Shri Vile Parle Kelavani Mandal's

Dwarkadas J. Sanghvi College of Engineering

(Autonomous College Affiliated to the University of Mumbai)

Course Structure and Syllabus

of

Second Year B. Tech.

in

Computer Engineering

Prepared by: - Board of Studies in Computer Engineering

Recommended by: - Academic Council of D. J. Sanghvi College of Engineering

Approved by: - Governing Body of D. J. Sanghvi College of Engineering

Revision: 2 (2022)

With effect from the Academic Year: 2023-2024

Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

Scheme for Second Year B.Tech. Program in Computer Engineering : Semester III (Autonomous)
(Academic Year 2023-2024)

Semester III

Sr	Course Code	Course	Teaching Scheme				Semester End Examination (A)						Continuous Assessment (B)			Aggregate (A+B)	Credits earned	
			Theory (hrs.)	Practical (hrs.)	Tutorial (hrs.)	Credits	Duration (Hrs)	Theory	Oral	Pract	Oral & Pract	End Sem Exam Total	Theory	Term Work Total	CA Total			
1	DJS22CEC301	Engineering Mathematics-III	3	--	--	3	2	65	--	--	--	65	35	--	35	100	3	**4**
	DJS22CET301	Engineering Mathematics-III Tutorial	--	--	1	1	--	--	--	--	--	--	--	25	25	25	1	
2	DJS22CEC302	Data Structures	3	--	--	3	2	65	--	--	--	65	35	--	35	100	3	**4**
	DJS22CEL302	Data Structures Laboratory	--	2	--	1	2	--	--	--	25	25	--	25	25	50	1	
3	DJS22CEC303	Discrete Structures	4	--	--	4	2	65	--	--	--	65	35	--	35	100	4	**4**
4	DJS22CEC304	Database Management Systems	3	--	--	3	2	65	--	--	--	65	35	--	35	100	3	**4**
	DJS22CEL304	Database Management Systems Laboratory	--	2	--	1	2	--	--	--	25	25	--	25	25	50	1	
5	DJS22CEC305	Digital Electronics	3	--	--	3	2	65	--	--	--	65	35	--	35	100	3	**4**
	DJS22CEL305	Digital Electronics Laboratory	--	2	--	1	2	--	--	--	25	25	--	25	25	50	1	
6	DJS22CEL306	Programming Laboratory-I (Python)	--	4*	--	2	2	--	--	--	50	50	--	50	50	100	2	**2**
7	DJS22ILLA1	Innovative Product Development-I	--	2	--	--	--	--	--	--	--	--	--	--	--	--	--	--
8	DJS22A2	Constitution of India	1	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
		Total	**17**	**12**	**2**	**22**	**18**	**325**	**0**	**0**	**125**	**450**	**175**	**150**	**325**	**775**	**22**	**22**

*** 2 hours shown as Practical's to be taken class wise and other 2 hours to be taken as batch wise**

Prepared by: **HoD**

Name and Signatures (with date) **Department of Computer Engineering** **Principal**

Checked By

Name and Signatures (with date)

Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

Scheme for Second Year B.Tech. Program in Computer Engineering : Semester IV (Autonomous)
(Academic Year 2023-2024)

Semester IV

Sr	Course Code	Course	Teaching Scheme				Semester End Examination (A)						Continuous Assessment (B)			Aggregate (A+B)	Credits earned	
			Theory (hrs.)	Practical (hrs.)	Tutorial (hrs.)	Credits	Duration	Theory	Oral	Pract	Oral & Pract	End Sem Exam Total	Theory	Termwork: Term Work Total	CA Total			
1	DJS22CEC401	Engineering Mathematics-IV	3	--	--	3	2	65	--	--	--	65	35	--	35	100	3	**4**
	DJS22CET401	Engineering Mathematics-IV Tutorial	--	--	1	1	--	--	--	--	--	--	--	25	25	25	1	
2	DJS22CEC402	Operating Systems	3	--	--	3	2	65	--	--	--	65	35	--	35	100	3	**4**
	DJS22CEL402	Operating Systems Laboratory	--	2	--	1	2	--	--	--	25	--	--	25	25	50	1	
3	DJS22CEC403	Design and Analysis of Algorithms	3	--	--	3	2	65	--	--	--	65	35	--	35	100	3	**4**
	DJS22CEL403	Design and Analysis of Algorithms Laboratory	--	2	--	1	2	--	--	--	25	25	--	25	25	50	1	
4	DJS22CEC404	Processor Organization and Architecture	3	--	--	3	2	65	--	--	--	65	35	--	35	100	3	**4**
	DJS22CEL404	Processor Organization and Architecture Laboratory	--	2	--	1	2	--	--	--	25	25	--	25	25	50	1	
5	DJS22CEL405	Programming Laboratory-II (Web Design)	--	4	--	2	2	--	--	--	50	50	--	50	50	100	2	**2**
6	DJS22IHC1	Universal Human Values	2	--	--	2	2	65	--	--	--	65	35	--	35	100	2	**3**
	DJS22IHT1	Universal Human Values Tutorial	--	--	1	1	--	--	--	--	--	--	--	25	25	25	1	
7	DJS22ILLA2	Innovative Product Development-II	--	2	--	--	--	--	--	--	--	--	--	--	--	--	--	--
8	DJS22A3	Environmental Studies	1	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--
			15	**12**	**2**	**21**	**18**	**325**	**0**	**0**	**125**	**425**	**175**	**175**	**350**	**800**	**21**	**21**

