Discuss the various forms of get() function supported by the input strem. How are they used.	7	CO5	L2

Course Title : Computer Organization			
Course Code : P15CS36	**Semester :** III	**L :T:P:H** : 4:0:0:4	**Credits:** 4
Contact Period: Lecture: 52 Hr, 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. Discuss basic principles of computer system, methodical treatment of machine instructions, addressing techniques, and instruction sequencing.
2. Identify the major hardware, Input/output components of the system.
3. Discuss 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:

1. Basically gives us a very fundamental understanding of how things work in computers
2. Let us know how exactly each instruction is executed. The data flow, timing analysis, memory hierarchy, trade offs between execution cycles, hardware requirements, software and hardware tradeoffs can be known
3. Useful for a code or compiler designer, as computer organisation teaches us how "hazards" and "stalls" can be avoided by changing the coding order and algorithms, and to make use of "pipeline" as best as possible.
4. Useful in study of embedded system, processor design

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, Assembly Language, Stacks

11 Hours

Unit 2

Instruction Set Architecture (Cont'd):
Subroutines, Additional Instructions

Basic Input/output: Accessing I/O/Devices, Interrupts

Software: Interaction between Assembly Language and C Language, The Operating System

10 Hours

Unit 3

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, Interconnection Standards

11 Hours

Unit 4

The Memory System: Basic Concepts, Semiconductor RAM Memories, Read-Only Memories, Direct Memory Access, Memory Hierarchy, Cache Memories, Performance Considerations, Virtual Memory, Memory Management Requirements

10 Hours

Unit 5

Arithmetic: Addition and Subtraction of Signed Numbers, Design of Fast Address, Multiplication of Unsigned Numbers, Multiplication of Signed Numbers, Fast Multiplication, Integer Division, Floating-Point Numbers and Operations, Decimal-to-Binary Conversion

10 Hours

Text Book:

1. Computer Organization and Embedded Systems, Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Naraig Manjikian ,6th Edition, TMH, 2012.

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, 2004.

Course Outcomes :

1. Understand and analyze the machine instructions and program execution.
2. Understand and Explain the I/O organisation
3. Understand and explain the memory system.
4. Apply the algorithms used for performing various arithmetic operations.

5 Understand and Explain the Concept of Basic Input/Output

Course Title : Data Structures Laboratory			
Course Code : P15CSL37	**Semester : III**	**L :T:P:H : 0:0:3:3**	**Credits: 1.5**
No. of Hours per Week: 3, Exam: 3 Hr		**Weightage: CIE:50; SEE:50**	

Prerequisites:

1. C programming language
2. Data structures knowledge
3. Fundamentals of computers

Course Content

1. Write a C program to construct a stack and to perform the following operations.
   i) Push ii) Pop iii) Display

The program should print appropriate message for stack overflow, stack underflow & stack empty.

2. Write a C program to convert and print a given valid parenthesized infix arithmetic expression to prefix expression. The expression consists of single character operands and binary operators + (Plus), - (Minus), * (Multiply), / (Divide).
3. Write a C program to evaluate a valid prefix expression using stack. Assume that the prefix expression is read as single line consisting of non negative single digit operands and binary arithmetic operations.
4. Write a C program to check whether a given string is palindrome or not using stack.

Programs on Recursion

5. Write a recursive C programs for
   a) To find larger of 'n' elements in an array
   b) To multiply two natural numbers
   c) Solving the Towers of Hanoi Problem

Programs on Queues

6. Write a C program to simulate the working of a queues using an array provide the following operation
   i) Insert ii) Delete iii) Display
7. Write a C program to simulate the working of a circular queues with items as strings. Provide the following operations
   i) Insert ii) Delete iii) Display
8. Write a C program to simulate the working of Double Ended Queue of integers using Structures. Provide the following operations
   i) Insert from front/rear end ii) Delete from front/rear end iii) Display
9. Write a C program to implement priority queues using structures (Assume a maximum of 3 queues).

Programs on Linked List

10. Write a C program using dynamic variables and pointers, to construct a Singly Linked List consisting of the following information in each node : Employee id (integer), Employee name (character string) and Department (character string). The operation to be supported are

    a) The insertion operation

    i) At the front end of the list ii) At the rear end of the list iii) At any portion in the list

    - Deleting a node based on employee id. If the specified node is not present in the list an error message should be displayed. Both the options should be demonstrated.

- Searching a node based on employee id and updates the information content. If the specified node is not present in the list an error message should be displayed. Both situations should be displayed. Displaying all the nodes in the list

11. Write a C program to construct a **Ordered Singly Linked List** and to perform the following operations

    i) Reverse a list ii) Concatenation of two lists

- Write a C program to support the following operations on a Doubly Linked List where each node consists of integers
- Create a Doubly Linked List by adding each node at the front
- Insert a new node to the right of the node whose key value is read as an input
- To delete all nodes whose info is same as key item.
- Display the contents of the list

Programs on Tree

12. Write a C program

i) To create a tree ii) To search for an item

iii) To get the exact copy of a tree iv) To display the elements

13. Write a C program

- To construct a binary search tree of integers
- To traverse the tree using In-Order, Pre-Order and Post-Order traversal method
- To display the elements

14. Write a C program

- To construct a ordered BST of items
- To insert an item into an ordered BST (No duplicates are allowed)
- To search an item in BST
- To display the elements

15. Write a C program to sort the given list of ‘n’ numbers using

i) Merge Sort ii) Quick Sort

Exercise Problems

16. Write a C program to implement queues using Singly Linked List.
17. Write a C program

    i) To create a binary tree ii) To find the height of a tree

    iii) To count the number of nodes in a tree iv) To display the contents

18. Write a C program to search en element in a given list of ‘n’ numbers using i) Linear Search ii) Binary Search

Course Outcomes:

Students will be able to :

1. Develop and implement programs on stacks
2. Develop and implement programs on recursion
3. Develop and implement programs on queues
4. Develop and implement programs on linked lists
5. Develop and implement programs on trees

Course Title : Digital Logic Design Laboratory			
Course Code : P15CSL38	**Semester :** III	**L :T:P:H :** 0:0:3:3	**Credits:** 1.5
No. of Hours per Week: 3**, Exam:** 3 Hr		**Weightage:** CIE:50; SEE:50	

Prerequisites:

1. Knowledge of Logic gates, design
2. Knowledge of Boolean algebra
3. Fundamentals of computers

Course Content

Experiment on combinational logic circuits

1. Introduction to basic gates.
2. Realization of boolean expressions

Code Converters

1. Binary to Grey code using basic gates.
2. Excess -3 to BCD using universal gates and display the result in 7 segment display.
3. Write the Verilog /VHDL code for both (a) and (b).

Experiment on data processing circuit.

1. Given any 3 and 4 variable logic expression simplify using EVM and realize the simplified logic expression using
    * 3:8 multiplexer.
    * Use suitable Decoder

2. Write the Verilog /VHDL code for an 8:1 multiplexer

Decoder and Arithmetic circuits

1. 1 bit and 2 bit Magnitude comparator using Decoder IC and NAND gates.
2. Full adder and Full subtractor using Decoder IC and NAND gates.
3. Write the Verilog/VHDL code for full subtractor and Full adder.

