R.V.COLLEGE OF ENGINEERING

(Autonomous Institution Affiliated to VTU, Belagavi)
R.V. Vidyaniketan Post, Mysore Road
Bengaluru – 560 059

Bachelor of Engineering (B.E.)
Scheme and Syllabus for III & IV Semesters

2016 SCHEME

INFORMATION SCIENCE & ENGINEERING

Department Vision

To be the hub for innovation in Information Science & Engineering through Teaching, Research, Development and Consultancy; thus make the department a well known resource centre in advanced, sustainable and inclusive technology.

Department Mission

ISE1: To enable students to become responsible professionals, strong in fundamentals of information science and engineering through experiential learning.

ISE2: To bring research and entrepreneurship into class rooms by continuous design of innovative solutions through research publications and dynamic development oriented curriculum.

ISE3: To facilitate continuous interaction with the outside world through student internship, faculty consultancy, workshops, faculty development programmes, industry collaboration and association with the professional societies.

ISE4: To create a new generation of entrepreneurial problem solvers for a sustainable future through green technology with an emphasis on ethical practices, inclusive societal concerns and environment.

ISE5: To promote team work through inter-disciplinary projects, co-curricular and social activities**.**

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

PEO1: To provide adaptive and agile skills in Information Science and Engineering needed for professional excellence / higher studies /Employment, in rapidly changing scenarios.

PEO2: To provide students a strong foundation in basic sciences and its applications to technology.

PEO3: To train students in core areas of Information science and Engineering, enabling them to analyze, design and create products and solutions for the real world problems, in the context of changing technical, financial, managerial and legal issues.

PEO4: To inculcate leadership, professional ethics, effective communication, team spirit, multi-disciplinary approach in students and an ability to relate Information Engineering issues to social and environmental context.

PEO5: To motivate students to develop passion for lifelong learning, innovation, career growth and professional achievement.

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO	Description
PSO1	Recognize and appreciate the principles of theoretical foundations, data organization, data communication, security and data analytical methods in the evolving technology
PSO2	Learn the applicability of various system software for the development of quality products in solving real-world problems with a focus on performance optimization
PSO3	Demonstrate the ability of team work, professional ethics, communication and documentation skills in designing and implementation of software products using the SDLC principles

Lead Society:

Program Criteria

All programs seeking accreditation from the Computing Accreditation Commission of ABET must demonstrate that they satisfy all of the specific Program Criteria implied by the program title.

PROGRAM CRITERIA FOR COMPUTER SCIENCE AND SIMILARLY NAMED COMPUTING PROGRAMS

Lead Society: CSAB

Computer Science	1. Coverage of fundamentals of algorithms, data structures, software design, concepts of programming languages and computer organization and architecture.[CS] 2. An exposure to a variety of programming languages and systems.[CS] 3. Proficiency in at least one higher-level language. [CS] 4. Advanced course work that builds on the fundamental course work to provide depth. [CS]
Information Technology	1. The core information technologies of human computer interaction, information management, programming, networking, web systems and technologies. [IT] 2. information assurance and security.[IT] 3. System administration and maintenance. [IT] 4. System integration and architecture. [IT]

R.V.COLLEGE OF ENGINEERING

(Autonomous Institution Affiliated to VTU, Belagavi)

R.V. Vidyaniketan Post, Mysore Road

Bengaluru – 560 059

Bachelor of Engineering (B.E.)

Scheme and Syllabus for III & IV Semesters

2016 SCHEME

INFORMATION SCIENCE

&

ENGINEERING

Abbreviations

Sl. No.	Abbreviation	Meaning
1.	**VTU**	Visvesvaraya Technological University
2.	**BS**	Basic Sciences
3.	**CIE**	Continuous Internal Evaluation
4.	**CS**	Computer Science and Engineering
5.	**CV**	Civil Engineering
6.	**CHY**	Chemistry
7.	**EC**	Electronics and Communication Engineering
8.	**EE**	Electrical and Electronics Engineering
9.	**ES**	Engineering Science
10.	**HSS**	Humanities and Social Sciences
11.	**ME**	Mechanical Engineering
12.	**PHY**	Engineering Physics
13.	**SEE**	Semester End Examination
14.	**MAT**	Engineering Mathematics

INDEX

R V COLLEGE OF ENGINEERNG, BENGALURU-560 059
(Autonomous Institution Affiliated to VTU, Belagavi)

THIRD SEMESTER CREDIT SCHEME								
Sl. No.	Course Code	Course Title	BoS	CREDIT ALLOCATION				
				L	T	P	SS	Total Credits
1.	16MA31A	Laplace Transforms, Fourier Series and Linear Algebra	Maths	3	1	0	0	4
2.	16EB32	Biology for Engineers	BT	2	0	0	0	2
3.	16IS33	Discrete Mathematical Structures	ISE	3	1	0	0	4
4.	16IS34	Computer Organization and Architecture	ISE	4	0	0	1	5
5.	16IS35	Data Structures and File Structures	ISE	3	0	1	1	5
6.	16IS36	Object Oriented Programming using C++	ISE	3	0	1	1	5
7.	16DCS37	Bridge Course C Programming	CSE	2	0	0	0	0
Total number of Credits				18	2	2	3	25
Total Number of Hours / Week				20	4	4	12	

FOURTH SEMESTER CREDIT SCHEME								
Sl. No.	Course Code	Course Title	BoS	CREDIT ALLOCATION				
				L	T	P	SS	Total Credits
8.	16MA41A	Graph & Probability Theory	Maths	3	1	0	0	4
9.	16ET42	Environmental Technology	BT	2	0	0	0	2
10.	16IS43	Operating Systems	ISE	3	1	0	0	4
11.	16IS44	Design and Analysis of Algorithms	ISE	3	0	1	1	5
12.	16IS45	Micro controllers and Embedded Systems	ISE	3	0	1	1	5
13.	16IS46	Unix System Programming	ISE	3	0	1	1	5
14.	16HS47	Professional Practice-II (Communication Skills and Professional Ethics)$	HSS	0	0	1	0	1
15.	16DMA48	Bridge Course Mathematics	Maths	2	0	0	0	0
Total number of Credits				17	2	4	3	26
Total Number of Hours / Week				19	4	8	12	

Semester: III		
LAPLACE TRANSFORMS, FOURIER SERIES AND LINEAR ALGEBRA (COMMON TO CS, IS)		
Course Code: 16MA31A		**CIE Marks:** 100
Credits: L:T:P:S: 3:1:0:0		**SEE Marks:** 100
Hours: 36L+12T		**SEE Duration:** 03Hrs
Course Learning Objectives: The students will be able to		
1	Adequate exposure to basics of engineering mathematics so as to enable them to visualize the applications to engineering problems.	
2	Analyze periodic phenomena using concept of Fourier series.	
3	Apply Laplace transform technique to solve differential equation which includes the concept of convolution.	
4	Use basic terminology of linear algebra in Euclidean spaces, including linear independence, spanning, basis, rank, nullity, subspaces, and linear transformations.	
5	Students will become capable to participate and succeed in competitive exams.	

UNIT-I	
Laplace Transform: Existence and uniqueness of Laplace Transform (LT), Transform of elementary functions, RoC. Properties of LT - Linearity, change of scale and first shifting. Transform of function - multiplied by t^n, division by t, derivatives and integral. LT of periodic function, Heaviside unit step function, Unit impulse function. Heaviside shift (second shift) theorem.	**07 Hrs**
UNIT-II	
Inverse Laplace Transform: Definition, properties of inverse Laplace transform, evaluation using different methods. Convolution theorem, problems. Application to solve ordinary linear differential equations and simultaneous differential equations.	**07 Hrs**
UNIT-III	
Fourier Series: Introduction, periodic function, even and odd functions, properties. Special waveforms - square wave, half wave rectifier, saw-tooth wave and triangular wave. Dirichlet‟s conditions, Euler‟s formula for Fourier series. Fourier series for functions of period 2L (particular cases) - problems. Half Range Fourier series- Construction of Half range cosine and sine series. Parseval‟s theorem for Root mean square value of a function(without proof). Complex form of Fourier series.	**08 Hrs**
UNIT-IV	
Linear Algebra - I: Vector spaces, subspaces, Linear dependence, basis and dimension, four fundamental sub-spaces. Rank of a matrix, rank and nullity theorem, Orthonormal Bases, Gram-Schmidt process, QR-factorization.	**07 Hrs**
UNIT-V	
Linear Algebra - II: Linear Transformation, Geometric meaning, Matrix representation of linear transformation, Projection, reflection, rotation of linear transformation. Eigen values, Eigen vectors, Geometric meaning of Eigen values and Eigen vectors, Algebraic and Geometric multiplicity of Eigen values, Diagonalization of a Matrix, Singular Value Decomposition.	**07 Hrs**

Course Outcomes: After completing the course, the students will be able to	
CO1:	Demonstrate the fundamental concepts - in Fourier Series, Laplace transforms and Basics of Linear Algebra.
CO2:	Identify - appropriate methods to find the Fourier constants, Rank, Nullity, Orthonormal basis, Linear transformation and properties of Laplace transforms.
CO3:	Apply - the acquired knowledge to construct the Half range Fourier series, Finding Laplace transforms and Inverse Laplace transforms for some functions, Eigen values and Eigen vectors of matrix.
CO4:	Evaluate - Complex form of Fourier series, solutions of differential equations with initial and boundary conditions using Laplace transforms, QR factorization, Diagonalization of matrix and Singular value decomposition.

Reference Books	
1.	Higher Engineering Mathematics, B.S. Grewal, 40th Edition, 2007, Khanna Publishers, ISBN: 81-7409-195-5.
2.	Linear Algebra and Its Applications, Gilbert Strang, 4th Edition, 2006, Cengage Learning India Edition, ISBN: 81-315-0172-8.
3.	Advanced Engineering Mathematics, Erwin Kreyszig, 9th Edition, 2007, John Wiley & Sons, ISBN: 978-81-265-3135-6.
4.	A Text Book of Engineering Mathematics, N.P Bali & Manish Goyal, 7th Edition, 2010, Lakshmi Publications, ISBN: 978-81-7008-992-6.

Continuous Internal Evaluation (CIE); Theory (100 Marks)

CIE is executed by way of quizzes (Q), tests (T) and Assignment. A minimum of three quizzes are conducted and each quiz is evaluated for 10 marks adding up to 30 marks. All quizzes are conducted online. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three also. The three tests are conducted for 50 marks each and the sum of the marks scored from three tests is reduced to 60. The marks component for assignment is 10. The total marks of CIE are 100.

Semester End Evaluation (SEE); Theory (100 Marks)

SEE for 100 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the complete syllabus. Part – B consists of five main questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have sub questions. The question from Units I, IV and V have no internal choice. Units II and III have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom‟s taxonomy level.

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

Low-1 Medium-2 High-3

BIOLOGY FOR ENGINEERS **(Theory)** **(Common to BT, CS and IS)**		
Course Code:16EB32		**CIE Marks:** 50
Credits: L:T:P:S: 2:0:0:0		**SEE Marks:** 50
Hours : 23L		**SEE Duration:** 2Hrs
Course Learning Objectives: The students will be able to		
1	Familiarize themselves with basic biological concepts	
2	Get an interdisciplinary vision of biology and engineering	
3	Gain an understanding that the design principles from nature can be translated into novel devices and structures	
4	Gain an appreciation for how biological systems can be designed and engineered to substitute natural systems	

UNIT-I	
Cells and Biomolecules: Structure and function of plant, animal and microbial cell. Stem cells: types and applications. Biomolecules: Carbohydrates, lipids, Proteins, Nucleic acids, Enzymes, Hormones, Vitamins.	**06 Hrs**
UNIT II	
Human physiology: Digestive, Blood circulatory, Respiratory, Excretory and Nervous system. Structure and Function of sense organs- Skin, Ear, Eye, Tongue and Nose.	**05 Hrs**
UNIT III	
Photosynthesis: Chloroplasts, Light reaction and Dark reaction. Plants as Bio inspirations: Bionic leaf and Photovoltaic cells.	**04 Hrs**
UNIT IV	
Bio inspired Engineering: Lotus leaf effect (Super hydrophobic and self-cleaning surfaces), Echolocation of bats and whales (Ultrasonography), Human brain (Artificial neural networks), Natural recognition receptors (Biosensors), Silk from insects and spiders (High performance fibers and flexible medical tapes), Plant burrs (Velcro).	**05 Hrs**
UNIT V	
Biomimetics: Medical implants: Orthopaedic, Dental, Cardiovascular, Optical and Auditory. Artificial senses: Electronic Nose and Electronic Tongue.	**03 Hrs**

Course Outcomes: After completing the course, the students will be able to	
CO1:	Remember and explain the fundamentals of biology
CO2:	Describe the basic principles of design in biological systems.
CO3:	Comprehend how biological principles have served as a source of inspiring innovation
CO4:	Address the problems associated with the interaction between living and non-living materials and systems

Reference Books	
1.	Principles of Biochemistry, Donald Voet, Judith G. Voet, Charlotte W. Pratt, 4th Edition, 2012, John Wiley & Sons, ISBN-10: 1118092449, ISBN-13: 978-1118092446
2.	Principles of Physiology, Pramanik Debasis, 5th Edition, 2015, Jaypee Brothers Medical Publishers, ISBN-10: 9351529290, ISBN:13: 978-9351529293
3.	Biomimetics: Biologically Inspired technologies, Yoseph Bar-Cohen, 1st Edition, 2005, CRC press, ISBN: 9780849331633
4.	Bioinspired Engineering, Jenkins, C.H., 2011, Momentum press, ISBN-10: 1606502239 ISBN:13: 978-1606502235

Continuous Internal Evaluation (CIE); Theory (50 Marks)

CIE is executed by way of quizzes (Q), tests (T) and Assignment. A minimum of three quizzes are conducted and each quiz is evaluated for 05 marks adding up to 15 marks. All quizzes are conducted online. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three also. The three tests are conducted for 25 marks each and the sum of the marks scored from three tests is reduced to 30. The marks component for Assignment is 05. The total marks of CIE are 50.