Prepared by:

Name and Signatures (with date)

Checked By **HoD**

Name and Signatures (with date) **Department of Computer Engineering** **Principal**

Continuous Assessment (A):

Course	Assessment Tools	Marks	Time (hrs.)
Theory	a. One Term test (based on 40 % syllabus)	20	1
	b. Second Term test (next 40 % syllabus) / presentation / assignment / course project / group discussion / any other.	15	1
	Total marks (a + b)	35	--
Audit course	Performance in the assignments / quiz / power point presentation / poster presentation / group project / any other tool.	--	As applicable
Laboratory	Performance in the laboratory and documentation.	As per the scheme 25	
Tutorial	Performance in each tutorial & / assignment.		

The final certification and acceptance of term work will be subject to satisfactory performance upon fulfilling minimum passing criteria in the term work / completion of audit course.

Semester End Assessment (B):

Course	Assessment Tools	Marks	Time (hrs.)
Theory	Written paper based on the entire syllabus.	65	2
Oral	Questions based on the entire syllabus.	25	As applicable
Practical	Performance of the practical assigned during the examination and the output / results obtained.	25	2
Oral & Practical	Project based courses - Performance of the practical assigned during the examination and the output / results obtained. Based on the practical performed during the examination and on the entire syllabus.	As per the scheme	2

Prepared by Checked by Head of the Department Principal

Program: Computer Engineering	**S.Y B. Tech.**	**Semester: III**
Course: Engineering Mathematics - III (DJS22CEC301)		
Course: Engineering Mathematics - III Tutorial (DJS22CET301)		

Pre-requisite: Knowledge of

1. Engineering Mathematics – I
2. Engineering Mathematics – II
3. Calculus

Objectives:

The objective of this course is to introduce students with basic Integral Transform techniques. Application of these transforms techniques is used in solving differential equations. It will familiarize the students with some higher-level concepts that will prepare them for future research and development projects.

Outcomes: On completion of the course, learner will be able to:

1. Use Laplace and inverse Laplace Transform to the Ordinary Differential Equations.
2. Expand the periodic function by using Fourier series and complex form of Fourier series.
3. Apply Fourier Transform in the future subjects like signal processing.
4. Apply the concept of Z- transformation and its inverse of the given sequence.

Engineering Mathematics - III (DJS22CEC301)		
Unit	**Description**	**Duration**
1	**Laplace Transform:** 1.1 Introduction, Definition of Laplace transform, Laplace transform of some standard functions: $1, e^{at}, \sin at, \cos at, \sinh at, \cosh at, t^n, \operatorname{erf}\sqrt{t}$. 1.2 Heavi-side unit step function, Dirac-delta function, LT of Periodic function. 1.3 Properties of Laplace Transform: Linearity, first shifting property, second shifting property, change of scale property, multiplication by t^n, division by t. 1.4 Laplace Transform of derivatives and integrals. **Inverse Laplace Transform:** 1.5 Inverse Laplace Transform by Partial fraction method, Convolution theorem. 1.6 Application to solve initial and boundary value problem involving Ordinary differential equations and simultaneous differential equations.	**14**
2	**Fourier series:** 2.1 Dirichlet's conditions, Fourier series of periodic functions with period 2π and $2L$. 2.2 Fourier series for even and odd functions. 2.3 Half range sine and cosine series, Parsevel's identities (without proof). 2.4 Complex form of Fourier series, Orthogonal and Orthonormal set of functions. 2.5 Fourier Integral representation.	**8**