Encoder

1**.** Implement an 8-to-3-line encoder
2. Write the Verilog/VHDL code for encoder

Experiment on Flip-Flops

1. JK Master slave using Nand gates.
2**.** Convert JK flip flop to D and T flip flop
3. Write the Verilog/VHDL code for D Flip-Flop with positive-edge triggering

Shift Register

1. Implement a ring counter and Johnson counter using 4-bit shift register.
2. Design a 3-bit serial-in –serial out and a parallel-in –parallel out shift register using J-K flip flop
3. Design a 3 bit sequence detector and verify its operation
4. Write the Verilog/VHDL code for ring and Johnson counter.

Counters

1. Implement an asynchronous counter using decade counter and use 7 segment display to display the count 0 to n (n<=9).
2. Write the Verilog/VHDL code for decade counter.
3. Design and implement 3 bit (n<8) synchronous up counter using J-K Flip -Flop ICs.
4. Design a counter for the given sequence with lock in condition.
5. Design a 2 bit down counter using D- Flip –Flop.
6. Write the Verilog/VHDL code for sequence counter

Course Title: Aptitude and Reasoning Development - BEGINNER. (ARDB)			
Course Code : P15HU39	**Semester :** III	**L - T - P :** 0 - 0 - 2	**Credits:** NA
Contact Period: Lecture: 32 Hrs, Exam: 3 Hrs		**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

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

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. **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. **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.

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. **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. **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 Title : Additional Mathematics-I (A Bridge course for Diploma qualified students of III Sem. B. E.)			
Course Code : P15MADIP31	**Semester :** III	**L :T:P:H :** 2:2:0:4	**Credits:** NA
Contact Period: Lecture: 52 Hrs,		**Weightage: CIE:**100%, [P/NP]	

Course contents

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 n^{th} 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, 42^{nd} Ed. 2012.

References:

1.E. Kreyszig: Advanced Engineering Mathematics, John Wiley & Sons, 6^{th} Ed., 2007.
2.N.P.Bali and Manish Goyal: Engineering Mathematics, Laxmi Publishers, 7^{th} Ed., 2007.

Course Title: Indian Constitution, Human Rights and Professional Ethics (A course for Diploma qualified students of III Sem. B. E.)			
Course Code: P15HMDIP310	**Semester :** III	**L-T-P-H:** 2–0-0-0-2	**Credits:** NA
Contact Period : Lecture :26 Hr		**Weightage :** CIE:100% - [P/NP]	

COURSE CONTENT

I. Indian Constitution:

1 Introductory Part - The preamble, Fundamental rights
2 Directive principles of state policy - and fundamental duties
3 The union executive, union legislature and the union judiciary
4 The state executive, state legislature and the high court in the states
5 Special provision for scheduled caste and scheduled tribes
6 Election commission - Functions - Emergency provisions and amendment of the constitution

II. Human rights:

Aims and objectives to create responsible citizenship with awareness of human rights and latest development.

1. Protection of human rights and protection of human rights act - 1993
2. Human right - with related to rights of women, children disabled, tribal's, aged and minorities

III. Professional Ethics:

1. Aims, objects - advantages with national and international, recent development.

4th SEMESTER

Course Title: Engineering Mathematics-IV (Common to E&C, E&E, CS&E and IS&E Branches)		
Course Code: P15MAES41	**Semester: 4**	**L – T – P – H : 3 –2 – 0 – 5**
Contact Period - Lecture: 52Hrs.; Exam: 3Hrs.	**Weightage: CIE: 50%; SEE: 50%**	

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 method, Newton–Raphson method and Fixed point iteration method: Aitken's Δ^2 - process - Illustrative examples only.

Numerical solution of ordinary differential equations (ODE's): Numerical solutions of ODE's of first order and first degree – Introduction. Taylor's series method. Euler's and modified Euler's method. Runge - Kutta method of IV order. Milne's and Adams predictor & corrector methods (All formulae without proof) - Illustrative examples from engineering field. **10Hrs**

UNIT-II

Linear Algebra-II: Introduction to vector spaces – subspaces, linearly independent/dependent sets ; Bases and dimension. Linear transformation - rank and nullity. 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 method**.** Determination of largest eigen value and corresponding eigen vector by power method **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, 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 = $z+\frac{1}{z}(z\neq 0)$ and Bilinear transformations.

Complex integration: Complex line integrals. Cauchy's theorem, Cauchy's integral formula. Taylor's and Laurent's series (Statements only).and problems. Singularities, poles and residues. Cauchy's residue theorem (statement only). Simple illustrative examples.**11 Hrs**

UNIT-IV

Statistics: Brief review of measures of central tendency and dispersion. Moments, skewness and kurtosis. Curve fitting – least square method $y = a + bx; y = ax^b, y = ab^x$ *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. **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: Series solution-Frobenius method. Series solution of Bessel's equation 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. Rodrigue's formula(No Proof)- simple illustrative examples. **10 Hrs**

Text Books:

1. B.S. Grewal: Higher Engineering Mathematics, 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.

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

Engineering Mathematics-IV(P15MAES41)			
Time- 3Hrs		**Max. Marks- 100**	
Note: Answer any FIVE full questions choosing at least one full question from each unit			
Model Question Paper	Marks	CO's	Levels
UNIT- I			
1. a) Using Regula–Falsi method find the approximate root of the equation $x\log_{10}x = 1.2$ (perform three iterations)	6	1	L2
b) Use Newton – Raphson method to find a real root of $x\sin x + \cos x = 0$ near $x = \pi$. Carry out the iterations upto four decimal places of accuracy.	7	1	L2
c) Find the smallest root of the equation $x^2 + 2x - 2 = 0$, using fixed point iteration method and accelerate the convergence by Aitken's Δ^2 – method.	7	1	L2
2. a) From Taylor's series method, find $y(0.1)$ considering upto fourth degree term if $y(x)$ satisfies the equation $\frac{dy}{dx} = x - y^2, y(0) = 1$	6	1	L2
b) Using modified Euler's method find y at $x = 0.2$ given $\frac{dy}{dx} = 3x + \frac{1}{2}y$ with $y(0) = 1$ taking $h = 0.1$. Perform three iterations at each step	7	1	L3
c) Apply Milne's method to compute $y(1.4)$ correct to four decimal places given $\frac{dy}{dx} = x^2 + \frac{y}{2}$ and the data: $y(1) = 2,\quad y(1.1) = 2.2156,\quad y(1.2) = 2.4649,\quad y(1.3) = 2.7514$	7	1	L2

UNIT- II			
1. a) Define (i) vector space and (ii) subspace with suitable examples.	6	2	L2
b) Define basis of a vector space. Is the set $\{(1,1,2),(-3,1,0),(1,-1,1),(1,2,-3)\}$ a basis for the vector space $R^4(R)$?	7	2	L2
c) Define a linear transformation. Find the matrix of linear transformation $T: V_2(R) \to V_3(R)$ defined by $T(x,y) = (x+y,\ x,\ 3x-y)$ with respect to bases $\{(1,0),(0,1)\}$ and $\{(1,1,0),(1,0,1),(0,1,1)\}$	7	2	L3
2. a) Solve the system of the equations $x + y + 54z = 110,\quad 27x + 6y - z = 85,\quad 6x + 15y + 2z = 72$ by Gauss –Seidel method to obtain the numerical solution correct to three places of decimals.	6	2	L2
b) Solve the system equations $2x_1 + 8x_2 - x_3 = 24; 12x_1 + x_2 + x_3 = 31;$ $3x_1 + 4x_2 + 10x_3 = 58$ by relaxation method?	7	2	L2
c) Find the dominant eigen value and the corresponding eigen vector of A= $\begin{bmatrix} 6 & -2 & 2 \\ -2 & 3 & -1 \\ 2 & -1 & 3 \end{bmatrix}$ by Power method taking the initial eigen vector a $[1,1,1]^T$	7	2	L2