Semester End Evaluation (SEE); Theory (50 Marks)

SEE for 50 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 10 marks covering the complete syllabus. Part – B consists of five main questions, one from each unit for 8 marks adding up to 40 marks. Each main question may have sub questions. The question from Units I, IV and V have no internal choice. Units II and III have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom‟s taxonomy level.

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

High-3 Medium-2 Low-1

DISCRETE MATHEMATICAL STRUCTURE **(Theory)** **(COMMON TO CSE, ISE)**		
Course Code: 16IS33		**CIE Marks:** 100
Credits: L:T:P:S: 3:1:0:0		**SEE Marks:** 100
Hours: 36L+24T		**SEE Duration:** 03Hrs

Course Learning Objectives: The students will be able to	
1	Gain intense foundational introduction to fundamental concepts in discrete mathematics.
2	Interpret, identify, and apply the language associated with logical structure, sets, relations and functions, modular arithmetic.
3	Write and interpret logical statements using quantifiers.
4	Understand and apply the concepts of group and coding theory and applications.

UNIT-I	
Fundamental Principles of Counting: The Rule of Sum and Product, Permutations, Combinations, The Binomial Theorem, Combinations with repetition. **Mathematical Induction, Recursive Definitions:** Different proof of techniques, Method of Mathematical induction and examples, Recursive definition and examples.	**09 Hrs**
UNIT-II	
Recurrence Relations and Fundamentals of Logic: First order linear recurrence relation –Formulation problems and examples, Second order linear homogenous recurrence relations with constant coefficients, The non – Homogenous recurrence relations. Basic connectives and truth tables, Logical equivalence: The laws of logic, Rules of inference. Open Statement, Quantifiers, Definitions and the proofs of theorems.	**09 Hrs**
UNIT-III	
Relations: Properties of relations, Composition of Relations, Partial Orders, Hasse Diagrams, Binary heap as a Partial order, Equivalence Relations and Partitions. **Functions:** Functions-plain, One-to-one, onto functions, Function composition and Inverse function, computational complexity, analysis of algorithms.	**09 Hrs**
UNIT-IV	
Introduction to Finite Automaton and Languages: Formal language as a set, Mathematical Notations, Definitions and examples of DFA, Languages recognized by Finite Automata, Finding Equivalence classes and minimization of DFA.	**08 Hrs**
UNIT-V	
Groups theory: Definition, Examples and Elementary properties, Abelian groups, Homomorphism isomorphism, cyclic groups, cosets and Lagrange‟s theorem. **Coding theory:** Elementary coding theory, the hamming metric, the parity Check and Generator Matrices.	**09 Hrs**

Course Outcomes: After completing the course, the students will be able to	
CO1:	Reason mathematically about basic data types and structures (such as numbers, sets, graphs, and trees) used in computer algorithms and systems
CO2:	Model and analyze computational processes using analytic and combinatorial methods
CO3:	Use abstract structures to represent discrete objects and their interrelationships
CO4:	Apply the mathematical concepts learned to various areas of computer science

Reference Books	
1.	Discrete and Combinatorial Mathematics- An Applied Introduction, Ralph P. Grimaldi and B V Ramana, 5th Edition – 2007. Pearson Education, Asia ISBN: 978-81-7758-424-0
2.	Discrete Mathematical Structures with Applications to Computer Science, J.P. Tremblay and R. Manohar, 35th reprint 2008. Tata – McGraw Hill, ISBN: 13:978-0-07-463113-3
3.	Discrete Mathematics and its Applications, Kenneth H. Rosen, 6th Edition, Sixth reprint 2008. Tata – McGraw Hill, ISBN:13:978-0-07-064824-1
4.	Elementary Discrete Mathematics, C. L. Liu and D P Mohapatra, 6th Edition. Tata- McGraw Hill ISBN:10:0-07-066913-9
5.	An Introduction To Formal Languages & Automata, Peter Linz, 4th Edition, 2007, Narosa Publishing House, ISBN: 978-1-4496-1552-9.

Continuous Internal Evaluation (CIE); Theory (100 Marks)

CIE is executed by way of quizzes (Q), tests (T) and Assignment. A minimum of three quizzes are conducted and each quiz is evaluated for 10 marks adding up to 30 marks. All quizzes are conducted online. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three also. The three tests are conducted for 50 marks each and the sum of the marks scored from three tests is reduced to 60. The marks component for assignment is 10. The total marks of CIE are 100.

Semester End Evaluation (SEE); Theory (100 Marks)

SEE for 100 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the complete syllabus. Part – B consists of five main questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have sub questions. The question from Units I, IV and V have no internal choice. Units II and III have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom‟s taxonomy level.

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

Low-1 Medium-2 High-3

COMPUTER ORGANIZATION AND ARCHITECTURE (Theory)		
Course Code: 16IS34		**CIE Marks:** 100
Credits: L:T:P:S: 3:1:0:1		**SEE Marks:** 100
Hours: 36L+24T		**SEE Duration:** 03Hrs

Course Learning Objectives: The students will be able to	
1	Understand the main components of computers and the basic principles of their operation and Interconnection Structures that realize the architecture.
2	Analyze the relationship between hardware design and instruction set architecture.
3	Explore and apply the methods for evaluating and comparing processor performance.
4	Provide a comprehensive coverage of Parallel Processing and Multi-core Architecture.

UNIT-I	
A Top-Level view of Computer Function and Interconnection: Computer Components, Computer Function, Interconnection structures, Bus Interconnection, PCI Express. **Cache Memory:** Computer Memory System Overview, Cache Memory Principles, Elements of Cache Design, Pentium 4 and ARM Cache Organization.	**09 Hrs**
UNIT-II	
Internal Memory: Semiconductor Main Memory, Error Correction. **External Memory:** Magnetic Disk, RAID, Solid State Drives, Optical Memory, Magnetic Tape. **Input/Output:** External Devices, I/O Modules, Programmed I/O, Interrupt-Driven I/O, Direct Memory Access, I/O Channels and Processors, External Interface: InfiniBand.	**09 Hrs**
UNIT-III	
Computer Arithmetic: The Arithmetic and Logic Unit, Integer Representation, Integer Arithmetic, Floating-Point Representation, Floating-Point Arithmetic. **Processor Structure and Function:** Processor Organization, Register Organization, Instruction Cycle.	**09 Hrs**
UNIT-IV	
Instruction Pipelining: An Overview of Pipelining, Data Hazards: Forwarding versus Stalling , Control Hazards. **Control Unit Operation:** Micro-Operations, Control of the Processor, Hardwired Implementation.	**09 Hrs**
UNIT-V	
Instruction-level parallelism And superscalar processors: Overview, Superscalar versus Super pipelined Constraints , Design Issues **Parallel Processing:** Multiple Processor Organizations, Symmetric Multiprocessors, Cache Coherence and the MESI Protocol, Multithreading and Chip Multiprocessors, Clusters, Non-uniform Memory Access.	**08 Hrs**

Course Outcomes: After completing the course, the students will be able to	
CO1:	Demonstrate the relationship between software and hardware and focuses on the fundamental concepts that are the basis for current computer design.
CO2:	Describe various data representation and explain how arithmetic and logical operations are performed by computers.
CO3:	Articulate design issues in the development of processor or other components that satisfy design requirements
CO4:	Conceptualize, evaluate and design single and parallel processor systems to meet desired needs, within the realistic constraints specific to the field.

Reference Books	
1.	Computer Organization and Architecture, William Stallings, PHI, 9th Edition ISBN:10: 013293633X
2.	Computer Organization and Design, David A. Patterson and John L. Hennessy, Elsevier, 4th Edition, 2012, ISBN: 9780123747501.
3.	Computer Organization, Carl Hamacher, Z Vranesic& S Zaky, 5th Edition, 2012, McGraw Hill ISBN: 9781259005275
4.	Multi-core Programming, Shameem Akhter and Jason Roberts, 2006, Intel Press, ISBN: 0-9764832-4-6

Continuous Internal Evaluation (CIE); Theory (100 Marks)

CIE is executed by way of quizzes (Q), tests (T) and Self-Study(S). A minimum of three quizzes are conducted and each quiz is evaluated for 10 marks adding up to 30 marks. All quizzes are conducted online. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three also. The three tests are conducted for 50 marks each and the sum of the marks scored from three tests is reduced to 50. The marks component for Self-study is 20. The total marks of CIE are 100.

Semester End Evaluation (SEE); Theory (100 Marks)

SEE for 100 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the complete syllabus. Part – B consists of five main questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have sub questions. The question from Units I, IV and V have no internal choice. Units II and III have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom‟s taxonomy level.

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

Low-1 Medium-2 High-3

DATA STRUCTURES AND FILE STRUCTURES (Theory & Practice)		
Course Code: 16IS35		**CIE Marks:** 100+50
Credits: L:T:P:S: 3:0:1:1		**SEE Marks:** 100+50
Hours: 36L+24T		**SEE Duration:** 03Hrs+03Hrs

Course Learning Objectives: The students will be able to	
1	Learn fundamentals of data structures and their applications essential for programming/problem solving.
2	Analyse and apply linear data structures: Stack, Queues, Lists to solve problem.
3	Analyse and apply non- linear data structures: Graphs, Trees to solve problem.
4	Learn fundamentals of file structures and its programming essentials.

UNIT-I	
Performance Analysis: Mathematical Background, Running time analysis and calculations; Asymptotic notations, **Abstract Data Types**–ADTs, **Stacks:** Stack ADT; Stack applications: Infix to postfix conversion, Evaluation of postfix expression, Recursion	**07 Hrs**
UNIT-II	
Queues: Queue ADT; Circular queues; Priority queues; Queue applications: A Mazing Problem, Multiple Stacks and Queues, **Linked Lists:** List ADT; Linked implementation of Stacks, Queues; Header node; Circular linked lists; Doubly linked lists; Applications of Linked lists: Polynomial Manipulation, Multiple Precision Arithmetic.	**08 Hrs**
UNIT-III	
Hashing: Symbol table; Hash function; Collision resolution techniques: Open addressing, Separate chaining, **Graph:** Graph ADT; Preliminaries; Matrix and Adjacency List representation of Graphs, **Tree:** Tree ADT; Preliminaries; Binary Trees; Representation of Binary Trees; Application of Binary Tree: Evaluation of Expression, Symbol Table construction.	**08 Hrs**
UNIT-IV	
Search trees: Search tree ADT; Binary Search Tree; 2-3 tree; Application of Search Tree: Dictionary, Longest Prefix Matching, **Heaps:** Heap ADT, Binary heap, Binomial heap.	**07 Hrs**
UNIT-V	
Files Structures: Concepts of fields: Records and Files; Sequential, Indexed and Relative/Random File Organization; Indexing structure for index files; Hashing for direct files, B-Tree.	**06 Hrs**

LABORATORY EXPERIMENTS	
Students are required to implement following programs using C/C++. **Part A (Compulsory)** **1.** Implementation of integer stack ADT using arrays **2.** Implementation of integer queue ADT using arrays **3.** Implementation of integer List ADT **4.** Implementation of Graph ADT using List **5.** Implementation of tree ADT using List **6.** Implementation of basic operation on Files. **7.** Implementation of simple hash algorithm for files with records. **Part B** **At-least one application from each of the following group.** **Application of Stack** • Implementation of Infix to Postfix conversion • Implementation of Infix to Prefix conversion	

• Implementation of postfix evaluation • Implementation of prefix evaluation **Application of Queue** • Implementation of Priority queue program using array. • Implementation of multiple stacks and queues • Implementation of Johnsons Algorithm • Implementation of maze problem **Application of List** • Implementation of sparse matrix multiplication. • Implementation of polynomials operations (addition, subtraction) using Linked List. • Implementation of Linked Lists menu driven program (stack and queue) • Implementation of Double ended queue using Linked Lists. **Application of Graph & Tree** • Implementation of construction of expression tree using postfix expression. • Implementation of various operations on tree like – copying tree, mirroring a tree, counting the number of nodes in the tree, counting only leaf nodes in the tree. • Implementation of dictionary using Binary Search Tree • Implementation of Longest Prefix Matching. • Implementation of Binary Heap program **Application of File Structures** • Implementation of Open addressing technique • Implementation of separate chaining with linked list • Implementation of B-Tree • Implementation of secondary index on set of Records	

Course Outcomes: After completing the course, the students will be able to	
CO1:	Comprehend how choice of data structure and file structure influences the performance of programs.
CO2:	Analyse the running time of operations like searching, insertion, deletion traversing on various data structure and file structure.
CO3:	Implement and demonstrate program design and implementation competence through the choice of appropriate data structure and file structure.
CO4:	Apply appropriate data structure and file structure in solving real world problems from various domains.