3	**Fourier Transform:** 3.1 Definition: Introduction to Fourier Transform and Inverse Fourier Transform, 3.2 Fourier Cosine Transform and Fourier Sine Transform of Functions. 3.3 Evaluation of Fourier Transform of various functions. 3.4 Properties of Fourier Transform: Linearity Property, Shifting Properties of Fourier Transform, Change of Scale and Modulation Properties of Fourier Transform, Fourier Transform of Dirac Delta Function. 3.5 Fourier Transform of Derivative and Integral of a Function 3.6 Fourier Transform of Convolution of two functions, Parseval's Identity. 3.7 Evaluation of Definite Integrals using Properties of Fourier Transform. 3.8 Finite Fourier Transform, Finite Fourier Sine Transform, Finite Fourier Cosine Transform.	**11**
4	**Z transform** 4.1 Z-transform of standard functions such as Z (a^n), Z (n^p). 4.2 Properties of Z-transform: Linearity, Change of scale, Shifting property, Multiplication of K, Initial and final value, Convolution theorem (all without proof). 4.3 Inverse Z transform: Binomial Expansion and Method of Partial fraction.	**6**
	Total	**39**

Engineering Mathematics - III Tutorial (DJS22CET301)	
Tut.	**Suggested Tutorials**
1	Laplace Transform problems based on standard forms and special function.
2	Properties of Laplace Transform.
3	Inverse Laplace Transform, Convolution theorem.
4	Application of Laplace Transform.
5	Fourier Series, Half Range Series.
6	Complex form of Fourier series, Fourier Integral representation.
7	Fourier Transform, FST, FCT.
8	Properties of Fourier Transform.
9	Evaluation of integrals by FT, Finite Fourier Transform.
10	Z- Transform and its properties.
11	Inverse Z – Transform.

Minimum eight tutorials from the above suggested list or any other experiment based on syllabus will be included, which would help the learner to apply the concept learnt.

Books Recommended:

Textbooks:

1. Advanced Differential Equations by M. D. Raisinghnia, 20th Edition, S. Chand Publications.
2. Higher Engineering Mathematics by Grewal B. S. 38th edition, Khanna Publication 2005.

Reference Books:

1. Advanced Engg. Mathematics by C. Ray Wylie & Louis Barrett.TMH International Edition.
2. Mathematical Methods of Science and Engineering by Kanti B. Datta, Cengage Learning.
3. Integral Transforms and their Engineering Applications by Dr. B. B. Singh, Synergy Knowledgewar.
4. Laplace Transforms by Murry R. Spieget, Schaun‟s out line series-McGraw Hill Publication.

Prepared by Checked by Head of the Department Principal

Program: Computer Engineering	**S.Y B. Tech.**	**Semester: III**
Course: Data Structures (DJS22CEC302)		
Course: Data Structures Laboratory (DJS22CEL302)		

Pre-requisite: Knowledge of -

1. C – Programming

Course Objectives: The objective of the course is to introduce and familiarize students with linear and non-linear data structures, their use in fundamental algorithms and design & implementation of these data structures. To introduce students to the basics of algorithms and time complexity. To familiarize students with various sorting and searching techniques, and their performance comparison.

Course Outcomes: On successful completion of this course, student should be able to:

1. Understand the concept of time complexity for algorithms.
2. Assimilate the concept of various linear and non-linear data structures.
3. Solve the problem using appropriate data structure.
4. Implement appropriate searching and sorting algorithms for a given problem.

Data Structures (DJS22CEC302)		
Unit	**Description**	**Duration**
1	**Basics of Algorithms:** Algorithms, Characteristics of an Algorithm, Time and Space Complexities, Order of Growth functions, Preliminary Asymptotic Notations. **Data Structures:** Introduction, need of Data Structures, Types of Data Structures, Abstract Data Types (ADT)	**04**
2	**Linear Data Structures – LIST:** List as an ADT, Array-based implementation, Linked List implementation, singly linked lists, circularly linked lists, doubly-linked lists, All operations (Insertion, Deletion, Merge, Traversal, etc.) and their analysis, Applications of linked lists - (Polynomial Addition).	**06**
3	**Linear Data Structure – STACK:** Stack as an ADT, Operations, Array and Linked List representation of Stack, Applications – Reversing data, Conversion of Infix to prefix and postfix expression, Evaluation of postfix and prefix expressions, balanced parenthesis, etc. **Linear Data Structure – QUEUE:** Queue as an ADT, Operations, Implementation of Linear Queue, Circular and Priority Queue using arrays and Linked List, DEQueue, Applications – Queue Simulation.	**08**

4	**Non-Linear Data Structure – TREES:** Tree Terminologies, Tree as an ADT, Binary Tree - Operations, Tree Traversals, Binary Search Tree (BST) - Operations, Expression Trees **Height Balanced Tree**: Creation of AVL Tree, **Heap**- Operations on heap **Applications** - Huffman coding	**10**
5	**Non-Linear Data Structure – GRAPHS:** Graph Terminologies, Types of Graphs, Representation of Graph using arrays and Linked List, Breadth-First Search (BFS), Depth–First Search (DFS), **Applications of Graphs** -Topological sorting.	**03**
6	**Searching**- Linear Search, Binary Search and Fibonacci search. **Sorting:** Bubble Sort, Selection Sort, Heap Sort, Insertion Sort, Radix Sort, Merge Sort, Quick Sort. Analysis of Searching and Sorting Techniques. **Hashing:** Hash Functions, Overflow handling, Collision & Collision Resolution Techniques, Linear hashing, Hashing with chaining, Separate Chaining, Open Addressing, Rehashing and Extendible hashing.	**08**
	Total	**39**