UNIT- III			
5 a) If $\phi + i\psi$ represents the complex potential of an electrostatic field where $\psi = (x^2 - y^2) + \frac{x}{x^2 + y^2}$, find ϕ and also the complex potential as a function of the complex variable z.	6	3	L2
b) Discuss the transformation $w = z + 1/z$, $z \neq 0$.	7	3	L3
c) Find the bilinear transformation which maps the points $z = \infty, i, 0$ into $w = -1, -i, 1$. Also find the invariant points of the transformation.	7	3	L3
6 a) Evaluate $\int_0^{2+i} (\bar{z})^2 dz$ along (i) the line $x = 2y$ (ii) the real axis up to 2 and then vertically to $2+i$.	6	3	L2
b) Expand $f(z) = \frac{z+1}{(z+2)(z+3)}$ as Laurent's series in the regions $(i)\ \|z\| > 3$ *and* $(ii)\ 2 < \|z\| < 3$.	7	3	L3
c) Evaluate $\int_C \frac{e^{2z}}{(z+1)^2(z-2)} dz$ where C is the circle $\|z\| = 3$ by Cauchy residue theorem.	7	3	L3
UNIT- IV			
7. a) The first four moments about an arbitrary value 5 of a frequency distribution are -4 , 22, -117 and 560. Find the skewness and kurtosis based on moments.	6	4	L1
b) Fit a best fitting parabola $y = a + bx + cx^2$, by the method of least squares for the data:	7	4	L2
c) The following data gives the age of husband (x) and the age of wife (y) in years. Find the correlation coefficient and hence obtain the regression lines. Also calculate the age of husband corresponding to wife of 16 years age :	7	4	L2
8. (a) Find the value of k such that the following distribution represents a finite probability Distribution: Also, find $P(x \leq 1), P(x > 1)$ *and* $P(-1 < x \leq 2)$	6	4	L2
(b) The number of telephone lines at an instant of time is a binomial variate with probability 0.1 that a line is busy. If 10 lines are chosen at random , what is the probability that (i) no line is busy (ii)all lines are busy (iii)at least one line is busy (iv)almost 2 lines are busy	7	4	L2
(c) State probability density function of Gaussian (normal) distribution. An analog signal received at a detector (measured in micro-volts) may be modeled as a Gaussian random variable with mean 200 and variance 256 at a fixed point of time. What is the probability that the signal will exceed 240 micro-volts?	7	4	L3

x	2	4	6	8	10
y	3.07	12.85	31.47	57.38	91.29

x	36	23	27	28	28	29	30	31	33	35
y	29	18	20	22	27	21	29	27	29	28

x	-3	-2	-1	0	1	2	3
$p(x)$	k	$2k$	$3k$	$4k$	$3k$	$2k$	k

UNIT- V

9. a) A random variable of X and Y having the following joint distribution — 6 | 5 | L2

Y / X	-3	2	4
1	0.1	0.2	0.2
2	0.3	0.1	0.1

Find (i) Marginal distributions of X and Y (ii) Cov (X,Y) (iii) Are the variables X,Y statically independent?

b) Define (i) stochastic matrix (ii) regular stochastic matrix. Find the unique probability vector for the regular stochastic matrix $\begin{bmatrix} 0 & 1 & 0 \\ 1/6 & 1/2 & 1/3 \\ 0 & 2/3 & 1/3 \end{bmatrix}$ — 7 | 5 | L3

c) Verify that $f(x,y)=\begin{cases} e^{-(x+y)}, & x \geq 0,\ y \geq 0 \\ 0, & otherwise \end{cases}$ is a probability density function of two -dimensional probability function. Evaluate $P(x<1), P(x \leq y)$ and $P(1/2 < x < 2, 0 < y < 4)$ — 7 | 5 | L2

10.a) Develop a series solution of the equation $(1+x^2)y'' + xy' - y = 0$ — 6 | 5 | L3

b) Solve the Bessel's differential equation : $x^2 \frac{d^2y}{dx^2} + x\frac{dy}{dx} + (x^2 - n^2)y = 0$) — 7 | 5 | L3

c) Express $4x^3 - 2x^2 - 3x + 8$ in terms of Legendre's polynomials. — 7 | 5 | L2

Course Title : Graph Theory & Combinatorics			
Course Code : P15CS42	**Semester : IV**	**L :T:P:H : 3:2:0:5**	**Credits: 4**
Contact Period: Lecture: 52 Hr, Exam: 3 Hr		**Weightage: CIE:50; SEE:50**	

Prerequisites: Set theory, relations and functions, basic knowledge of counting theory.

Course Learning Objectives (CLOs)

This course aims to

1. **Develop** the ability to identify different types of graphs and its properties.(L1,L2,L3)
2. **Learn** the concept of planar graphs and dual graph and Summarize the knowledge of colouring a graph and its applications.(L2)
3. **Apply** and Construct optimal Prefix code tree for the given set of codes**.(**L3)
4. **Apply** the generalized principle of inclusion and exclusion theorem, concept of generating functions to solve the given problems of counting theory(L3)
5. **Learn** the methods to solve simple recurrence relations of second and third order.(L2)

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. Find opportunity to establish results by counting a combinatorial identities in the expression forms.

Course contents

Unit 1

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. 10 Hours

Unit 2

Planar graphs, Kuratowski's two graphs, different representations of a planar graph, Euler's formula, Detection of planarity, Geometric dual.
Colouring: Cutsets, some properties of a cut-set Graph colouring, chromatic number, chromatic partitioning and chromatic polynomials. 10 Hours

Unit 3

Trees: Definitions, properties, and examples, rooted trees, trees and sorting, Weighted trees and prefix codes. Optimization: Dijkstra's shortest path algorithm, minimal spanning trees - The algorithms of Kruskal and Prim, Transport networks-Maxflow, Min-cut theorem.
10 Hours

Unit 4

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, The exponential generating function, The summation operator. 12 Hours

Unit 5

Recurrence relations: First order linear recurrence relation, the second order linear homogeneous recurrence relation with constant coefficients.
Third and higher –order Homogenous Recurrence relations, The non homogeneous recurrence relation, The method of generating functions for second order recurrence relations.
10 Hours

Text Books:

1. Discrete and Combinatorial Mathematics, Ralph.P. Grimaldi & B.V. Ramana, 5th Edition, **PHI/Pearson education. Chapter 8, 9, 10, 11, 12.**
2. **Graph Theory with Applications to Engineering and Computer Science - Narsing Deo. Chapters**-1, 2, 3, 4.1, 4.2, 5, 8.1 to 8.4.

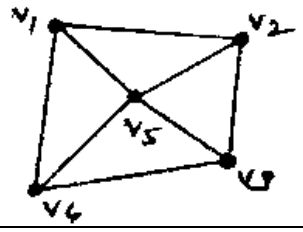
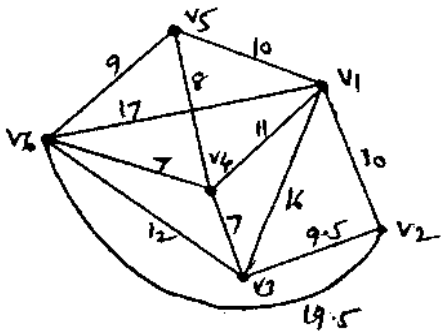
Reference Books:

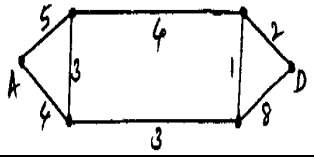
1. Graph Theory and Combinatorics, Dr. D.S. Chandrasekharaiah, Prism, 2005.
2. Introduction to Graph Theory, Chartrand Zhang, TMH, 2006.