Reference Books	
1.	Data Structures using C and C++, YedidyahLangsam Moshe J. Augenstein and Aaron M. Tanenbaum, 2nd Edition, 2009. PHI/Pearson, ISBN 10: 0131997467 ISBN 13: 9780131997462
2.	Introduction to algorithms, Cormen, Thomas H., Leiserson, Charles E., Rivest, Ronald L. and Clifford Stein; 3rd Edition, 2009, MIT Press, ISBN: 9780262033848
3.	An Introduction to Data Structures with Applications, Jean Paul Tremblay and Paul G Sorenson , 2nd Edition, 2002, Tata McGraw Hill, ISBN: 10: 0070651574 ISBN 13: 9780070651579
4.	Data Structures and Program Design in C++, R.Kruse, C.L Tondo and B.Leung, 2nd Edition, 2009, Pearson Education, ISBN:10: 013288366X ISBN-13: 978-0132883665

Continuous Internal Evaluation (CIE): Total marks: 100+50=150

Theory – 100 Marks

CIE is executed by way of quizzes (Q), tests (T) and Self-Study(S). A minimum of three quizzes are conducted and each quiz is evaluated for 10 marks adding up to 30 marks. All quizzes are conducted online. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three also. The three tests are conducted for 50 marks each and the sum of the marks scored from three tests is reduced to 50. The marks component for Self-study is 20. The total CIE for theory is 100.

Laboratory- 50 Marks

The Laboratory session is held every week as per the time table and the performance of the student is evaluated in every session. The average of marks over number of weeks is considered for 40 marks. At the end of the semester a test is conducted for 10 marks. Total marks for the laboratory is 50.

Semester End Evaluation (SEE): Total marks: 100+50=150

Theory – 100 Marks

SEE for 100 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the complete syllabus. Part – B consists of five main questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have sub questions. The question from Units I, IV and V have no internal choice. Units II and III have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom‟s taxonomy level.

Laboratory- 50 Marks

Experiment Conduction with proper results is evaluated for 40 marks and Viva is for 10 marks. Total SEE for laboratory is 50 marks.

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

Low-1 Medium-2 High-3

OBJECT ORIENTED PROGRAMMING USING C++ (Theory & Practice)		
Course Code: 16IS36		**CIE Marks:** 100+50
Credits: L:T:P:S: 3:0:1:1		**SEE Marks:** 100+50
Hours: 36L+24T		**SEE Duration:** 03Hrs+03Hrs

Course Learning Objectives: The students will be able to	
1	Explain the benefits of object oriented design and understand when it is an appropriate methodology to use.
2	Apply the object-oriented programming paradigm, focusing on the definition and use of classes along with the fundamentals of object-oriented design.
3	Design object oriented software solutions for small systems involving multiple classes and objects. Implement solutions in C++.
4	Test and debug C++ implementations. Apply generic programming for real time applications

UNIT-I	
Introduction to Object Oriented Programming Concepts: Principles of object oriented programming: Procedure oriented programming Vs object oriented programming, Underlying concepts of object oriented programming, Benefits and applications of object oriented programming. The Origins of C++, A Closer Look at the I/O Operators, The bool Data Type, The C++ Headers, Namespaces, C++ programming fundamentals, , Introducing C++ Classes & objects, Constructors and Destructors, The C++ Keywords, The General Form of a C++ program, C++ I/O basics, Portability, Compiling & Linking, Pointers, Reference Types, Managing Memory in C++, Storage Classes	**07 Hrs**
UNIT-II	
Classes & Objects: Discovering Classes, Interfaces, Encapsulation, Abstraction, Member Functions, Classes and Objects, Object has an interface, Structures and Classes, Unions and Classes, Friend Functions, Friend Classes, Inline Functions, Static Class Members, Static Data, Static Member Functions, Constructors and Destructors, The Scope Resolution Operator, Nested Classes, Local Classes, Passing Objects to Functions, Returning Objects, Object Assignment and Accessing Data Fields. Introduction to Object Oriented Design & The Unified Modelling Language.	**07 Hrs**
UNIT-III	
Inheritance and Polymorphism: Inheritance, Access Control in derived classes, Encapsulation & protected access, Advanced operations with inheritance, Function Overloading and Default arguments, Polymorphism , operator overloading, Virtual functions and Abstract Classes	**07 Hrs**
UNIT-IV	
Streams and Files, Exception Handling: Streams classes, Stream Errors, Disk File I/O with streams, file pointers, error handling in file I/O with member function, overloading the extraction and insertion operators, memory as a stream object, command line arguments, and printer output. Exception Handling Fundamentals, Catching Class Types, Using Multiple catch Statements, Handling Derived-Class Exceptions, Exception Handling Options, Catching All Exceptions, Understanding terminate( ) and unexpected( )	**07 Hrs**
UNIT-V	
Template Functions and Classes – Generic Programming: Template Functions, compile-time Polymorphism, Template Classes, Template Linked List, Nontype Template Arguments, Setting Behavior Using Template Arguments, **Standard Template Library (STL) of C++:** Inline Member Function and Template, C++ Standard Library- The "String" Class, The Fundamental Containers, The Stack and Queue Adapters, Template Class "vector", Template Class "map", Template Class "list", Iterators and Algorithms The Standard Function Library and The Standard C++ Class Library	**08 Hrs**

LABORATORY EXPERIMENTS	

Students are required to implement following programs using C/C++.

Part A (Compulsory)

1. Encapsulation: Objects & Classes - C++ object, class and data abstraction fundamentals.
   The C++ programming skills that should be acquired in this lab session:
   To implement the basic principles of encapsulation, data hiding, class, object, object instance and message, use keyword public and private, use constructor and destructor, use inline function, use object packaging.
2. Encapsulation: The C++ programming abilities that should be acquired in this lab session:
   - Class and arrays.
   - Pointer within class.
   - Pointer of the objects.
   - Static member variable.
   - Pointer of object to another object: list and linked list examples.
   - Class and strings.
   - Nesting the classes.
   - New and delete operators.
   - This pointer.
   - Default methods.
3. Inheritance: C++ object/class inheritance, extending the classes. The C++ programming abilities that should be acquired in this lab session:
   - Implement inheritance concept, base class (parent class), derived class (child class).
   - Implement and use pre-processor directive to avoid the multiple inclusion of the same file.
   - Implement class hierarchy.
   - Scope operator (::).
   - protected, private and public keywords.
4. Inheritance: The C++ Inheritance programming abilities**:** Able to implement and use:
   - Method vs function.
   - Constructor Execution Order.
   - Destructor Execution Order.
   - Pointer, Array and Objects.
   - Friend functions and classes, keyword friend.
5. Inheritance: Multi inheritance - C++ object/class multi inheritance, generic types. The C++ inheritance programming abilities: Able to implement and use:
   - Multiple inheritances.
   - Duplicated methods issue.
   - Duplicated member variables issue.
6. Generic Programming: The C++ multi inheritance programming knowledge should be acquired: Able to design and implement:
   - Parameterized type - Function template.
   - Parameterized type - Class template.
   - Generic Programming applications.

7. Polymorphism - C++ polymorphism, virtual functions. The C++ programming skills that should be acquired: Able to implement and use:
   - Polymorphism concept, Virtual function, Late and early binding.
     - Operators overloading.
     - Functions overloading.

• C++ Formatted I/O - Standard C++ formatted input/output - cin, cout, cerr etc. The C++ formatted I/O programming skills: use various member functions for C++ formatted I/O, use various stream manipulators for C++ formatted I/O. 8. C++ File I/O - Standard C++ file input/output - read, write, create file streams. The C++ file input/output programming skills: • Use the ifstream, ofstream and fstream class objects. • Use a sequential access file – Read and Write member functions. • Use a random access file – Read and Write member functions. • Be familiar with other file I/O member functions. 9. Storage Classes: const, volatile, static, auto, register - const, static, auto, register, volatile, mutable etc. The C++ storage classes programming abilities: • Use storage classes: auto, extern, static and register. • Use the const for variable and member function. • Use the volatile keyword. • External and internal linkages terms. 10. C / C++ Exception Handling - Simple and structured exception handling (SEH) - try-catch-throw etc. The C & C++ programming skills that should be acquired in this session: • Use C++ exception handlings in general. • Use the assert() function. • Use try-throw-catch, structured exception handling 11. C++ Typecasting - The simple/automatic and advanced C++ type castings - simple cast, up/down/cross cast. The C and C++ programming skills that should be acquired: • Basic of type casting. • Use the automatic type casting. • Use static_cast, const_cast, dynamic_cast and reinterpret_cast. • Use the explicit keyword. 12. C++ Namespaces - The C++ namespaces - std, using directive etc. The C++ programming abilities that should be acquired: • Use and create the namespace. • Use namespace alias, anonymous/un-named, using directive and std. • Using C Standard Library in C++ programs (C++ wrappers). • Understand and appreciate the Standard C++ library.	

Course Outcomes: After completing the course, the students will be able to	
CO1:	Exhibit program design and implementation competence through the choice of appropriate object oriented concept and explain the benefits of the same.
CO2:	Design and analyze the programming applications using object-oriented programming paradigm, focusing on the definition and use of classes along with the fundamentals of object-oriented design.
CO3:	Envision the solutions for real-time problems using Object Oriented concepts.
CO4:	Understand and apply advanced features of C++ specifically stream I/O, templates and operator overloading which influences the performance of programs.

Reference Books	
1.	The Complete Reference C++, Herbert Schildt, 4th Edition, 2011, McGrawHill, ISBN: 9780070532465.
2.	C++ How to Program, Paul Deitel and Harvey Deitel, 8th Edition, 2012, Prentice Hall, ISBN: 9780132990448.
3.	Big C++, Cay S. Horstmann, Timothy Budd, 1st Edition, 2009, Wiley India (P.) Ltd ISBN: 9788126509201.
4.	Thinking in C++ - Introduction to standard C++, Bruce Eckel, http://iacs-courses.seas.harvard.edu/courses/cs207/resources/TIC2Vone.pdf Vol 1, 2nd Edition, 2002 , Pearson, ISBN:10: 8131706613
5.	Problem Solving with C++, Walter Savitch, 9th Global Edition, 2015, Addison-Wesley ISBN: 13:9781292018249.

Continuous Internal Evaluation (CIE): Total marks: 100+50=150

Theory – 100 Marks

CIE is executed by way of quizzes (Q), tests (T) and Self-Study(S). A minimum of three quizzes are conducted and each quiz is evaluated for 10 marks adding up to 30 marks. All quizzes are conducted online. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three also. The three tests are conducted for 50 marks each and the sum of the marks scored from three tests is reduced to 50. The marks component for Self-study is 20. The total CIE for theory is 100.

Laboratory- 50 Marks

The Laboratory session is held every week as per the time table and the performance of the student is evaluated in every session. The average of marks over number of weeks is considered for 40 marks. At the end of the semester a test is conducted for 10 marks. Total marks for the laboratory is 50.