Data Structures Laboratory (DJS22CEL302)	
Tut.	**Suggested Practical (Any 10)**
1.	Implementation of Linked List using menu driven approach.
2.	Implementation of different operations on linked list –copy, concatenate, split, reverse, count no. of nodes etc.
3.	Implementation of polynomials operations (addition, subtraction) using Linked List.
4.	Implementation of stack using menu driven approach.
5.	Implementation of Infix to Postfix conversion.
6.	Implementation of prefix and postfix evaluation using menu driven approach.
7.	Implementation of parenthesis checker using stack.
8.	Implementation of Linear queue using menu driven approach.
9.	Implementation of circular queue using menu driven approach.
10.	Implementation of double ended queue using menu driven approach.
11.	Implementation of Priority queue program using array and Linked list.
12.	Implementation of Binary Tree using menu driven approach.
13.	Implementation of Binary Tree Traversal.
14.	Implementation of BST using following operations – create, delete, display.
15.	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.
16.	Implementation of Graph traversal using menu driven program (DFS & BFS).
17.	Implementations of Selection sort, Radix sort using menu driven.
18.	Implementation of Heap & Heap Sort using menu driven program.

19.	Implementation of Advanced Bubble Sort and Insertion Sort using menu driven Program.
20.	Implementation of searching methods (Index Sequential, Fibonacci search, Binary Search) using menu driven program.
21.	Implementation of hashing functions with different collision resolution techniques.

Any other experiment based on syllabus may be included, which would help the learner to understand topic/concept.

Books Recommended:

Textbooks:

1. R. F. Gilberg and B. A. Forouzan, "Data Structures – A Pseudocode Approach with C", 2nd Edition, Cengage Learning, 2005.
2. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, "Fundamentals of Data Structures in C", 2nd Edition, W. H. Freeman and Company 2008.

Reference Books:

1. Mark A. Weiss, "Data Structures and Algorithm Analysis in C", 4th Edition, Pearson, 2014.
2. M. T. Goodritch, R. Tamassia, D. Mount, "Data Structures and Algorithms in C++", Wiley, Second Edition, 2011.
3. Kruse, Leung, Tondo, "Data Structures and Program Design in C", 2nd Edition, Pearson Education, 2013.
4. Tenenbaum, Langsam, Augenstein, "Data Structures using C", Pearson, Second edition 2015.
5. J. P. Tremblay and P. G. Sorenson, "Introduction to Data Structures and its Applications", 2nd Edition, McGraw- Hill, 1984.
6. Aho, Hopcroft, Ullman, "Data Structures and Algorithms", Addison-Wesley, 2010.
7. Reema Thareja, "Data Structures using C", Oxford, 2017.
8. Seymour Lipschutz, Data Structures, Schaum's Outline Series, 1st Edition, Tata McGraw-Hill, 2014.

Prepared by Checked by Head of the Department Principal

Program: Computer Engineering	S.Y B. Tech.	Semester: III
Course: Discrete Structures (DJS22CEC303)		

Objectives:

1. To use mathematically correct terminology and notation.
2. To cultivate clear thinking and creative problem solving.
3. To thoroughly train in the construction and understanding of mathematical proofs. Exercise common mathematical arguments and proof strategies.
4. To thoroughly prepare for the mathematical aspects of other Computer Engineering courses.

Outcomes: On completion of the course, the learner will be able to:

1. Understand discrete and fuzzy set theory.
2. Verify the correctness of an argument using propositional and predicate logic and truth tables.
3. Understand Relations, Diagraph and lattice and functions.
4. Apply principles and concepts of graph theory and trees in practical situations.
5. Understand the different Algebraic structures and demonstrate use of groups and codes in Encoding and Decoding.