Course Outcomes

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

1. Identify different parameters of graphs and its applications –L1
2. Understand planar graphs and its properties To detect planarity of a given graph –L2
3. Apply optimization techniques to construct a minimal spanning tree of a graph, Prefix code for a given message.-L3
4. Apply and understand the principle of inclusion and exclusion, generating functions to solve the given problem. –L3
5. Solve simple récurrence relation of second and third order.- L3

	Model Question Paper	Marks	CO's	Levels
1. a.	Define a complete graph and Prove that the number of edges in a complete graph of n vertices is always n(n-1)/2 edges	6	CO1	L1,L2
b.	i) Draw two graphs that are isomorphic. ii) Draw two graphs that are not isomorphic but have same number of vertices and same number of edges	7	CO1	L2
c.	Write short notes on Konigsberg's bridge problem	7	CO1	L1
2 a.	Define Bipartite graph .Can a bipartite graph contain a cycle of odd length, Explain	6	CO1	L1,L2
b.	Define Hamiltonian graph and Draw all edge disjoint Hamiltonian cycles of K_7	7	CO1	L1,L2
c.	Determine the |V| for the following graphs (i) regular with 15 edges (ii) 10 edges with two vertices of degree 4 and remaining of degree 3	7	CO1	L3
	UNIT II			
3 a.	Define a planar graph and prove that K_5 is non planar graph.	6	CO2	L1,L2
b.	List the properties between the graph and its dual and draw a self dual graph	7	CO2	L2
c.	Prove that a connected planar graph with "n" vertices of "e" edges has e – n + 2 regions.	7	CO2	L3
4 a.	Define chromatic number, chromatic polynomial of a graph and find the same for the graph given below. 	7	CO2	L3
b.	Define vertex connectivity and edge connectivity . Find the relation between them .	7	CO2	L1
c.	How to detect planarity in a graph?	6	CO2	L2
	UNIT III			
5a.	Prove that a tree with n vertices always has n-1 edges and also list all other properties of a tree.	7	CO3	L2
b.	Find the minimal spanning tree of the weighed graph given below using Krushkal's algorithm 	7	CO3	L3
c.	Obtain an optimal Prefix code for the message "ROAD IS GOOD" using labeled binary tree. Indicate the code	7	CO3	L3

6 a.	Explain the steps involved in Dijikstra's algorithm with an example	7	CO3	L2
b.	Find the maximum flow between the two vertices A and D in the following graph.  	6	CO3	L3
c.	Obtain an optimal Prefix code for the message "ROAD IS GOOD" using labeled binary tree. Indicate the code	7	CO3	L3
	UNIT IV			
7 a.	Define derangement. Find the number of derangements of 1,2,….10 in a line so that no even number is in its natural place	6	CO4	L1,L2
b.	An apple, a banana, a mango and an orange are to be distributed to four boys A, B, C, D. The boys A and B do not wish to have the apple; the boy C does not want the banana or mango and D returns the orange. In how many ways the distribution can be made so that no boy is displeased?	7	CO4	L3
c.	Find the number of permutations of the English letters which contain (i) exactly two (ii)atleast two (iii)exactly three and (iv)at least three of the patterns CAR, DOG, PUN and BYTE.	7	CO4	L3
8 a.	Write the sequences generated by the following functions (i)$(3+x)^3$ii)$2x^2(1-x)^{-1}$ (iii)$2x^3+1/(1-x)$	6	CO4	L2
b.	Find a generating function for each of the following sequences (i) 1,2,3,4,5,…… (ii) 1,-2,3,-4,…….. (iii) 1^3 ,2^3 ,3^3 ,4^3,……..	7	CO4	L2
c.	Find the number of ways of forming a committee of 9 students drawn from3 different classes so that students from the same class do not have an absolute majority in the committee.	7	CO4	L3
	UNIT V			
9 a.	The number of bacteria in a culture is 1000(approximately) and this number increases 255% every two hours. Use a recurrence relation to determine the number of bacteria present after one day.	6	CO5	L3
b.	Solve the recurrence relation $a_n+ a_{n-1}- 6a_{n-2}= 0$ for n>=2 given that $a_0= -1$ and $a_1= 8$	7	CO5	L3
c.	Solve the recurrence relation $a_n+ 4a_{n-1}+ 4a_{n-2}= 5(-2)^n$ for n>=2 .	7	CO5	L3
10.a	A bank pays a certain % of annual income on deposits, compounding the interest in 3months. If the amount doubles in 6years and 6months. What is the annual % of interest paid by the bank.	6	CO5	L3
b.	Solve the recurrence relation $a_n- 6a_{n-1}+ 9a_{n-2}=0$ for n>=2 given that$a_0=5$ and $a_1=12$.	7	CO5	L3
c.	Find the generating function for the recurrence relation $a_{n+1}- a_n = 3^n$ for n>=0 and $a_o=1$.Hence solve the relation. A bank pays a certain % of annual income on deposits, compounding the interest in 3months. If the amount doubles in 6years and 6months. What is the annual % of interest paid by the bank. Solve the recurrence relation $a_n- 6a_{n-1}+ 9a_{n-2}=0$ for n>=2 given that$a_0=5$ and $a_1=12$.	7 6 7	CO5 CO5 CO5	L3 L3 L3

Course Title : Theory of Computation			
Course Code : P15CS43	**Semester : IV**	**L :T:P:H : 4:0:0:4**	**Credits: 4**
Contact Period: Lecture: 52 Hr, Exam: 3 Hr		**Weightage: CIE:50; SEE:50**	

Prerequisites : Subject requires student to know about
1. Basic C programming skills
2. Elementary mathematics : Algebra – Set theory, relations.
3. Discrete mathematics
4. Data Structure

Course Learning Objectives (CLOs)

This course aims to
1. Understand the concept of finite automata theory.
2. Understand the concept of regular expression.
3. Identify the syntax of higher level language.
4. Understand and classify PDA, design PDA for CFG
5. Understand Turing machine and its applications

Relevance of the Course:
1. Theory of computation (referred to as TOC here on) lays a strong foundation for a lot of abstract areas of computer science. If you look at it from a distance, theory of computation is a very close cousin of Artificial Intelligence than say Probability or Computer vision.
2. TOC teaches about the elementary ways in which a computer can be made to think. There is a great deal of work that was made possible in the area of Natural Language Processing that involved building Finite State machines also known as Finite State Automata [1].
3. State machines are also used in certain areas of mathematics like Number theory [2].
4. Regular expressions can be beautifully represented using Non-deterministic Finite Automata.
5. Any algorithm can be expressed in the form of a finite state machine and can serve as a really helpful visual representation of the same. Sometimes, the finite state machines are easier to understand thus helping the cause furthermore.

Course Content

Unit-1

Introduction to Finite Automata

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; Equivalence and minimization of automata. 10 Hours

Unit-2

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; Decision properties of regular languages; 10 Hours

Unit-3

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; The pumping lemma for CFGs; Closure properties of CFLs. 10 Hours

Unit-4

Pushdown Automata

Definition of the Pushdown automata; The languages of a PDA; Equivalence of PDA's and CFG's; Deterministic Pushdown Automata. 12 Hours

Unit-5

Introduction to Turing Machine, Undecidability

Problems that Computers cannot solve; The turning machine; Programming techniques for Turning Machines; Extensions to the basic Turning Machines; Turing Machine and Computers. Undecidable problem that is RE; Post's Correspondence problem; 10 Hours

Text Book :

1. John E.. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman: Introduction to Automata Theory, Languages and Computation, 3rd Edition, Pearson education, 2014.