Semester End Evaluation (SEE): Total marks: 100+50=150

Theory – 100 Marks

SEE for 100 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the complete syllabus. Part – B consists of five main questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have sub questions. The question from Units I, IV and V have no internal choice. Units II and III have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom‟s taxonomy level.

Laboratory- 50 Marks

Experiment Conduction with proper results is evaluated for 40 marks and Viva is for 10 marks. Total SEE for laboratory is 50 marks.

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

Low-1 Medium-2 High-3

C PROGRAMMING (BRIDGE COURSE) **(Theory)**		
Course Code: 16DCS37		**CIE Marks:** 100
Credits: L:T:P:S : 2:0:0:0 (Audit Course)		**SEE Marks:** 100
Hours: 24L		**SEE :** 03 Hrs
Course Learning Objectives: The students will be able to		
1	Develop arithmetic reasoning and analytical skills to apply knowledge of basic concepts of programming in C.	
2	Learn basic principles of problem solving through programming.	
3	Write C programs using appropriate programming constructs adopted in programming.	
4	Solve complex problems using C programming.	

UNIT-I	
Introduction to Reasoning, Algorithms and Flowcharts: Skill development – Examples related to Arithmetical Reasoning and Analytical Reasoning. Fundamentals of algorithms and flowcharts.	**02 Hrs**
Introduction to C programming: Basic structure of C program, Features of C language, Character set, C tokens, Keywords and Identifiers, Constants, Variables, Data types.	**01 Hrs**
Handling Input and Output operations: Reading a character, Writing a character, Formatted input/output functions, Unformatted input/output functions.	**02 Hrs**
UNIT-II	
Operators and Expressions: Arithmetic operators, Relational operators, Logical Operators, Assignment operators, Increment and decrement operators, Conditional operators, Bit-wise operators, Arithmetic expressions, evaluation of expressions, Precedence of arithmetic operators, Type conversion in expressions, Operator precedence and associativity.	**02 Hrs**
Programming Constructs **Decision Making and Branching:** Decision making with „if" statement, Simple „if" statement, the „if…else" statement, nesting of „if…else" statements, The „else if" ladder, The „switch" statement, The „*?: '* operator, The „goto" statement. **Decision making and looping**: The while statement, the do statement, The „for" statement, Jumps in loops**.**	**03 Hrs**
UNIT-III	
Arrays: one dimensional array, Declaration of one dimensional array. Initialization of one dimensional array, Two dimensional arrays, Initializing two dimensional arrays.	**02 Hrs**
Character Arrays and Strings : Declaring and Initializing String Variables, Reading Strings from Terminal, Writing strings to screen, Arithmetic Operations on characters, String operations using with and without String handling functions.	**02 Hrs**
UNIT-IV	
User-defined functions : Need for User Defined Functions, Definition of functions, Return values and their types, Function calls, Function declaration, Category of functions, Nesting of functions, Functions with arrays, Storage classes.	**03 Hrs**
Structures and Unions: Introduction, Structure definition, Declaring structure variables, Accessing structure members, Structure initialization, Copying and comparing structure variables, Arrays of structure, Arrays within structures, Structures and functions, Unions.	**03 Hrs**
UNIT – V	
Pointers : Introduction , Accessing the address of a variable, Declaring and initializing of pointer variables, Accessing a variable using pointers, Chain of pointers, Pointer expressions, Pointer increments and scale factor, Pointers and arrays, Pointers and character strings.	**03 Hrs**
File Managements in C: Basic concepts of files, Defining and opening a file, closing of a file, Input/output operations on files.	**01 Hrs**

Course Outcomes: After completing the course, the students will be able to	
CO1:	Understand and explore the fundamental computer concepts and basic programming principles like data types, input/output functions, operators, programming constructs and user defined functions.
CO2:	Analyze and Develop algorithmic solutions to problems.
CO3:	Implement and Demonstrate capabilities of writing „C" programs in optimized, robust and reusable code.
CO4:	Apply appropriate concepts of data structures like arrays, structures, and files to implement programs for various applications.

Reference Books:	
1.	Programming in C, P. Dey, M. Ghosh, 1st Edition, 2007, Oxford University press, ISBN -13: 9780195687910.
2.	The C Programming Language, Kernighan B.W and Dennis M. Ritchie, 2nd Edition, 2005, Prentice Hall, ISBN -13: 9780131101630.
3.	Turbo C: The Complete Reference, H. Schildt, 4th Edition, 2000, Mcgraw Hill Education, ISBN-13: 9780070411838.
4.	Understanding Pointers in C, Yashavant P. Kanetkar, 4th Edition, 2003, BPB publications, ISBN-13: 978-8176563581.

Scheme of Continuous Internal Evaluation:
CIE is executed by way of quizzes (Q), tests (T) and Assignment. A minimum of three quizzes are conducted and each quiz is evaluated for 10 marks adding up to 30 marks. Faculty may adopt innovative methods for conducting quizzes effectively. The two tests are conducted and each test is evaluated for 30 marks adding up to 60 marks The marks component for assignment is 10. The total marks of CIE are 100.

Scheme of Semester End Examination:
SEE for 100 marksis executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the complete syllabus. Part – B consists of five main questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have sub questions. The question from Units I, IV and V have no internal choice. Units II and III have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom"s taxonomy level.

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

Low-1 Medium-2 High-3

Semester: IV		
GRAPH THEORY AND PROBABILITY THEORY (COMMON TO CS, IS)		
Course Code: 16MA41A		**CIE Marks:** 100
Credits: L:T:P:S: 3:1:0:0		**SEE Marks:** 100
Hours: 36L+12T		**SEE Duration:** 03Hrs

Course Learning Objectives: The students will be able to	
1	Learn the fundamental concepts in graph theory in view of its applications in modern science.
2	Learn to understand and create mathematical proofs, including an appreciation of its significance in computer science.
3	Use the concepts of Graph theory in subsequent courses in the design and analysis of algorithms, computability theory, software engineering and computer systems.
4	Apply concepts of the theory of probability in study of random phenomena, analyzing and interpreting data that involves uncertainties.

UNIT-I	
Graph: Introduction, basic terminology, simple graph, degree of a vertex, types of graphs, subgraphs and isomorphic graphs, operations of graphs and connectivity, Eulerian and Hamiltonian graph, shortest path problems, representation of graphs in network flows.	**07 Hrs**
UNIT-II	
Trees: Introduction, trees and their properties, types of trees, spanning tree, Cayley‟s theorem (with proof)-problems, minimum spanning tree. Binary tree, properties of binary trees, m-array trees. **Planar Graphs:** Definition, Euler‟s formula (with proof), applications and problems, Kuratowski‟s theorem, matching.	**07 Hrs**
UNIT-III	
Colorings: Introduction, coloring of graphs, vertex coloring, chromatic number, chromatic index, chromatic polynomial, chromatic partitioning, Five color theorem (with proof), four color theorem (without proof). Edge coloring of graphs.	**08 Hrs**
UNIT-IV	
Probability: Baye‟s rule, Random Variables, Discrete and continuous. Probability mass function, probability density function, cumulative density function, mean, variance, standard deviation-problems. Joint probability distributive function, Discrete and continuous, mean, covariance and correlation.	**07 Hrs**
UNIT-V	
Probability Distributions: Some standard discrete and continuous Distributions - Binomial, Poisson, Normal, Exponential and Geometric distributions. **Sampling Theory:** Sampling, sampling distributions, standard errors, student‟s t-distribution, chi-square distribution as a test of goodness of fit.	**07 Hrs**

Course Outcomes: After completing the course, the students will be able to	
CO1:	Demonstrate - the knowledge of fundamental concepts in Graph theory and Probability theory.
CO2:	Apply - models of Graph theory, Probability theory respectively to solve problems of connectivity and uncertainty.
CO3:	Analyze - graphs, trees and random phenomena occurring in real life situations using Graph theory and Probability theory respectively.
CO4:	Interpret - the models of Graph theory, Probability theory for real life and engineering problems.

Reference Books	
1.	Graph Theory, Frank Harary, Narosa Publishing House, ISBN: 978-81-850-1555-2.
2.	Probability and Statistics with Reliability, Queuing and Computer applications, Kishor S. Trivedi, 2nd Edition, Wiley Publication, ISBN: 978-0-471-33341-8.
3.	Graph Theory-Modeling, Applications and Algorithms, Geir Agnarsson & Raymond Greenlaw, Pearson Education, 2008, ISBN: 978-81-317-1728-8.
4.	Theory and Problems of Probability, Seymour Lipschutz & Marc Lars Lipson, 2nd Edition, Schaum‟s Outline Series, ISBN: 0-07-118356-6.

Continuous Internal Evaluation (CIE); Theory (100 Marks)

CIE is executed by way of quizzes (Q), tests (T) and Assignment. A minimum of three quizzes are conducted and each quiz is evaluated for 10 marks adding up to 30 marks. All quizzes are conducted online. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three also. The three tests are conducted for 50 marks each and the sum of the marks scored from three tests is reduced to 60. The marks component for assignment is 10. The total marks of CIE are 100.

Semester End Evaluation (SEE); Theory (100 Marks)

SEE for 100 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the complete syllabus. Part – B consists of five main questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have sub questions. The question from Units I, IV and V have no internal choice. Units II and III have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom‟s taxonomy level.

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

Low-1 Medium-2 High-3

ENVIRONMENTAL TECHNOLOGY (Theory)		
Course Code: 16ET32/16ET42		**CIE Marks:** 50
Credits: L:T:P:S: 2:0:0:0		**SEE Marks:** 50
Hours: 25L		**SEE Duration:** 02Hrs

Course Learning Objectives: The students will be able to	
1	Understand the various components of environment and the significance of the sustainability of healthy environment.
2	Recognize the implications of different types of the wastes produced by natural and anthropogenic activity.
3	Learn the strategies to recover the energy from the waste.
4	Design the models that help mitigate or prevent the negative impact of proposed activity on the environment

UNIT-I	
Introduction: Ecosystem – Types and structure of ecosystem. Components of environment, Environmental education, Environmental act & regulations. Global environmental issues, ISO 14000, Environmental Impact Assessment and Challenges.	**05 Hrs**
UNIT II	
Environmental pollution: Causes, effects and control measures of Air, noise and land pollution. Air Pollution. Clean air act, Pollution standard index. Indoor air quality. Global atmospheric change - Global warming, Acid rain &Ozone depletion and their controlling measures.	**05 Hrs**
UNIT III	
Water pollution and management: Pollutants in surface & ground water, water borne diseases. Water purification systems: physical & chemical treatment - aeration, solids separation, settling operations, coagulation, softening, filtration, disinfection, The common technologies for purification of drinking water - Ultraviolet radiation treatment, Reverse Osmosis. Rain water harvesting, water recycling, STP plant.	**05 Hrs**
UNIT IV	
Renewable energy sources and technology for generation of energy: Different types of energy, conventional sources & non-conventional sources of energy, solar energy, wind energy, hydro electric energy, Geothermal Energy, Nuclear energy, Fossil Fuels & Biomass energy.	**05 Hrs**
UNIT V	
Solid waste management: Types, causes, control and processing. Typical generation rates, estimation of solid waste quantities, factors that affect generation rates. Management - On site handling, collection, storage and processing techniques, ultimate disposal, landfills. Reduction and recycling of waste – waste to composite, energy.	**05 Hrs**

Course Outcomes: After completing the course, the students will be able to	
CO1:	Identify the components of environment and exemplify the detrimental impact of anthropogenic activities on the environment.
CO2:	Differentiate the various types of wastes and suggest appropriate safe technological methods to manage the waste.
CO3:	Aware of different renewable energy resources and can analyse the nature of waste and propose methods to extract clean energy.
CO4:	Adopt the appropriate recovering methods to recover the essential resources from the wastes for reuse or recycling.

Reference Books	
1.	Introduction to environmental engineering and science, Gilbert, M.M., Pearson Education. 2nd Edition, 2004, ISBN: 8129072770.
2.	Environmental Engineering, Howard S. Peavy, Donald R. Rowe and George Tchobanoglous, 2000, McGraw Hill Series in water resources and Environmental Engg., ISBN: 0070491348
3.	Environmental Science , 15th Edition, G. Tyler Miller, Scott Spoolman, 2012, Publisher: Brooks Cole, ISBN-13: 978-1305090446 ISBN:10: 130509044
4.	Environment Management, Vijay Kulkarni and T. V. Ramachandra, 2009, TERI Press, ISBN: 8179931846, 9788179931844

Continuous Internal Evaluation (CIE); Theory (50 Marks)

CIE is executed by way of quizzes (Q), tests (T) and Assignment. A minimum of three quizzes are conducted and each quiz is evaluated for 05 marks adding up to 15 marks. All quizzes are conducted online. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three also. The three tests are conducted for 25 marks each and the sum of the marks scored from three tests is reduced to 30. The marks component for Assignment is 05. The total marks of CIE are 50.