Discrete Structures (DJS22CEC303)		
Unit	**Description**	**Duration**
1	**Sets and Logic:** **Set Theory:** Introduction to Set Theory, Venn diagrams, Operations on Sets, Power sets, Laws of set theory, Cartesian Product, Partitions of sets, The Principle of Inclusion and Exclusion, Introduction to Fuzzy sets, Properties of Fuzzy sets, Fuzzy set operations, Fuzzy Cartesian product **Mathematical Logic:** Propositions and Logical operations, Truth tables, Laws of Logic, Logical Equivalence, Normal Forms, Predicates, Fallacies, Quantifiers, Mathematical induction Introduction to First Order Predicate Logic, Inference Rules: Universal and Existential instantiation, Universal and Existential generalization, Universal Modus Ponens, Universal Modus Tollens, Multiple Quantifiers, Negation of more than one variable	8

2	**Relations, Posets and Lattices** **Introduction:** Relations and their properties, Paths and Digraphs, Types of binary relations, Operations on relations, Equivalence relations: Closures, Warshall's algorithm, Composition of relations. Introduction to Fuzzy Relations. Properties of Fuzzy relations, Fuzzy composition of relations: Max-min composition and Max-product composition **Posets and Lattices:** Partial ordered sets, Hasse diagram, Lattice and its types, Boolean algebra.	10
3	**Functions** Types of functions - Injective, Surjective and Bijective, Composition of functions, Identity and Inverse function, Pigeon hole principle	5
4	**Graphs and Trees** **Introduction to Graph theory**: Definitions, Paths and circuits, **Types of Graphs**: Eulerian and Hamiltonian, Sub Graphs, Planar Graphs, Chromatic number, Graph coloring(Welch-powell algorithm) **Functions and Graphs**: Isomorphism of graphs Clique, Independent set, bipartite graph **Introduction to Trees:** Trees, rooted trees, path length in rooted trees, Prefix codes and optimal prefix codes (Huffman coding) **Tree Traversals**: Binary search trees, tree traversals, spanning trees, Minimal spanning trees, **Application of Trees**: The Max flow–Min cut theorem (transport network)	6
5	**Algebraic Structures** **Groups:** Binary operations, Group, Semigroup, Monoid, Sub-group, Cyclic group, Homomorphism and Isomorphism of groups, Cosets. **Rings and Fields:** Definition, Sub rings, Integral domain, Field, Integer modulo n, Ring homomorphism.	5
6	**Coding theory:** Group codes, Parity-check and Generator matrix, Hamming codes, Maximum likelihood technique	5
	Total	**39**

Books Recommended:

Text books:

1. Bernad Kolman, Robert Busby, Sharon Cutler Ross, Nadeem-ur-Rehman, "Discrete Mathematical Structures", Pearson Education 2015.
2. C.L. Liu, D P Mohapatra, "Elements of Discrete Mathematics", 4E, McGraw-Hill 2012.
3. Douglas B West.," Introduction to Graph Theory" 2nd Edition, Eastern Economy Edition published by PHI Learning Pvt. Ltd.
4. Ralph Grimaldi, "Discrete and Combinatorial Mathematics" 5th ed, Pearson Education
5. S.N. Sivanandam, S. N. Deepa, "Principles of Soft Computing", 2nd Edition, 2011 Wiley India Pvt. Ltd

Reference Books:

1. Y N Singh, "Discrete Mathematical Structures", Wiley-India.
2. J. L. Mott, A. Kandel, T. P. Baker, "Discrete Mathematics for Computer Scientists and Mathematicians", Prentice Hall of India.
3. J. P. Trembley, R. Manohar "Discrete Mathematical Structures with Applications to Computer Science", McGraw-Hill.
4. Seymour Lipschutz, Marc Lipson, "Discrete Mathematics", Schaum's Outline Series McGraw Hill Education.

Prepared by Checked by Head of the Department Principal

Program: Computer Engineering	**S.Y B. Tech.**	**Semester: III**
Course: : Database Management Systems (DJS22CEC304)		
Course: Database Management Systems Laboratory (DJS22CEL304)		

Prerequisite: Computer Basics

Objectives: The course intends to introduce the students to the management of database systems, with an emphasis on how to design, organize, maintain and retrieve information efficiently and effectively from a database.

Outcomes: On successful completion of this course, learner will be able to:

1. Design an optimized database.
2. Construct SQL queries to perform operations on the database.
3. Demonstrate appropriate transaction management and recovery techniques for a given problem.
4. Apply indexing mechanisms for efficient retrieval of information from database.