References Books:

1. Raymond Greenlaw, H.James Hoover: Fundamentals of the Theory of Computation, Principles and Practice, Morgan Kaufmann, 1998.
2. John C Martin: Introduction to Languages and Automata Theory, 3rd Edition, Tata McGraw Hill, 2007.
3. Daniel I.A. Cohen: Introduction to Computer Theory, 2nd Edition, John Wiley & Sons, 2004. Thomas A. Sudkamp: An Introduction to the Theory of Computer Science, Languages and Machines, 3rd Edition, Pearson Education, 2006

Course Outcomes:

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

1. Design finite automata
2. Apply regular expression for lexical analysis phases
3. Design grammars for various languages
4. Design push-down automata from grammars and grammar to pda
5. Design Turing machines for simple languages and design problem reductions to determine the undecidability of languages

Course Title : Analysis and Design of Algorithms			
Course Code : P15CS44	**Semester : IV**	**L :T:P:H : 4:0:0:4**	**Credits: 04**
Contact Period: Lecture: 52 Hr, Exam: 3 Hr		**Weightage: CIE:50%; SEE:50%**	

Course Learning Objectives (CLOs)

This course aims to

1. **Describe** the Fundamentals of Algorithmic Problem Solving.
2. **Illustrate** the divide and conquer technique, Transform and Conquer methods.
3. **Explain** the concept of space and time tradeoffs.
4. **Apply** the Backtracking method for solving different problems like N-Queens problem, Hamiltonian Circuit problems etc.
5. **Apply** solutions to overcome the limitations of algorithms.

Relevance of the Course:

This course has been designed to help the students of computer Science and Engineering to learn different algorithm techniques.
Obtaining efficient algorithms is very important in modern computer engineering world, where they want applications to be time and space and energy efficient.
The projects involve solving complex computational problems, for which simplistic or naive solutions may not be efficient enough. The complex problems may involve numerical data, but often they involve discrete data. Students can design the algorithms to solve the problems.

Course Content

Unit-1

Introduction
Notion of Algorithm, Fundamentals of Algorithmic Problem Solving, Graphs.
Fundamentals of the analysis of algorithm efficiency
Analysis Framework, Asymptotic notations and basic efficiency classes. Mathematical Analysis of non-Recursive Algorithms and Mathematical Analysis of Recursive Algorithms.
Brute Force: Selection sort, bubble sort 11 Hours

Unit-2

Divide and Conquer
Merge sort, Quick sort, Binary Search, Binary tree traversals and related properties,,Defective Chess Board
Decrease and Conquer
Insertion Sort, Depth First Search, Breadth First Search, Topological Sorting,
Transform and Conquer
Presorting, Balanced Search Trees, Heaps and Heap sort 11 Hours

Unit-3

Space and Time Tradeoffs
Sorting by Counting, Input Enhancement in String Matching, Hashing,
Dynamic Programming
Computing a Binomial Coefficient, Warshall's and Floyd's Algorithms
10 Hours

Unit-4

Dynamic Programming
The Knapsack Problem and Memory Functions.
Greedy Technique
Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees.
Limitations of Algorithm Power
Decision Trees, P, NP and NP-Complete Problems 10 Hours

Unit-5

Copy with the Limitations of Algorithm Power
Backtracking, Branch-and-Bound, The Traveling Salesperson problem.
Approximation Algorithms for NP-Hard Problems.
Pram Algorithm
Introduction, Computational Model, Parallel Algorithms for Prefix Computation, List Ranking, and Graph Problems 10 Hours

Text Books:
1. Introduction to the Design & Analysis of Algorithms**,** Anany Levitin, 2nd Edition, Pearson Education, 2007.
2. Fundamentals of Computer Algorithms, Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, 2nd Edition, Universities Press, 2007

Reference Books:
1. Introduction to Algorithms , Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 2nd Edition, PHI, 2006.
2. Introduction to the Design and Analysis of Algorithms A Strategic Approach, R.C.T. Lee, S.S. Tseng, R.C. Chang & Y.T.Tsai, TMH, 2005.

Course Outcomes:
1. **Analyze** algorithms and find best- case, worst-case and average – case, running times of algorithms using asymptotic notations.
2. **Apply** the decrease and conquer approach, divide and conquer approach, to solve problem.
3. **Solve** graph based problems using the different algorithm techniques.
4. **Identify and Apply** branch-and-bound technique to solve assignment problem, knapsack problem and TSP.
5. **Design** and implement algorithms using greedy strategy, decrease and conquer approach, divide and conquer approach, dynamic programming approach for a given problem.

Model Question Paper:

Q. No	Questions	Marks	CO's	Levels
	Unit - 1			
1 a)	Define an algorithm. Explain the characteristics of the algorithm.	7	CO1	L1
b)	**Show** that f(n)+g(n)=O (n2) where f(n) = 3n2- n + 4 and g(n) = nlogn+5.	7	CO1	L4
c)	**Explain** the different types of asymptotic notations with examples.	6	CO1	L2
	OR			
2 a)	Explain the various stages of algorithm design and analysis process with a flow chart.	6	CO1	L2
b)	**Write** the algorithm for addition and obtain run times for n=1,10,20,30.	6	CO1	L6
c)	**Write** an algorithm for matrix multiplication and find step count to calculate complexity	8	CO1	L6
	Unit - 2			
3a)	**Write** merge sort algorithm and also find its time efficiency.	8	CO2	L6
b)	**Apply** quick sort on data set 45, 50, 25, 10, 35, 25, 75, 30.	5	CO2	L3
c)	**Justify** the equality x=n+1 by mathematical induction where x is number of external nodes and n is number of internal nodes	7	CO2	L4
	OR			
4 a)	**Design** a Divide and Conquer algorithm for computing the number levels in a binary tree	6	CO2	L6
b)	**Design** a DFS – Based algorithm for cheking if a graph is cyclic or not.	6	CO2	L6
c)	**Show** how heap sort sorts the following sequences of keys in ascending order 22, 55, 33, 11, 99, 77, 55, 66, 54, 21, 32.	8	CO2	L6
	Unit - 3			
5a)	**Design** an algorithm for the multiplying corresponding numbers from two arrays of size n, whose values are n distinct integer numbers from 1 to n.	5	CO3	L6
b)	**Show** how distribution sorting sorts the following sequences of keys in ascending order 45, 24, 36, 59, 68, 7, 99, 17.	7	CO3	L6
c)	**Apply** Horspool's algorithm to search for the pattern AT_THAT in the text WHICH_FINALLY _HALTS._ _AT_THAT.	8	CO3	L3
	OR			