Semester End Evaluation (SEE); Theory (50 Marks)

SEE for 50 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 10 marks covering the complete syllabus. Part – B consists of five main questions, one from each unit for 8 marks adding up to 40 marks. Each main question may have sub questions. The question from Units I, IV and V have no internal choice. Units II and III have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom‟s taxonomy level.

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

High-3 Medium-2 Low-1

OPERATING SYSTEMS		
Course Code: 16IS43		**CIE Marks:** 100
Credits: L:T:P:S: 3:1:0:0		**SEE Marks:** 100
Hours: 36L+24T		**SEE Duration:** 03Hrs
Course Learning Objectives: The students will be able to		
1	Study the basic concepts of operating systems and understand the structure and functions of OS.	
2	Learn about Processes, Threads, Scheduling algorithms and the principles of concurrency and Deadlocks.	
3	Learn various memory management schemes and study I/O management and File systems.	
4	Learn the basics of Linux system and perform administrative tasks on Linux Servers.	

UNIT-I	
Operating Systems Overview: What Operating Systems Do ,Computer-System Organization , Computer- System Architecture ,Operating System Structure, Operating System Operations, Process Management ,Memory Management ,Storage Management , Protection and Security, Distributed Systems , Special Purpose Systems, Computing Environments **System Structures -** Operating System Services, User Operating System Interface, System Calls , Types of System Calls ,System Programs, Operating System Design and Implementation, Operating-System Structure, Virtual Machines, Operating System Debugging , Operating System Generation, System Boot	**09 Hrs**
UNIT-II	
Process Concept, Process Scheduling, Operations on Processes, Interprocess Communication, Examples of IPC Systems, Communication in Client Server Systems. **Process Scheduling:** Basic Concept, Scheduling Criteria -Scheduling Algorithms -Thread Scheduling –Multiple -Processor Scheduling. **Synchronization:** Background , The Critical-Section Problem ,Semaphores ,Classic Problems of Synchronization **Deadlocks:** System Model ,Deadlock Characterization ,Methods for Handling Deadlocks, Deadlock Prevention , Deadlock Avoidance	**09 Hrs**
UNIT-III	
Memory Management Strategies: Background, Swapping, Contiguous Memory Allocation, Paging, Structure of the Page Table. **Virtual-Memory Management:** Background , Demand Paging ,Copy-on-Write , Page Replacement , Allocation of Frames , Thrashing , Memory-Mapped Files, Allocating Kernel Memory **System Protection:** Goals of Protection, Principles of Protection, Domain of Protection , Access Matrix , Implementation of Access Matrix ,Access Control	**09 Hrs**
UNIT-IV	
File System: File Concept ,Access Methods , Directory and Disk Structure, File-System Mounting , File Sharing **Implementing File Systems:** File, System Structure, File-System Implementation, Directory Implementation, Allocation Methods	**09 Hrs**
UNIT-V	
Case Study: History of Unix And Linux, Overview Of Linux, Processes In Linux, Memory Management In Linux, Input/output In Linux, The Linux File System, Security In Linux, The Android Operating System	**08 Hrs**

Course Outcomes: After completing the course, the students will be able to	
CO1:	Analyse the basic working , functions , and design issues related to different operating systems
CO2:	Conceptualize and evaluate the different techniques and algorithms used in management of processes, memory and files.
CO3:	Demonstrate and analyse the internal working and implementation of the different modules and its interaction in the OS.
CO4:	Perform administrative and security tasks on various OS.

Reference Books	
1.	Operating System Concepts, Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, 9th Edition, 2012. John Wiley and Sons Inc., ISBN: 978-1-118-06333-0
2.	Modern Operating Systems, Andrew S. Tanenbaum,4th Edition, 2015, Addison Wesley, ISBN:13: 978-0133591620
3.	Operating Systems – Internals and Design Principles, William Stallings, 8th Edition, 2015, Prentice Hall, ISBN: 10: 0133805913
4.	Operating Systems: A Concept-Based Approach, D M Dhamdhere, 2nd Edition, 2007, Tata McGraw-Hill Education, ISBN: 9780070611948,

Continuous Internal Evaluation (CIE); Theory (100 Marks)

CIE is executed by way of quizzes (Q), tests (T) and Assignment. A minimum of three quizzes are conducted and each quiz is evaluated for 10 marks adding up to 30 marks. All quizzes are conducted online. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three also. The three tests are conducted for 50 marks each and the sum of the marks scored from three tests is reduced to 60. The marks component for assignment is 10. The total marks of CIE are 100.

Semester End Evaluation (SEE); Theory (100 Marks)

SEE for 100 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the complete syllabus. Part – B consists of five main questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have sub questions. The question from Units I, IV and V have no internal choice. Units II and III have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom‟s taxonomy level.

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

Low-1 Medium-2 High-3

DESIGN AND ANALYSIS OF ALGORITHMS (Theory & Practice)		
Course Code: 16IS44		**CIE Marks:** 100+50
Credits: L:T:P:S: 3:0:1:1		**SEE Marks:** 100+50
Hours: 36L+24T		**SEE Duration:** 03Hrs+03Hrs

Course Learning Objectives: The students will be able to	
1	Learn a mathematical model to find complexity of algorithms
2	To learn different algorithm design techniques and the algorithms that employs these techniques
3	Analyze the efficiency of algorithms using time and space complexity theory
4	Understand different algorithmic design strategies

UNIT-I	
Fundamentals of Algorithm Analysis : Definition of algorithm, Algorithmic Problem Solving, Framework for Analysis of algorithm efficiency **Asymptotic Notations :** Basic Efficiency Classes Mathematical Analysis of Non recursive algorithms, Mathematical Analysis of Recursive Algorithms, Empirical analysis of algorithms. **Divide and Conquer:** Introduction to Divide and Conquer, Master Theorem, Merge sort	**07 Hrs**
UNIT-II	
Divide and Conquer: Quick sort, Multiplication of Large Integers **Decrease and conquer :** Depth First Search(DFS) and Breadth First Search(BFS) with applications, Topological Sorting, Fake coin Problem, Computing a median and selection problem **Transform and Conquer :** Introduction, Presorting, Balanced Search Trees, Heap sort	**08 Hrs**
UNIT-III	
Dynamic Programming(DP): Floyd‟s Algorithm, The Knapsack Problem – Brute force method, bottom-up DP method and Memory Functions. **Greedy Technique:** Introduction, Prim‟s Algorithm, Kruskal‟s Algorithm, Dijkstra‟s Algorithm, Huffman Trees, Iterative improvement: Introduction, The maximum-flow problem, maximum matching in bipartite graphs, Lower-Bound Arguments, Decision Trees	**08 Hrs**
UNIT-IV	
Space and Time tradeoff – Naïve method of string matching, Boyer-Moore and Horspool‟s string matching algorithms, **Limitations of algorithm power** – Lower bound arguments, Decision Trees, P, NP and NP-Complete Problems,	**07Hrs**
UNIT-V	
Coping with limitations of algorithm Power - Backtracking(BT) :N-queens and Subset-Sum problems, **Branch-and-Bound:** Assignment problem, Knapsack problem and travelling salesman problem	**08 Hrs**

LABORATORY EXPERIMENTS	
1. Write a program to sort a given set of elements using Merge sort method and find the time required to sort the elements. 2. Write a program to sort a given set of elements using Quick sort method and find the time required to sort the elements. 3. Write a program to print all the nodes reachable from a given starting node in a graph using Depth First Search method. Also check connectivity of the graph. If the graph is not connected, display the number of components in the graph.	

4. Write a program to obtain the Topological ordering of vertices in a given digraph using a. Vertices deletion method b. DFS method 5. Write a program to print all the nodes reachable from a given starting node in a graph using Breadth First Search method. Also check connectivity of the graph. If the graph is not connected, display the number of components in the graph. 6. Write a program to sort a given set of elements using Heap sort method. Find the time complexity. 7. a. Write a program to implement Horspool algorithm for String Matching b. Write a program to implement all pair shortest paths problem using 8. Write a program to implement 0/1 Knapsack problem using dynamic programming. 9. Write a program to find Minimum cost spanning tree of a given undirected graph using Prim‟s algorithm. 10. Write a program to find Minimum cost spanning tree of a given undirected graph using Kruskal‟s algorithm. 11. Write a program to find the shortest path using Dijkstra‟s algorithm for a weighted connected graph. 12. Write a program to implement Subset-Sum problem using Back Tracking 13. Write a program to implement Assignment Problem using branch and bound algorithm 14. Write a program to implement n-queens problem	

Course Outcomes: After completing the course, the students will be able to	
CO1:	Analyze the running time of the basic algorithms for those classic problems in various domains
CO2:	Apply and implement learned algorithm design techniques and data structures to solve problems
CO3:	Develop algorithms for various computing problems
CO4:	Identify the limitations of algorithms in problem solving

Reference Books	
1.	Introduction to The Design and Analysis of Algorithms,LevitinA.,2003, Pearson Education, ISBN:9780201743951
2.	Introduction to Algorithms, CormenT.H.,Leiserson C.E.,Rivest R.L.,Stein C., 3rd Edition, 2010, PHI, ISBN:9780262033848
3.	Computer Algorithms, Horowitz E., Sahani S., Rajasekharan S., 2001,Galgotia Publications, ISBN:9780716783169
4.	Data structures and algorithm analysis in C++,Mark AllenWeiss,2003,Pearson Education ,ISBN:032144146

Continuous Internal Evaluation (CIE): Total marks: 100+50=150

Theory – 100 Marks

CIE is executed by way of quizzes (Q), tests (T) and Self-Study(S). A minimum of three quizzes are conducted and each quiz is evaluated for 10 marks adding up to 30 marks. All quizzes are conducted online. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three also. The three tests are conducted for 50 marks each and the sum of the marks scored from three tests is reduced to 50. The marks component for Self-study is 20. The total CIE for theory is 100.

Laboratory- 50 Marks

The Laboratory session is held every week as per the time table and the performance of the student is evaluated in every session. The average of marks over number of weeks is considered for 40 marks. At the end of the semester a test is conducted for 10 marks. Total marks for the laboratory is 50.

Semester End Evaluation (SEE): Total marks: 100+50=150

Theory – 100 Marks

SEE for 100 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the complete syllabus. Part – B consists of five main questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have sub questions. The question from Units I, IV and V have no internal choice. Units II and III have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom‟s taxonomy level.

Laboratory- 50 Marks

Experiment Conduction with proper results is evaluated for 40 marks and Viva is for 10 marks. Total SEE for laboratory is 50 marks.

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

Low-1 Medium-2 High-3

MICRO CONTROLLERS AND EMBEDDED SYSTEMS (Theory & Practice)		
Course Code: 16IS45		**CIE Marks:** 100+50
Credits: L:T:P:S: 3:0:1:1		**SEE Marks:** 100+50
Hours: 36L+24T		**SEE Duration:** 03Hrs+03Hrs

Course Learning Objectives: The students will be able to	
1	Provide the student with the basic understanding of microcontroller and embedded systems design.
2	Learn the addressing modes, instructions, and assembler directives and develop the flow chart, algorithms to solve problems.
3	Use of subroutines, multi-segments, macros, interrupts, procedures, stacks programs in applications
4	Develop embedded C programs for microcontrollers and run on the simulator, target board and various interfaced hardware devices.