Detailed Syllabus: (unit wise)		
Unit	**Description**	**Duration**
1	**Introduction Database Concepts:** Introduction, Characteristics of databases, File system v/s Database system, Users of Database system, Schema and Instance, Data Independence, DBMS system architecture, Database Administrator	03
2	**Relational Data Model:** **Entity–Relationship Model:** The Entity-Relationship (ER) Model: Entity types: Weak and strong entity sets, Entity sets, Types of Attributes, Keys, Relationship constraints: Cardinality and Participation, Extended Entity-Relationship (EER) Model: Generalization, Specialization and Aggregation **Relational Model:** Introduction to the Relational Model, relational schema and concept of keys, Mapping the ER and EER Model to the Relational Model, Introduction to Object-Relational Databases, ORDBMS Vs Relational Databases **Relational Algebra –** Unary and Set operations, Relational Algebra Queries.	09
3	**Structured Query Language (SQL):** Overview of SQL, Data Definition Commands, Data Manipulation commands, Integrity constraints - key constraints, Domain Constraints, Referential integrity, check constraints, Data Control	09

	commands, Transaction Control Commands, Set and String operations, aggregate function - group by, having, Views in SQL, joins, Nested and complex queries, Triggers, Security and authorization in SQL	
4	**Relational–Database Design:** Pitfalls in Relational-Database designs, Concept of normalization, Function Dependencies, Normal Forms- 1NF, 2NF, 3NF, BCNF	05
5	**Transaction Management and Recovery:** Transaction Concept, ACID properties, Transaction States, Implementation of atomicity and durability, Concurrent Executions, Serializability, Concurrency Control Protocols: Lock-based, Timestamp based, Validation Based, Deadlock Handling, Recovery System: Failure classification, Log based recovery, Shadow Paging, ARIES recovery algorithm.	09
6	**Indexing Mechanism:** Hashing techniques, Types of Indexes: Single Level Ordered Indexes, Multilevel Indexes, Overview of B-Trees and B+ Trees.	04
	Total	39

List of Laboratory Experiments:

Database Management Systems Laboratory (DJS22CEL304)	
Tut.	**Suggested Practical**
1	To draw an ER diagram for a problem statement.
2	Map the ER/EER to relational schema.
3	To implement DDL and DML commands with integrity constraints.
4	To access & modify Data using basic SQL.
5	To implement Joins and Views.
6	To implement Subqueries.
7	To implement triggers.
8	Examine the consistency of database using concurrency control technique (Locks)
9	To simulate ARIES recovery algorithm.
10	To implement B-trees/B+ trees.

Any other experiment based on syllabus may be included, which would help the learner to understand topic/concept.

Books Recommended:

Text books:

1. Korth, Silberchatz, Sudarshan, ―Database System Concepts, 7th Edition, McGraw – Hill, 2019.
2. Elmasri and Navathe, ―Fundamentals of Database Systems, 7th Edition, Pearson education, 2016.
3. Peter Rob and Carlos Coronel, ―Database Systems Design, Implementation and Management, Thomson Learning, 5th Revised Edition, 2002.
4. G. K. Gupta ―Database Management Systems, 3rd Edition, McGraw – Hill, 2018.

Reference Books:

1. Dr. P.S. Deshpande, SQL and PL/SQL for Oracle 10g, Black Book, Dreamtech Press, 2012
2. Sharaman Shah, ―Oracle for Professional, Shroff Publishers & Distributers Private Limited, 1st edition, 2008
3. Raghu Ramakrishnan and Johannes Gehrke, ― Database Management Systems, 3rd Edition, McGraw – Hill, 2014.
4. Patrick Dalton, "Microsoft SQL Server Black Book", Coriolis Group,U.S., 11th ed. edition (1 July 1997)
5. Lynn Beighley, "Head First SQL", O'Reilly Media, 1st edition (28 August 2007)

Prepared by Checked by Head of the Department Principal

Program: Computer Engineering	S.Y B. Tech.	Semester: III
Course: Digital Electronics (DJS22CEC305)		
Course Digital Electronics Laboratory (DJS22CEL305)		

Pre-Requisite – BEE & DE (First Year)

Objectives:

1. To introduce the fundamental concepts and methods for design of digital circuits and a pre-requisite for computer organization and architecture, microprocessor systems.
2. To provide the concept of designing Combinational and sequential circuits.
3. To provide basic knowledge of how digital building blocks are described in VHDL.

Outcomes: On completion of the course, learner will be able to:

1. Understand different number systems and their arithmetic.
2. Analyze and minimize logical expressions.
3. Design and analyze combinational circuits and sequential circuits.
4. Design and analyze synchronous and asynchronous counters.
5. Understand programming logic devices.

Digital Electronics (DJS22CEC305)		
Unit	**Description**	**Duration**
1	**Number Systems and Codes:** Introduction to number system: Binary, Octal, Decimal and Hexadecimal Number Systems. Binary arithmetic: addition, subtraction (1's and 2's complement), multiplication and division. Octal and Hexadecimal arithmetic: Addition and Subtraction (7's and 8's complement method for octal) and (15's and 16's complement method for Hexadecimal). Codes: Gray Code, BCD Code, Excess-3 code, Error Detection and Correction: Hamming codes.	**06**
2	**Boolean Algebra and Logic Gates:** Boolean Algebra, Boolean functions, function reduction using Boolean laws, SOP and POS form of logic functions. Logic simplifications using K-map method - 2 variable, 3 variables, 4 variables, Don't care condition, Logic simplifications using Quine-McClusky Method,	**09**