6 a)	**Construct** the closed hash table for the inputs 30, 20, 56,75,31,19 and the hash function h(K)=K mod 11.	7	CO3	L6
b)	**Distinguish Between** dynamic programming and Divide and Conquer	5	CO3	L2
c)	**Compute** the fobonacci series using dynamic programming.	8	CO3	L6
	Unit - 4			
7 a)	**Explain** the 0/1 knapsack problem algorithm with greedy concept.	6	CO4	L2
b)	**Explain** the Kruskal's algorithm with an example and analyze its time complexity.	7	CO4	L2
c)	**Compute** the optimal solution to the Knapsack instance n = 7 , m = 15, and (p1, p2, p3, p4, p5, p6, p7) = (10, 5, 15, 7, 6, 18, 3) and weights (w1, w2 ,w3, w4, w5, w6, w7) = (2,3,5,7,1,4,1).	8	CO4	L6
	OR			
8 a)	**Design** a $\Theta(n^2)$ algorithm for finding the optimal BST.	6	CO4	L2
b)	**Write** and explain Diijkstra's algorithm.	8	CO4	L2, L6
c)	**Apply** Prim's Algorithm to the given graph to find the minimum spanning tree.	7	CO4	L3
	Unit - 5			
9 a)	**Write** an algorithm of estimating the efficiency of backtracking.	7	CO5	L6
b)	**Draw** and explain the portion of the tree for 4-queens problem that is generated during backtracking	7	CO5	L5
c)	**Differentiate** between Dynamic Knapsack and Branch and Bound Knapsack problem.	6	CO5	L4
	OR			
10 a)	**Apply** the branch and bound algorithm to solve the TSP for the following the given cost matrix.	6	CO5	L3
b)	**Differentiate** between NP-complete and NP-Hard.	5	CO5	L4
c)	**Define** graph coloring? Write an algorithm, which finds m-coloring of a graph.	9	CO5	L6

Course Title : Unix System Programming			
Course Code : P15CS45	**Semester : IV**	**L :T:P:H : 3:2:0:5**	**Credits: 4**
Contact Period: Lecture: 52 Hr, Exam: 3 Hr		**Weightage: CIE:50; SEE:50**	

Prerequisites:

Basic information about Operating System.

Course Learning Objectives (CLOs)

This course aims to:

1. **Learn** the main concepts of the UNIX Operating System.
2. **Familiar** with the UNIX kernel structure and system calls.
3. **Understand** the basic UNIX commands and utilities are described in detail as are the command line wildcard and redirection facilities.
4. **Learn** how to write a shell program or shell script.
5. **Understand** how to manipulate system resources such as files, processes and system information.

Relevance of the Course:

1. This course has been designed to help the students of Computer Science and Engineering to learn the Unix Operating System.
2. Study Unix Concepts and Write a shell script or Shell program for any given problem.
3. You'll have expert knowledge of the basic Unix commands, Shell programming and how to provide a security for any type of network using Unix OS.

Course Content

Unit 1

Background and Basic Commands

Brief history, Salient features of a UNIX System, The UNIX Architecture, Internal and External Commands, Command structure, man: Browsing and Manual Pages On-line, cal: The Calendar, date: Displaying and System Date, echo: Displaying a Message, printf: An Alternative to echo, bc: The Calculator, script: Recording Your Session, passwd: Changing Your Password, who, uname: Knowing Your Machine's Characteristics, tty: Knowing Your Terminal, stty: Displaying and Setting Terminal Characteristics.

File systems: The File, The Parent-Child Relationship, The HOME Variable: The Home Directory, pwd: Checking Your Current Directory, cd: Changing the Current Directory, mkdir: Making Directories, rmdir: Removing Directories, Absolute Pathnames, Relative Pathnames, ls: Listing Directory Contents, The UNIX File System.

FILE handling Commands

cat: Displaying and Creating Files, cp: Copying a File, rm: Deleting Files, mv: Renaming Files, The lp Subsystem: Printing a File, file: Knowing the File Types, wc: Counting Lines, Words and Characters, od: Displaying Data in Octal, cmp: Comparing Two Files, comm command, diff command. 10 Hours

Unit 2

FILE Attributes

ls -l: Listing File Attributes, The -d Option: Listing Directory Attributes, File Ownership, File Permissions, chmod: Changing File Permissions, Directory Permissions, Changing File

Ownership,File Systems and Inodes, Hard Links, Symbolic Links and ln, umask: Default File and Directory Permissions, Modification and Access Times, find: Locating Files, Converting One File to Other, Compressing Files, gzip, gunzip, tar command.

The Process

Process Basics, ps: Process Status, System Processes, Mechanism of Process Creation, Internal and External Commands, Running Jobs in Background, nice: Job Execution With Low Priority, Killing Processes with Signals, Job Control, at and batch: Execute Later, cron: Running Jobs Periodically, time: Timing Processes.

Simple Filters

The Sample Database, pr: Paginating Files, head: Displaying the Beginning of a File, tail: Displaying the End of a File, cut: Slitting a File Vertically, paste: Pasting Files, sort: Ordering a File, uniq: Locate Repeated and Non repeated Lines, tr: Translating Characters, An Example: Displaying a Word-count List, grep: Searching for a pattern. 11 Hours

Unit 3

SHELL Programming : The Shell's Interpretive Cycle, Pattern Matching The Wild-cards, Escaping and Quoting, Redirection: The Three Standard Files, /dev/null and /dev/tty: Two Special Files, Pipes, tee: Creating a Tee, Command Substitution, Shell Variables, Environment Variables, Aliases (bash and ksh), Command History (bash and ksh). Shell Scripts, read and read-only commands, Using Command Line Arguments, exit and Exit Status of Command, The Logical Operators && and || -Conditional Execution, The if Conditional, Using test and [ ] to Evaluate Expressions, The case Conditional, expr: Computation and String Handling, $0: Calling a Script by Different names, while: Looping, for: Looping with a List, set and shift: Manipulating the Positional Parameters, The here Document (<<), trap: Interrupting a Program, Debugging Shell Scripts with set -x, eval: Evaluating Twice. 10 Hours

Unit 4

Introduction to Unix System Program: UNIX and ANSI Standards: The ANSI C Standard, The ANSI/ISO C++ Standards, Difference between ANSI C and C++, The POSIX Standards, The POSIX.1 FIPS Standard, The X/Open Standards. UNIX and POSIX APIs: The POSIX APIs, The UNIX and POSIX Development Environment, API Common Characteristics.

UNIX File APIs: General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs, Symbolic Link File APIs. 11 Hours

Unit 5

UNIX Processes: The Environment of a UNIX Process: Introduction, main function, Process Termination, Command-Line Arguments, Environment List, Memory Layout of a C Program, Shared Libraries, Memory Allocation, Environment Variables, setjmp and longjmp Functions, getrlimit, setrlimit Functions, UNIX Kernel Support for Processes.

Process Control : Introduction, Process Identifiers, fork, vfork, exit, wait, waitpid, wait3, wait4 Functions, Race Conditions, exec Functions, Changing User IDs and Group IDs, Interpreter Files, system Function, Process Accounting, User Identification, Process Times, I/O Redirection. Process Relationships: Introduction, Terminal Logins, Network Logins, Process Groups. 10 Hours

Text Books:

1. UNIX Concepts and Applications by Sumitabha Das, 4 edition, Tata McGraw Hill, 2014.
2. Terrence Chan: UNIX System Programming Using C++, First edition, Prentice Hall India, 2011.
3. W. Richard Stevens: Advanced Programming in the UNIX Environment, Second Edition, Pearson education, 2011

Reference Books:

1. UNIX Operating System, fifth edition**,** Jerry Peek , Pearson education.
2. UNIX and Shell Programming, Behrouz A. Forouzan and Richard F.Gilberg, Thomson, 2011.
3. Unix & Shell Programming, M.G. Venkatesh murthy, Pearson Education, 2011.