UNIT-I	
Introduction to Microcontrollers & Architecture Intel 8051 Microcontroller: Introduction, Microprocessor Versus Microcontroller, 8051 Block Diagram, Registers, Flags &PSW, Memory Organization: Program & Data Memory, Stack structure, Addressing Modes, Data transfer Instruction‟s, Structure of ALP , Working with Keil Software Tools to develop, simulate& debug ALP & embedded C programs, Assembler Directives. **Case Study: Study the architecture of 8051 variant - NXPs 89V51RD2**	**08 Hrs**
UNIT-II	
Intel 8051 Instruction Set & Assembly Language Programming: Study of Instruction set: Arithmetic, Logic, Jump, Loop & Call Instructions, Assembly Language Programming, Procedures, Working & Programming of Timers/Counters**,** Interrupts& ISR Programs, Writing Delay programs using Instructions &Timers. **Case Study: Comparison of Applications built using: Programmed I/O & Interrupt I/O**	**08 Hrs**
UNIT-III	
Intel 8051 Interfacing & Applications: Signal/Pin Descriptions, I/O Ports, Interfacing & Programming(using ALP/Embedded C) with LEDs, Switches, Seven segment displays, LCD, Matrix Keypad, Parallel ADC (ADC0804) , DAC (DAC0800), Stepper motor, DC Motor, Programming serial port of 8051, Communication of 8051 with the PC using serial port. **Case Study: Building PC based Embedded System Using 8051 kit & RS-232**	**08 Hrs**
UNIT-IV	
Introduction to Embedded Systems & ARM Processor/Controller: Definition, Desirable Features & General Characteristics of embedded systems, Embedded Systems Vs General Computing Systems, Model of an Embedded System, Classification of Embedded Systems. History of the ARM Processor, ARM Architecture, Interrupt vector table, brief overview of ARM Instruction Set & Simple ALP Programs, Current Trends. **Case Study: Example of embedded system– RFID**	**08 Hrs**
UNIT-V	
ARM7 MCU LPC2148 – Architecture & Peripheral Programming using embedded C : History of the ARM Processor, ARM Architecture, Interrupt vector table, The internal architecture of LPC 2148 (a typical and popular ARM7 MCU) – Features of the LPC 214X Family, Peripherals and Programming : GPIO, Timers, PWM , UART, SSP units, **Case Study: Building Data Acquisition System using MCB 2140 compatible board**.	**08 Hrs**

LABORATORY EXPERIMENTS	
Set 1: a) 8051 ALP programs to perform block data transfer and searching operations b) 8051 ALP/Embedded C to Interface Logical Controller and perform: **a** Write an ALP to read the status of 8 inputs bits from 8bit switch and display „FF‟ if it is even parity otherwise display 00. Also display number of 1‟s in the input data on the LED outputs, using interface module. **b** Write an ALP to read the status of two 8-bit inputs (X and Y) and display the result X*Y using the interface module **c** Write an ALP to implement BCD Up/Down counters Set 2: **a)** 8051 ALP programs to perform Arithmetic (addn/subn/mult/divn operations) **b)** 8051 ALP/Embedded C to Interface Seven Segment Display and perform: **a** Write a C program to display messages “FIRE” & “HELP” on 4 digit seven segment display alternately with a suitable delay. **b** Write a C program to display the given number on the seven segment display using look up table **Set 3:** a) 8051 ALP programs to perform number conversions, binary to BCD, binary to ASCII b) 8051 ALP/Embedded C to Interface Stepper Motor Module and perform: **a.** Write an Embedded C program to rotate stepper motor in clockwise direction for “M” steps, anti-clock wisedirection for “N” steps **b.** Rotate the Stepper Motor, for the given RPM **Set 4:** a) 8051 ALP programs to compute average & maximum/minimum values b) 8051 ALP/Embedded C to Interface DAC Module and perform: a Write an Embedded C program to generate without rectification / full rectified/ half rectified sine waveform using DAC module b Write the program to generate square waveform for the given frequency c Generate PWM wave on pin P0.1 to control speed of DC motor. Control the duty cycle by analog input. **Set 5:** a) 8051 ALP programs to perform sorting operations b) 8051 ALP/Embedded C to Interface Keyboard Module and perform: a Write an Embedded C program to interface 4 X 4 matrix keyboard using lookup table and display the key pressed on the Terminal b Interface an LCD Module and display the temperature read from ADC Module. **Set 6:** a) To write and simulate ARM assembly language programs for data transfer, arithmetic and logical operations b) Interface Graphics LCD and I2C device to ARM Microcontroller LPC 2148 / 1768 and write the suitable embedded C program **Mini Projects :** 1. Design & Development of PC based Embedded system using 8051 Kit, incorporating Application Development on both PC & Microcontroller 2. Design & Development of LPC 2148/1768 based Data Acquisition System	

Course Outcomes: After completing the course, the students will be able to	
CO1:	Acquire the knowledge of architecture of Microprocessors and Microcontrollers.
CO2:	Develop skill in simple program writing for micro controllers assembly level language and Embedded C.
CO3:	Apply acquired knowledge to design for interface and programming.
CO4:	Analyze the design and implement for applications.

Reference Books	
1.	The 8051 Microcontroller &Embedded Systems (Using Assembly & C), Muhammad Ali Mazidi, Janice Gillispie Mazidi , Rolin D. McKinlay, 2014, New Edition, Prentice Hall (Pearson) ISBN:13-978-1-292-02657-2.
2.	The 8051 Microcontroller Architecture, Programming &Applications Kenneth J. Ayala, 2nd Edition, 2007, Thomson Learning, ISBN: 10-8131502007 13-9788131502006
3.	Embedded Systems – An integrated approach, Lyla B. Das, First Impression 2013, Pearson Education, ISBN: 978-81-317-8766-3
4.	ARM system developers guide, Andrew N Sloss, Dominic Symes and Chris Wright, 2008, Elsevier, Morgan Kaufman publishers, ISBN: 9781558608740 1558608745

Continuous Internal Evaluation (CIE): Total marks: 100+50=150

Theory – 100 Marks

CIE is executed by way of quizzes (Q), tests (T) and Self-Study(S). A minimum of three quizzes are conducted and each quiz is evaluated for 10 marks adding up to 30 marks. All quizzes are conducted online. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three also. The three tests are conducted for 50 marks each and the sum of the marks scored from three tests is reduced to 50. The marks component for Self-study is 20. The total CIE for theory is 100.

Laboratory- 50 Marks

The Laboratory session is held every week as per the time table and the performance of the student is evaluated in every session. The average of marks over number of weeks is considered for 40 marks. At the end of the semester a test is conducted for 10 marks. Total marks for the laboratory is 50.

Semester End Evaluation (SEE): Total marks: 100+50=150

Theory – 100 Marks

SEE for 100 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the complete syllabus. Part – B consists of five main questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have sub questions. The question from Units I, IV and V have no internal choice. Units II and III have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom‟s taxonomy level.

Laboratory- 50 Marks

Experiment Conduction with proper results is evaluated for 40 marks and Viva is for 10 marks. Total SEE for laboratory is 50 marks.

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

Low-1 Medium-2 High-3

UNIX SYSTEM PROGRAMMING (Theory & Practice)		
Course Code: 16IS46		**CIE Marks:** 100+50
Credits: L:T:P:S: 3:0:1:1		**SEE Marks:** 100+50
Hours: 36L+24T		**SEE Duration:** 03Hrs+03Hrs

Course Learning Objectives: The students will be able to	
1	Students study the architecture of UNIX Operating System and its use.
2	Apply the different file handling APIs for different applications.
3	Students study the concepts of IPC, Process management, Semaphores and implement the same.
4	Analyze Client Server communication and its applications.

UNIT-I	
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 Files :** File Types, The UNIX and POSIX File System, The UNIX and POSIX File Attributes, Inodes in UNIX System V, Application Program Interface to Files, UNIX Kernel Support for Files, Relationship of C Stream Pointers and File Descriptors, Directory Files, Hard and Symbolic Links.	**07 Hrs**
UNIT-II	
UNIX File APIs : General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs, Symbolic Link File APIs, File Listing Program. **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.	**07 Hrs**
UNIT-III	
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, User Identification, Process Times, I/O Redirection. **Process Relationships :** Introduction, Terminal Logins, Network Logins, Process Groups, Sessions, Controlling Terminal, tcgetpgrp and tcsetpgrp Functions, Job Control, Shell Execution of Programs, Orphaned Process Groups. **Signals :** The UNIX Kernel Support for Signals, signal, Signal Mask, sigaction, The SIGCHLD Signal and the waitpid Function, The sigsetjmp and siglongjmp Functions, Kill, Alarm, Interval Timers, POSIX.lb Timers	**07 Hrs**
UNIT-IV	
Daemon Processes: Introduction, Daemon Characteristics, Coding Rules, Error Logging, Client-Server Model. **Inter-process Communication:** Overview of IPC Methods, Pipes, popen, pclose Functions, Coprocessors, FIFOs, System V IPC, Message Queues, Semaphores, Shared Memory, Client-Server Properties, Passing File Descriptors, An Open Server-Version 1, Client-Server Connection Functions.	**07 Hrs**
UNIT-V	
Remote Procedure Calls: History of RPC, RPC Programming Interface Levels, Rpc Library Functions, rpcgen – clnt_create, the rpcgen Program, a directory Listing Example Using rpcgen, rpcgen limitations, Lower Level RPC Programming Interfaces – XDR Conversion Functions, Lower Level RPC API‟s. **Multithreaded Programming:** Thread structure and Uses, Threads and Lightweight Processes, POSIX.1c Thread APIs – thread_create, pthread_exit, pthread_detach, pthred_join, pthread_sigmask, pthread_kill, sched_yield. **Thread SunchronizationObjects:** Mutually Exclusive Locks (mutex Locks), POSIX.1c Mutex Locks, Mutex Lock Examples, POSIX.1c Conditional Variables.	**07 Hrs**

LABORATORY EXPERIMENTS	
1. The students are informed to get acquainted to the UNIX operating system environment by working on some shell commands like: who, Absolute Pathnames, Relative Pathnames 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. 2. Write a C program that accepts valid file names as command line arguments and for each of the arguments, prints the type of the file (Regular file, directory file, Character special file, Block special file, symbolic link etc.,) 3. a. Write a C program to do the following: using fork () create a child process. The child process prints its own process-id and id of its parent, The child process should read commands from the standard input and execute them (a minimal implementation of a shell – like program).and then exits. The parent process waits for its child to finish (by executing the wait ()) and prints its own process-id and the id of its child process and then exists. b. Implement a program to execute a program periodically by a daemon. 4. a. Write a program to examine and modify process priorities with getpriority(), setpriority(). Demonstrate the impact of change in priority on the output. b. Write a program to demonstrate two way Inter Process Communications between two processes using FIFO 5. a. Implement message transfer using Message Queue form of IPC. The client process request for a message that is sent by the Server process. Server returns a suitable message if the message is not available. b. Write a program which illustrates the signal handling for SIGFPE signal. 6. a. Examining and modifying process priorities and scheduling policy with sched gettscheduler(), sched_setscheduler(). b. Write a program to demonstrate two way Inter Process Communications between two processes using pipes. Use SELECT system call to monitor the read-ends of the pipes. 7. Write a program to demonstrate Use semaphores to avoid race conditions. (Hint: Create an integer variable in Shared Memory that is incremented by multiple processes causing race condition). 8. Write a C program to solve Dining Philosophers Problem. 9. Write a C program to solve Producer – Consumer problem using semaphores 10. Write C programs to implement the CPU scheduling algorithms for FCFS Scheduling and SJF Scheduling. Compare their performance metrics in terms of average turnaround time, average waiting time and average weighted turnaround. 11. Write C programs to implement the CPU scheduling algorithms for priority Scheduling and Round robin Scheduling. Compare their performance metrics in terms of average turnaround time, average waiting time and average weighted turnaround. 12. Write a C program to implement the Banker‟s algorithm for deadlock-avoidance and verify whether a system is in a safe state. 13. Write C programs to implement the FIFO and Optimal Page replacement algorithms and compare them based on the number of page faults for sample reference strings. 14. Write C programs to implement the FIFO and SCAN disk scheduling algorithms. Compare the disk scheduling algorithms based on the total head movement for given cylinder numbers 15. Design a RPC application to add and subtract a given pair of integers.	

Course Outcomes: After completing the course, the students will be able to	
CO1:	Comprehend UNIX operating system architecture, features and relevant APIs.
CO2:	Demonstrate and implement the concepts of processes and their attributes.
CO3:	Design and implement UNIX system related features like inter process communication using semaphores, shared memory& signals.
CO4:	Design and implement client server communications using various techniques.

Reference Books	
1.	Advanced Programming in the UNIX Environment, W.Richard Stevens, 3rd Edition, 2013, Addison-Wesley, ISBN:13: 978-0321637734
2.	UNIX System Programming Using C++, Terrence Chan,2010, Prentice Hall India, ISBN:13: 9788120314689
3.	UNIX Concepts and Applications, Sumitabha Das, 4th Edition, 2011, Tata McGraw Hill, ISBN: 10: 0070635463
4.	The Design of the UNIX Operating System, Maurice.J.Bach, 3rdEdition, Prentice Hall of India, ISBN-13: 978-0132017992

Continuous Internal Evaluation (CIE): Total marks: 100+50=150

Theory – 100 Marks

CIE is executed by way of quizzes (Q), tests (T) and Self-Study(S). A minimum of three quizzes are conducted and each quiz is evaluated for 10 marks adding up to 30 marks. All quizzes are conducted online. Faculty may adopt innovative methods for conducting quizzes effectively. The number of quizzes may be more than three also. The three tests are conducted for 50 marks each and the sum of the marks scored from three tests is reduced to 50. The marks component for Self-study is 20. The total CIE for theory is 100.