	functions realization using basic gates and universal gates.	
3	**Combinational Logic Design:** Introduction, Half and Full Adder, Half subtractor, Full Subtractor, Four Bit Ripple adder, look ahead carry adder, 4 bit adder subtractor, one digit BCD Adder, Multiplexer, Multiplexer tree, Demultiplexer, Demultiplexer tree, Encoders and decoders, Priority encoder, One bit, Two bit , 4-bit Magnitude Comparator, ALU IC 74181.	**08**
4	**Sequential Logic Design:** Flip Flops: SR, D, J-K, T, Truth Tables and Excitation Tables, Race around condition, Master Slave J-K Flip Flops, Flip-flop conversions. Application of Sequential Logics. Counters: Design of Asynchronous and Synchronous Counters, Mod Counters, UP-DOWN counter, Shift Registers: SISO, SIPO, PIPO, PISO Bidirectional Shift Register, Universal Shift Register, Ring and twisted ring/Johnson Counter, sequence generator.	**12**
5	**Programming Logic Devices:** Concepts of Programmable Array Logic (PAL) and Programming Logic Array (PLA). FPGA Arthitectures. VHDL : Basics of VHDL, 3 models of architecture : dataflow, structural and behavioral.	**04**
	Total	**39**

Digital Electronics Laboratory (DJS22CEL305)	
Sr.	**Suggested Practical**
1	To study and verify the truth table of various logic gates using ICs and realize Boolean expressions using gates.
2	To realize basic gates using universal gates.
3	To realize binary to gray code and gray code to binary converter.
4	To realize parity generator and detector.
5	To realize arithmetic circuits i) Half adder ii) Full adder iii) Half subtractor iv) Full subtractor.
6	To realize 2 bit magnitude comparator.
7	To Study multiplexer IC and realization of full adder using multiplexer IC.
8	To Study decoder IC and realization of combinational logic using decoder IC.
9	Study of flip-flops using IC's.
10	To realize asynchronous 3 bit up counter.
11	To realize shift registers using flip flops.
12	To realize basic gates using VHDL.
13	To realize 4:1 multiplexer using VHDL.
14	To realize 4bit counter using VHDL.
15	Case study on practical uses of flip-flop's and Counters

Minimum ten experiments from the above suggested list or any other experiment based on syllabus will be included, which would help the learner to apply the concept learnt.

Books Recommended:

Text books:

1. R. P. Jain, “Modern Digital Electronics”, Tata McGraw Hill.
2. M. Morris Mano, “Digital Logic and computer Design”, PHI.
3. Norman Balabanian, “Digital Logic Design Principles”, Wiley.
4. J. Bhasker.“ VHDL Primer”, Pearson Education.

Reference Books:

1. Donald p Leach, Albert Paul Malvino,“Digital principles and Applications”,Tata McGraw
2. Yarbrough John M. , “Digital Logic Applications and Design “, Cengage Learning.
3. Douglas L. Perry, “VHDL Programming by Example”, Tata McGraw Hill.

Program: Computer Engineering	S.Y B. Tech.	Semester: III
Course: : Programming Laboratory-I (Python) (DJS22CEL306)		

Pre-requisite:

1. C Programming

Objectives:

1. To learn the basic and OOP concepts of Python.
2. To study various advanced python concepts like inheritance, exception handling, modules etc.
3. Learn to develop GUI based standalone and web application.

Outcomes: On completion of the course, the learner will be able to:

1. Understand basic and object-oriented concepts, data structure implementation in python.
2. Apply file, directory handling and text processing concepts in python.
3. Apply database connectivity, client-server communication using python.
4. Develop python-based application using Tkinter.

Programming Laboratory-I (Python) (DJS22CEL306)		
Unit	**Description**	**Duration**
1	**Python basics** Data types in python, Operators in python, Input and Output, Control statement, Arrays in python, String and Character in python, Functions, List and Tuples, Dictionaries.	6
2	**Control Statements and Functions:** If statement, if-elif-else, Repetition using while loop, for loop, defining a Function, Checking & Setting Your Parameters, Default arguments, Variable length arguments, Defining and calling functions within a function, Layers of Functions, Lambda and Filter, Zip (), Map (), Reduce () function, recursion, Function Decorators.	6
3	**Introduction to OOP:** Creating a Class, Self-Variables, Constructors, Types of Methods, Constructors in Inheritance, Polymorphism, the super () Method, Method Resolution Order (MRO), Operator Overloading, Method Overloading & Overriding, Interfaces in Python Exceptions Handling: Exceptions, Exception Handling, Types of Exceptions, Except Block, assert Statement, User Defined Exceptions	12