Course Outcomes

1. **Understand** the overview of the UNIX operating system and Shell Programming.
2. **Analyze** and **Write** shell scripts effectively to solve any problem.
3. **Design** scripts to automate common tasks in UNIX system and to guard against malicious intents against the UNIX operating system.
4. **Implement** the required visual and functional features using the appropriate technologies.
5. **Apply** UNIX commands to solve common problems.

[MODEL QUESTION PAPER]

Course Code: P15CS45 **Max Marks: 100**

Note: *Answer **FIVE** full questions, selecting **ONE** full question from each **unit.***

			Marks	CO's	Levels
		UNIT - I			
1.	a.	Explain the salient features of Unix OS.	5	CO 1	L2
	b.	Explain the Unix architecture with a neat diagram.	5	CO 1	L2
	c.	Explain with an example i)script ii)bc iii)who iv)uname v)tty	10	CO1	L2
2	a.	Define file? List the different types of files? Write an parent – child relationship.	6	CO 1	L2
	b.	Explain with an example i)pwd ii)cat iii)wc iv)cmp v)od	5	CO 1	L2
	c.	Explain with an example i)Absolute and Relative pathname ii)mv iii)lp iv)file v)diff vi)more	9	CO1	L2
		UNIT - II			
3	a.	Explain ls command with its options	5	CO 2	L2
	b.	Explain Absolute Permission and Relative Permission	6	CO 2	L3
	c.	Explain the following commands with an example i)Umask ii)touch iii)pr iv)tail v)uniq	5	CO2	L2
	d.	Explain the Mechanism of Process creation.	4	CO2	L2
4	a.	Define process? Explain ps command with all its options.	5	CO 2	L1,L2
	b.	Explain with an example hard link and symbolic link.	4	CO 2	L2
	c.	Explain the Basic file attributes.	4	CO2	L3
	d.	Explain the sort command with all its options.	7	CO2	L2
		UNIT - III			
5	a.	Define shell? Explain with an example wild cards used for pattern matching.	6	CO3	L1,L2
	b.	Explain the common environment variables	6	CO3	L2
	c.	Write an shell script to find the largest of two numbers	4	CO3	L4
	d.	Explain – i) /dev/null ii) /dev/tty	4	CO3	L2
6	a.	Describe the following positional parameter i)$0 ii)$# iii)$? iv)$$ v)$! Vi)$*	6	CO 3	L3
	b.	Explain Three Standard files with an example.	8	CO 3	L2
	c.	Write a shell script to find a factorial of a number and display the date, list the files and directories, display the machine characteristics and attribute of file.	6	CO3	L5
		UNIT - IV			
7	a.	List any four major differences between ANSI C and K&R C	4	CO 4	L4
	b.	Write a C/C++ POSIX compliant program that prints the POSIX defined configurations options supported on any given system using feature test macros.	10	CO 4	L4
	c.	Briefly Explain the POSIX Standards	6	CO4	L2
8	a.	List and Briefly explain the commonly occurred error status and their meanings	6	CO4	L1,L2
	b.	Explain the following General file API's i)Open ii)creat iii)read iv)Link	8	CO4	L2
	c.	Define file locking and record locking? Explain with its prototype	6	CO4	L1,L2

		description how an fcntl API is used for file and record locking.			
		UNIT - V			
9	a.	List the eight ways for a process to terminate	5	CO 5	L3
	b.	Explain the memory layout of a C program with a neat diagram.	5	CO 5	L2
	c.	Explain the different exec functions with their prototypes.	10	CO5	L2
10	a.	Explain the following functions with prototype i)fork ii)vfork iii)wait iv)exit	6	CO 5	L2
	b.	Explain with an neat diagram Unix Kernel Support for processes.	8	CO 5	L2
	c.	Define Race Condition? Write a C/C++ program to illustrate Race Condition	6	CO 5	L1,L4

Course Title : Microprocessor			
Course Code : P15CS46	**Semester : IV**	**L :T:P:H : 4:0:0:4**	**Credits: 3**
Contact Period: Lecture: 52 Hr, Exam: 3 Hr		**Weightage: CIE:50; SEE:50**	

Prerequisites: Any programming language. Logic Design**.**

Course Learning Objectives (CLOs):

This Course aims to;

1. Study in-depth the hardware and software included in micro computer systems.
2. 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.
3. Learn the programming concept of 8086 programming using (Microsoft assembler) MSAM Assembler.
4. Implementation of 8086 programming using string instruction set and I/O programming.
5. Study the min and max mode operation, Learn to interface memory to 8086 and working of interrupt controller.

Relevance of the Course:

Ability to debug and interface the external devices to the processor according to the user requirements to create products and solutions for the real time problems. This is a foundation courses

Course Content

Unit-1

8086 Architecture: CPU architecture, Internal Operation, Machine Language Instructions addressing modes, Instruction formats, Instruction Execution Timing **Assembler language programming :** Assembler instruction format. 10 Hours

Unit-2

Assembler language programming : Data transfer Instructions, arithmetic instructions, 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, alignment directives, value returning Attribute Operators. 11 Hours

Unit-3

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. 11 Hours

Unit-4

Byte and String Manipulation: String Instructions, REP prefix, table translation.

I/O programming: Fundamental I/O considerations, Programmed I/O, Interrupt I/O, Block transfers and DMA. 10 Hours

Unit-5

System Bus Structure : Basic 8086/8088 configurations – Minimum mode, Maximum mode, System Bus Timing, Memory Interfacing, Interrupt Priority Management – Interrupt System based on Single 8259A, Interrupt System Based on Multiple 8259As. 10 Hours

Text Book:

1. Yu-Cheng Liu, Glenn A.Gibson, ―Microcomputer Systems: The 8086 / 8088 Family Architecture, Programming and Design□, Second Edition, Prentice Hall of India, 2007

Reference Books :

1. **The Intel Microprocessors**, Barry.B.Brey, PHI Publication, 8th edition, 2009
2. **The Intel Microprocessor Family: Hardware and Software Principles and Applications, James L. Antonakos, Thomson,** 2007.
3. **Microprocessors and Interfacing, Programming and Hardware Doughlas V.Hall**,TMH, 2012

Course Outcomes

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

1. **Understand** the architecture of 8086 microprocessor. (Unit-I)
2. **Apply** 8086 instruction set for the given problems (Unit-II)
3. **Develop** different modules & link them. (Unit-III)
4. **Apply** string instruction set and I/0 Interrupt in 8086 programming (Unit-IV)
5. **Understand** min & max mode of 8086 and design memory interfacing. (Unit-V)

Model Question Paper	Marks	CO's of	Levels
Unit-1 Qn.1 a. What are the advantages of memory segmentation ? Illustrate the concept of segmentation with neat diagram b. Explain the flags of 8086 processor using suitable example. Qn.2 a. Discuss the following addressing modes with examples: 1 i. Direct ii. Register indirect iii. Base plus index iv. Immediate v. Scaled indexed b. Describe the memory map of a PC system, with a neat diagram.	20	1	L1
Unit-2 Qn.3 a. Write an ALP using 8086 instructions to generate and add the first l0 even numbers and save the numbers and result in memory location Num and Sum b. Describe the following instruction with suitable examples: i) PUSH ii) MUL iii) IN iv) AAA Qn.4 a. Bring out the importance of XLAT instruction using a suitable program b. Write an ALP using 8086 instructions to count the numbers of zeros in a given 8 bit number and store the result in memory location 'Res'.	20	2	L3
Unit-3 Qn.5 a. Explain the following assembler directives: i) Assume; ii) Proc iii) Ends iv) DB b. Briefly explain any four bit test instructions. Qn.6 a. Explain public and extrn directives of assembler and write ALP to read data through keyboard using external procedure and save the keycode in public data segment b. Write a program that uses function to display strings on output device.	20	3	L6