Laboratory- 50 Marks

The Laboratory session is held every week as per the time table and the performance of the student is evaluated in every session. The average of marks over number of weeks is considered for 40 marks. At the end of the semester a test is conducted for 10 marks. Total marks for the laboratory is 50.

Semester End Evaluation (SEE): Total marks: 100+50=150

Theory – 100 Marks

SEE for 100 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the complete syllabus. Part – B consists of five main questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have sub questions. The question from Units I, IV and V have no internal choice. Units II and III have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom‟s taxonomy level.

Laboratory- 50 Marks

Experiment Conduction with proper results is evaluated for 40 marks and Viva is for 10 marks. Total SEE for laboratory is 50 marks.

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

Low-1 Medium-2 High-3

Professional Practice – II **COMMUNICATION SKILLS AND PROFESSIONAL ETHICS**		
Course Code: 16HS47		**CIE Marks:** 50
Credits: L:T:P:S: 0:0:1:0		**SEE Marks:** NA
Hours: 18 Hrs		**CIE Duration:** 02 Hrs

Course Learning Objectives: The students will be able to	
1	Develop communication style, the essentials of good communication and confidence to communicate effectively.
2	Manage stress by applying stress management skills.
3	Ability to give contribution to the planning and coordinate Team work.
4	Ability to make problem solving decisions related to ethics.

III Semester	
UNIT-I	
Communication Skills: Basics, Method, Means, Process and Purpose, Basics of Business Communication, Written & Oral Communication, Listening. **Communication with Confidence & Clarity**: Interaction with people, the need the uses and the methods, Getting phonetically correct, using politically correct language, Debate & Extempore.	**06 Hrs**
UNIT-II	
Assertive Communication: Concept of Assertive communication, Importance and applicability of Assertive communication, Assertive Words, being assertive. **Presentation Skills**: Discussing the basic concepts of presentation skills, Articulation Skills, IQ & GK, How to make effective presentations, body language & Dress code in presentation, media of presentation.	**06 Hrs**
UNIT-III.A	
Team Work: Team Work and its important elements Clarifying the advantages and challenges of team work Understanding bargains in team building Defining behaviour to sync with team work Stages of Team Building Features of successful teams.	**06 Hrs**
IV Semester	
UNIT-III.B	
Body Language & Proxemics: Rapport Building - Gestures, postures, facial expression and body movements in different situations, Importance of Proxemics, Right personal space to maintain with different people.	**06 Hrs**
UNIT-IV	
Motivation and Stress Management: Self-motivation, group motivation, leadership abilities, Stress clauses and stress busters to handle stress and de-stress; Understanding stress - Concept of sound body and mind, Dealing with anxiety, tension, and relaxation techniques. Individual Counselling & Guidance, Career Orientation. Balancing Personal & Professional Life-	**06 Hrs**
UNIT-V	
Professional Practice: Professional Dress Code, Time Sense, Respecting People & their Space, Relevant Behaviour at different Hierarchical Levels. Positive Attitude, Self Analysis and Self-Management. **Professional Ethics**: values to be practiced, standards and codes to be adopted as professional engineers in the society for various projects**.** Balancing Personal & Professional Life	**06 Hrs**

Course Outcomes: After completing the course, the students will be able to	
CO1:	Inculcate skills for life, such as problem solving, decision making, stress management.
CO2:	Develop leadership and interpersonal working skills and professional ethics.
CO3:	Apply verbal communication skills with appropriate body language.
CO4:	Develop their potential and become self-confident to acquire a high degree of self.

Reference Books	
1.	The 7 Habits of Highly Effective People, Stephen R Covey, 2004 Edition, Free Press, ISBN: 0743272455
2.	How to win friends and influence people, Dale Carnegie, 1st Edition, 2016, General Press, ISBN: 9789380914787
3.	Crucial Conversation: Tools for Talking When Stakes are High, Kerry Patterson, Joseph Grenny, Ron Mcmillan, 2012 Edition, McGraw-Hill Publication ISBN: 9780071772204
4.	Aptimithra: Best Aptitude Book , Ethnus, 2014 Edition, Tata McGraw Hill ISBN: 9781259058738

Scheme of Continuous Internal Examination (CIE)
Evaluation of CIE will be carried out in TWO Phases.

Phase	Activity	Weightage
I	Test 1 is conducted in III Sem for 50 marks (15 Marks Quiz and 35 Marks Descriptive answers) after completion of Unit-1, Unit-2 and Unit -3.A for 18 hours of training sessions.	50%
II	Test 2 is conducted in IV Sem for 50 marks ((15 Marks Quiz and 35 Marks Descriptive answers) after completion of Unit -3B, Unit - 4 and Unit-5 for 18 hours of training sessions.	50%
	At the end of the IV sem Marks of Test 1 and Test 2 is consolidated for 50 marks (Average of Test1 and Test 2 (T1+T2/2). The grading is provided by the Coe. The final CIE marks is scrutinized by the committee comprising of HSS- Chairman, Training Co-ordinator, respective department Staff Placement co-ordinator before submitting to CoE.	

SEE: NA

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

Low-1 Medium-2 High-3

BRIDGE COURSE MATHEMATICS I / II		
Course Code: 16DMA37/48		**CIE Marks:** 100
Credits: L:T:P:S: 2:0:0:0		**SEE Marks:** 100
Audit Course		**SEE Duration:** 03Hrs

Course Learning Objectives: The students will be able to	
1	Understand the existence of polar coordinates as possible 2 - D geometry, approximate a function of single variable in terms of infinite series.
2	Gain knowledge of multivariate functions, types of derivatives involved with these functions and their applications.
3	Recognize linear differential equations, apply analytical techniques to compute solutions.
4	Acquire concepts of vector functions, vector fields and differential calculus of vector functions in Cartesian coordinates.
5	Explore the possibility of finding approximate solutions using numerical methods in the absence of analytical solutions of various systems of equations.

Prerequisites : Hyperbolic functions, Trigonometric formulas, methods of differentiation, methods of integration, reduction formulae, vector algebra.

UNIT-I	
Differential Calculus: Taylor and Maclaurin‟s series for function of single variable. Partial derivatives – Introduction, simple problems. Total derivative, Composite functions, Jacobian‟s- simple problems.	**05 Hrs**
UNIT-II	
Multiple Integrals: Evaluation of double and triple integrals – direct problems, change of order in double integral, change of variables to polar, cylindrical and spherical coordinate systems.	**05 Hrs**
UNIT-III	
Differential Equations: Higher order linear differential equations with constant coefficients, Complementary function and Particular integral, problems. Equations with variable coefficients – Cauchy and Legendre differential equations, problems.	**06 Hrs**
UNIT-IV	
Vector Differentiation: Introduction, simple problems in terms of velocity and acceleration. Concepts of Gradient, Divergence- solenoidal vector function, Curl- irrotational vector function and Laplacian, simple problems.	**05 Hrs**
UNIT-V	
Numerical Methods: Algebraic and transcendental equations – Regula-Falsi method, Newton-Raphson method.Ordinary Differential Equations – Taylor‟s, modified Euler‟s and 4^{th} order Runge-Kutta methods. Numerical Integration – Simpson‟s $1/3^{rd}$, $3/8^{th}$ and Weddle‟s rules.	**05 Hrs**

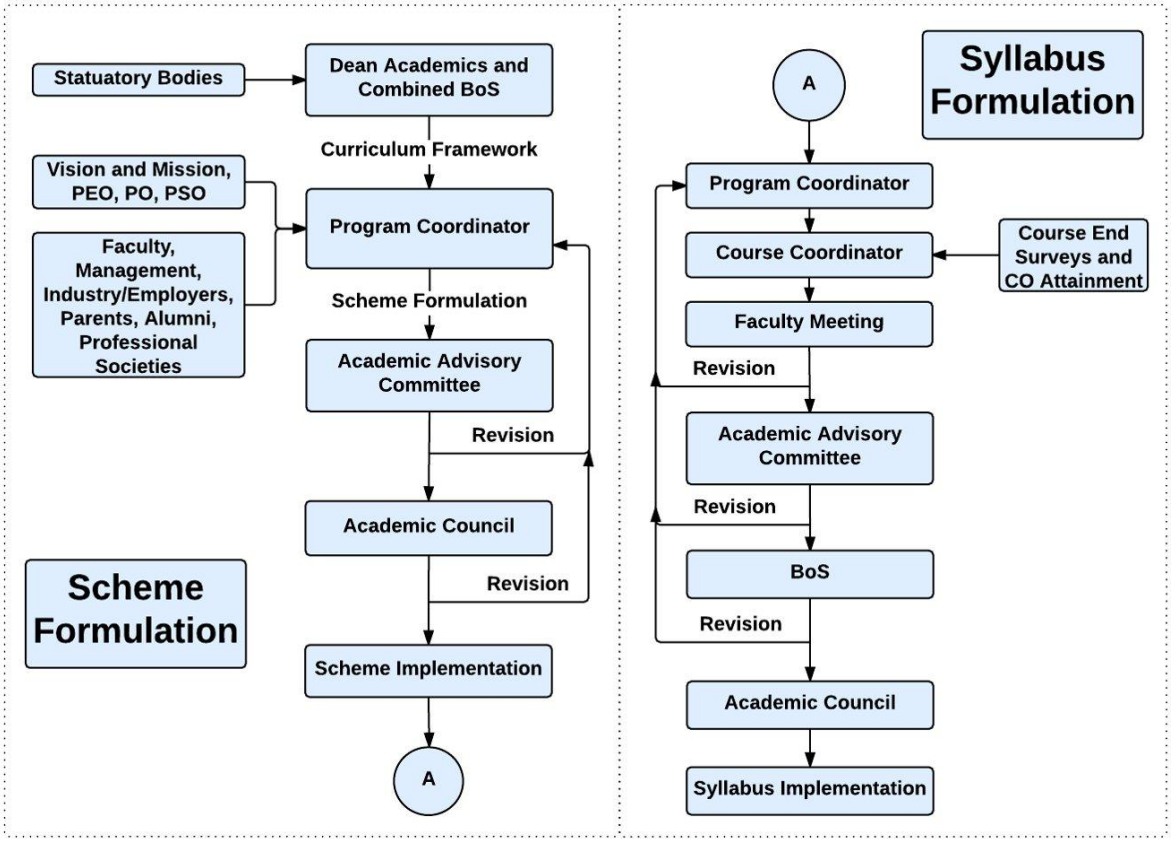
Course Outcomes: After completing the course, the students will be able to	
CO1:	Demonstrate the understanding of the basics of polar coordinates, partial differentiation, multiple integrals, vector differentiation, classification and types of solutions of higher order linear differential equations, requirement of numerical methods and few basic definitions.
CO2:	Solve problems on total derivatives of implicit functions, double integrals by changing order of integration, homogeneous linear differential equations, velocity and acceleration vectors.
CO3:	Apply acquired knowledge to find infinite series form of functions, multiple integrals by changing order, solution of non-homogeneous linear differential equations, and numerical solution of equations.
CO4:	Evaluate multiple integrals by changing variables, different operations using del operator and numerical solutions of differential equations and numerical integration.

Reference Books	
1.	Higher Engineering Mathematics, B.S. Grewal, 40th Edition, Khanna Publishers, 2007, ISBN: 81-7409-195-5.
2.	Advanced Engineering Mathematics, R. K. Jain & S.R.K. Iyengar, Narosa Publishing House, 2002, ISBN: 817-3-19-420-3. Chapters: 1, 2, 8, 15.
3.	Advanced Engineering Mathematics, Erwin Kreyszig, 9th Edition, John Wiley & Sons, 2007, ISBN: 978-81-265-3135-6. Chapters: 6, 10, 12.
4.	A Text Book of Engineering Mathematics, N.P Bali & Manish Goyal, 7th Edition, Lakshmi Publications, 2010, ISBN: 978-81-7008-992-6. Chapters: 6, 18, 16, 8, 26.