Unit-4 Qn.7 a. Write a program that uses function to display strings on output device. b. Write a program to sort N numbers in ascending order Qn.8 a. Differentiate between memory mapped I/O and I/O mapped I/O (isolated I/O) b. List the action taken by microprocessor when an interrupt occur	20	4	L3
Unit-5 Qn.9 a. Explain in brief the functions of 8086 pins. b. With neat timing diagram, explain memory read cycle. Qn.10 a. Interface 512 KB RAM to 8088 MP using 64 KB RAM using 3:8 decoder with starting address of memory as 80000H. Clearly mention decoding logic and memory map. b. Explain memory bank selection in 8086 and mention the number of memory bank in 80x86 MPs.	20	5	L1

Course Title: Analysis and Design of Algorithms Laboratory			
Course Code : P15CSL47	**Semester : IV**	**L :T:P:H : 0:0:3:3**	**Credits: 1.5**
No. of Hours per Week: 3, Exam: 3 Hr		**Weightage: CIE:50; SEE:50**	

Prerequisites:

1. Basic Knowledge of C language
2. Knowledge of Analysis and design of algorithm

Course Content

1. Problems using brute force technique
   - Recursive linear search
   - Selection sort
2. Problems using divide and conquer technique
   - Merge sort
   - Quick sort
   - Recursive binary search
3. Problems using decrease and conquer technique
   - Insertion sort
   - Topological sorting
   - DFS
   - BFS
4. Problems using transform and conquer technique
   - Heap sort
5. Problem on string matching
   - Horspool's algorithm
6. Problems using dynamic programming technique
   - Binomial coefficient
   - Warshall's algorithm
   - Floyd's algorithm
   - Knapsack algorithm
7. Problems using greedy technique
   - Prim's algorithm
   - Kruskal's algorithm
   - Dijkstra's algorithm
8. Problems using backtracking technique.
   - N queens problem

Course Outcomes:

Students will be able to :

1. **Develop** searching and sorting using the algorithm techniques such as decrease and conquer, divide and conquer, transform and conquer technique.
2. **Implement** solutions to the graph based problems using the algorithm techniques such as decrease and conquer, dynamic programming, and greedy technique.
3. **Identify and Apply** algorithm Techniques to solve a given contextual problems.
4. **Design** and **implement** algorithms for the realistic problems

Course Title : Object Oriented Programming with C++ Laboratory			
Course Code : P15CSL48	**Semester** : IV	**L :T:P:H :** 0:0:3:3	**Credits:** 1.5
No. of Hours per Week: 3, Exam: 3 Hr		**Weightage:** CIE:50; SEE:50	

Prerequisites:

1. Basic Knowledge of C and C++ Programming language
2. Knowledge of Object oriented programming concepts

Sl.No.	**Course Content/Experiments**
1	Programs using Class and Objects
2	Programs using Constructors and Destructors
3	Programs on Operator overloading
4	Programs on Inheritance
5	Programs on Polymorphism and Virtual functions
6	Programs on Templates – Function and Class Templates
7	Programs on Exception handling
8	Programs on Stream handling

Course Outcomes:

Students will be able to :

1. Students will be able to design, implement, test, debug, and document programs in C++.
2. Students will be able to solve the real world problems using object oriented concepts.
3. Students will be able to use the rich features provided in programming language to develop solutions to simple problems.
4. Students will be able to identify classes, objects, members of a class and the relationships among them needed for a specific problem.
5. Students will be able to use the object oriented concepts in software development.
6. Students will be able to build good quality software using object-oriented techniques.
7. Students will be able to breakdown simple programming goals into object-oriented components, propose and evaluate different designs for solving problems using knowledge of fundamental programming techniques

Course Title : Aptitude and Reasoning Development - INTERMEDIATE (ARDI)			
Course Code : P15HU49	**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. Identify the assumptions, analyse the given argument and evaluate the inference.
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

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. **6 hrs**

Unit – II

Analytical reasoning 2: The basics of logic, some informal tips, **Assumptions**– Some standard categories of assumptions, Where is the assumption invalid?, **Forcefulness of arguments**– Preliminary screening, Will the results really follow?, Is the result really desirable?, Are the argument and suggested course of action properly related?, **Evaluating Inferences**– A study of key words, How to avoid confusion?, **Evaluating given course of action**– Problem -solution relation, Fact-follow-up action relationship. **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. **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. **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. **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 statements and identify the assumptions and infer the results based on the arguments or premises. 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 Title : Additional Mathematics-II (A Bridge course for Diploma qualified students of IV Sem. B. E.)			
Course Code : P15MADIP41	**Semester :** IV	**L :T:P:H :** 2:2:0:4	**Credits:** NA
Contact Period: Lecture: 52 Hr,		**Weightage: CIE:**100%, [P/NP]	

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.

Course Title : Environmental Studies			
Course Code: P15EVDIP410	**Semester : I/II**	**L-T-P-H: 2–0-0-2**	**Credits: NA**
Contact Period : Lecture :26 Hr		**Weightage :CIE:100% - [P/NP]**	

Prerequisites:

The student should have undergone the course on Environmental Studies (Code: P15EV19/29)

a) ***Course Learning Objectives (CLO) :***

At the end of the course the students should be able to:

1 Explain the need for Environmental Management
2 Implement standard data like water, wastewater and air pollution.
3 Demonstrate the use of standard data to compare with the field data.
4 Choose appropriate data to protect environmental.
5 Design environmental amenities based on the needs.

b) Relevance of the Course

Environmental Studies is a foundation course in BE (Environmental Engineering) program, that builds the program design and implementation competence in student through choice of appropriate areas.

The course aims at developing the understanding variations in water, wastewater and air pollution and also the ability to build new ideas.

Course Content

Unit – I

Environment – Definition, Ecosystem – Balanced Ecosystem, Human activities – Food Shelter, Economic and Social Security. Transportation activities, Environmental impact Assessment, Sustainable Development. **6 Hrs.**

Unit – II

Natural Resources – Water resources – Availability and Quality aspects, Mineral Resources, Forest Wealth, Material Cycles – (Carbon, Nitrogen and Sulphur Cycles) Water borne diseases, water induced diseases, Fluoride problem in drinking water. 5**Hrs.**

Unit – III

Energy – Different types of energy, Conventional and Non-Conventional sources – Hydro Electric, Fossil fuel based, Nuclear, Solar, geothermal, tidal, wind, Biomass and Bio-gas. Hydrogen as an alternative future source of energy. **5 Hrs.**

Unit – IV

Environmental Pollution and their effects. Water pollution, Land pollution, Noise pollution Public Health aspects. Current Environmental issues of importance: Population Growth Climate change and Global warming – Effect, Urbanizations industrialization. **5 Hrs.**

Unit –V

Acid Rain, Ozone layer depletion, Animal Husbandry. Environmental protection – Role o Government, Legal aspects, initiatives by Non-Governmental Organizations (NGO Environmental Education, Women Education. **5 Hrs.**

Text Book:

1)Environmental Studies – Benny Joseph – Tata McGraw Hill – 2005

References:

1)Principles of Environmental Science and Engineering – .VenugopalaRao P, Prentice Hall 2005
2)Elements of Environmental Science and Engineering – Meenakshi P, Prentice Hall of India, 2
3)Environmental Studies – Anil Kumar D.C, New age International Publishers, 2007