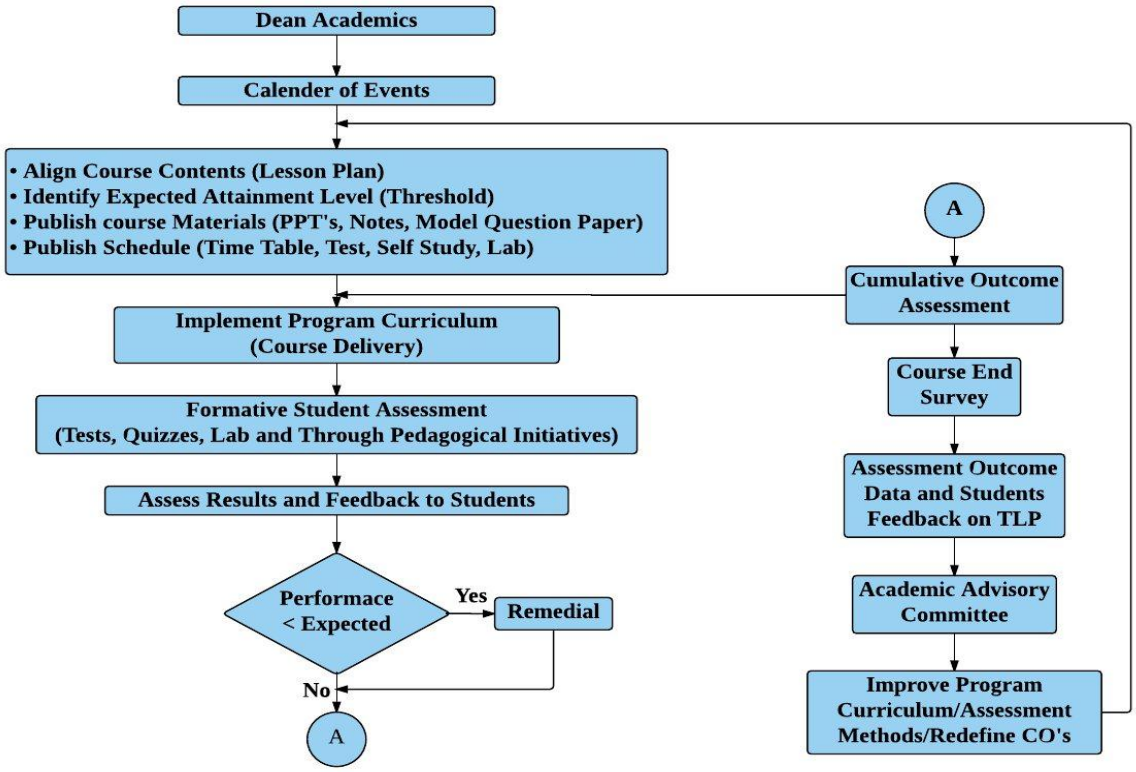
Continuous Internal Evaluation (CIE); Theory (100 Marks)
CIE consists of Two Tests each for 50 marks (20 marks for Quiz + 30 marks for descriptive questions)

Semester End Evaluation (SEE); Theory (100 Marks)
SEE for 100 marks is executed by means of an examination. The Question paper for each course contains two parts, Part – A and Part – B. Part – A consists of objective type questions for 20 marks covering the complete syllabus. Part – B consists of five main questions, one from each unit for 16 marks adding up to 80 marks. Each main question may have sub questions. The question from each unit have internal choice in which both questions cover entire unit having same complexity in terms of COs and Bloom‟s taxonomy level.

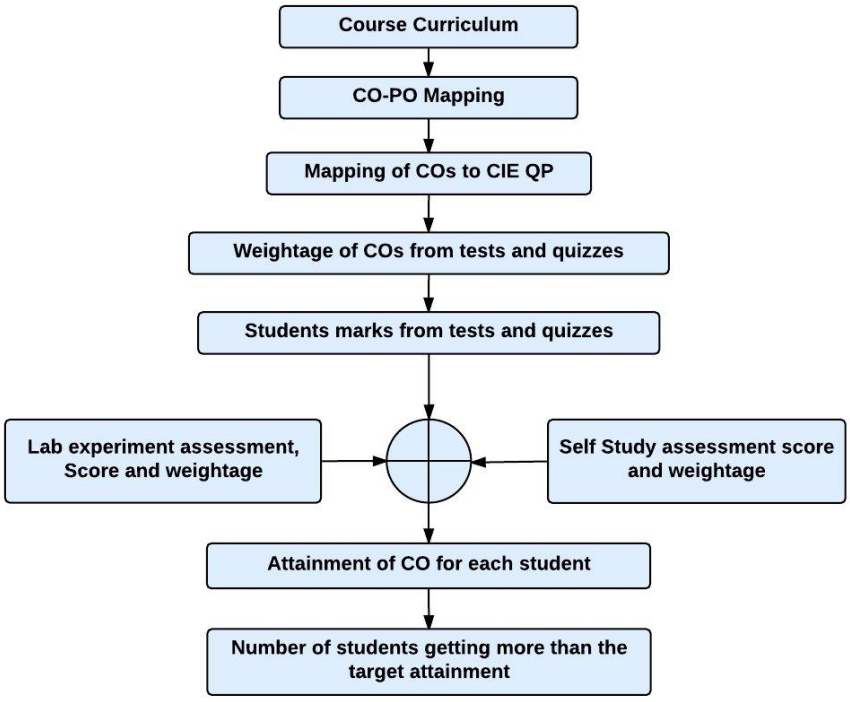
Curriculum Design Process



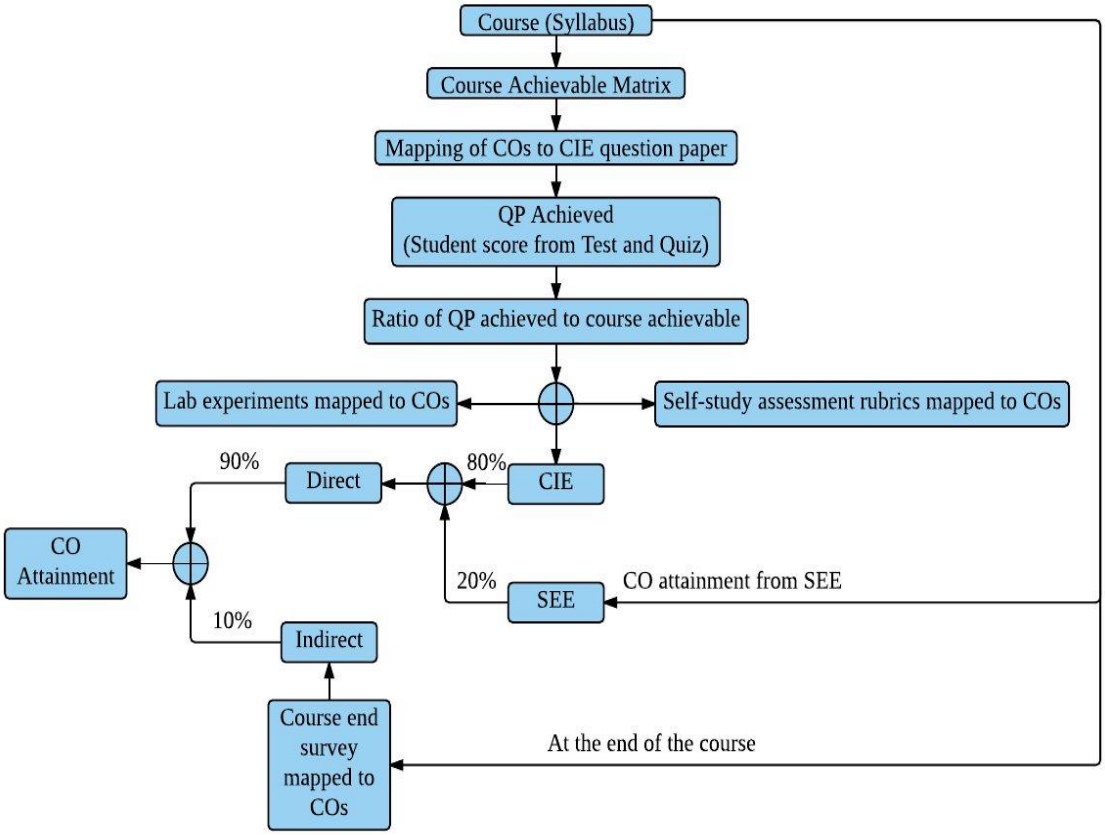
Academic Planning and Implementation



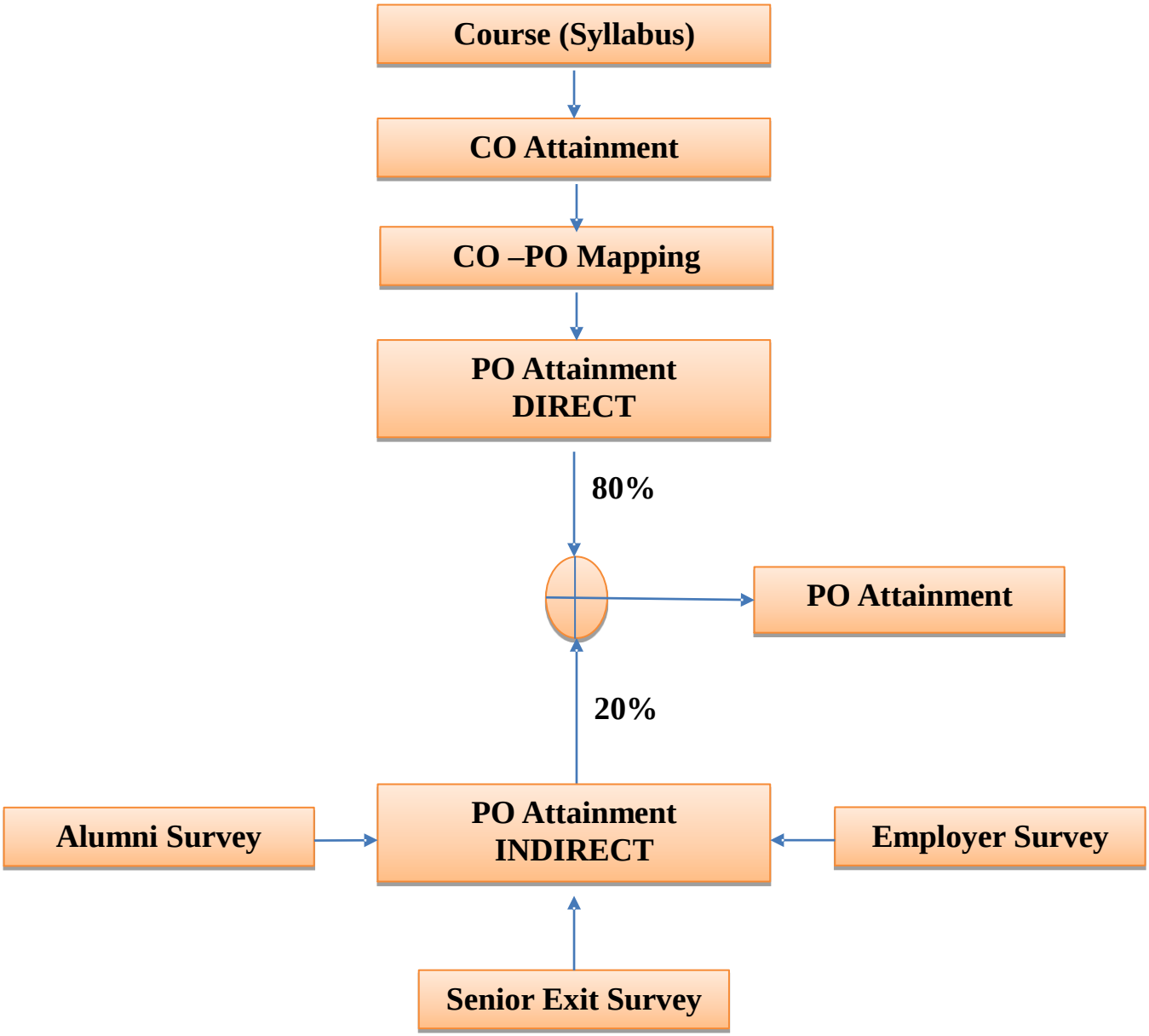
PROCESS FOR COURSE OUTCOME ATTAINMENT



Final CO Attainment Process



Program Outcome Attainment Process



Guidelines for Fixing Targets

- The target may be fixed based on last 3 years' average attainment

PROGRAM OUTCOMES (POs)

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialisation for the solution of complex engineering problems.

2. **Problem analysis:** Identify, formulate, research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

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

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

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

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

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

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

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

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

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

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