4	**Advanced Python** Building Modules, Packages: Python Collections Module, Opening and Reading Files and Folders (Python OS Module, Python Datetime Module, Python Math and Random Modules, Text Processing, Regular expression in python	8
5	**Python Integration Primer Graphical User interface using Tkinter** : Form designing, Networking in Python: Client Server socket programming Python database connectivity: Data Definition Language (DDL), and Data Manipulation Language (DML)	8
6	**Python advance Modules Numpy:** Working with Numpy, Constructing Numpy arrays, Printing arrays, Arithmetic Operations on matrix's, numpy zeros() Matplotlib: Matplotlib- Plot different charts, Pandas: Data Processing, Pandas-Data structure, Pandas-Series data, Data Frames, Introduction to data processing using pandas	12
	Total	39

Programming Laboratory-I (Python) (DJS22CEL306)	
Sr.	**Suggested Practical**
1	Write python programs to understand Expressions, Variables, Quotes, Basic Math operations.
2	Write python programs to demonstrate applications of different decision-making statements.
3	Write a Python program to implement Basic String Operations & String Methods.
4	Write a Python program to implement functions of List, Tuples, and Dictionaries.
5	Write a Python program to implement Arrays (1D, 2D) applications.
6	Write a Python program to implement Functions and Recursion.
7	Write a Python program to implement Programs based on Lambda, Map, and Reduce Functions.
8	Write a Python program to implement program to implement concept of Function decorators.
9	Write python programs to implement Classes & objects, Constructors
10	Write python programs to implement Inheritance & Polymorphism.
11	Write python programs to implement Exception handling.
12	Write python programs to understand different File handling operations with exception handling.
13	Write python programs to implement database connectivity and DDL and DML commands in Python using SQLite.
14	Write python programs to understand GUI designing (Programs based on GUI designing using Tkinter.
15	Implement different Machine learning packages like numpy, pandas and matplotlib.

Minimum 10 experiments from the above suggested list or any other experiment based on syllabus will be included, which would help the learner to apply the concept learnt.

Books Recommended:

Text books:

1. Dr. R. Nageswara Rao, "Core Python Programming", 3rd Edition, Dreamtech Press, 2018.

2. Mark Lutz, "Learning Python", 5th Edition, Oreilly Publication, 2013.

3. E Balagurusamy, "Introduction to computing and problem-solving using Python", McGraw Hill Education, 2018

Reference Books:

1. Zed A. Shaw, "Learn Python the Hard Way", 3rd Edition, Addison–Wesley Publication, 2014.

2. Laura Cassell, Alan Gauld, "Python Projects", Wrox Publication, 2015.

Prepared by Checked by Head of the Department Principal

Program: Computer Engineering	S.Y B. Tech.	Semester: III
Course: : Innovative Product Development-I (DJS22ILLA1)		

Objectives:

1. To acquaint the students with the process of identifying the need (considering a societal requirement) and ensuring that a solution is found out to address the same by designing and developing an innovative product.
2. To familiarize the students with the process of designing and developing a product, while they work as part of a team.
3. To acquaint the students with the process of applying basic engineering fundamentals, so as to attempt at the design and development of a successful value-added product.
4. To inculcate the basic concepts of entrepreneurship and the process of self-learning and research required to conceptualize and create a successful product.

Outcomes: On completion of the course, the learner will be able to:

1. Identify the requirement for a product based on societal/research needs.
2. Apply knowledge and skills required to solve a societal need by conceptualizing a product, especially while working in a team.
3. Use standard norms of engineering concepts/practices in the design and development of an innovative product.
4. Draw proper inferences through theoretical/ experimental/simulations and analyze the impact of the proposed method of design and development of the product.
5. Develop interpersonal skills, while working as a member of the team or as the leader.
6. Demonstrate capabilities of self-learning as part of the team, leading to life-long learning, which could eventually prepare themselves to be successful entrepreneurs.
7. Demonstrate product/project management principles during the design and development work and also excel in written (Technical paper preparation) as well as oral communication.

Guidelines for the proposed product design and development:

- Students shall form a team of 3 to 4 students (max allowed: 5-6 in extraordinary cases, subject to the approval of the department review committee and the Head of the department).
- Students should carry out a survey and identify the need, which shall be converted into conceptualization of a product, in consultation with the faculty supervisor/head of department/internal committee of faculty members.
- Students in the team shall understand the effective need for product development and accordingly select the best possible design in consultation with the faculty supervisor.

- Students shall convert the best design solution into a working model, using various components drawn from their domain as well as related interdisciplinary areas.
- Faculty supervisor may provide inputs to students during the entire span of the activity, spread over 2 semesters, wherein the main focus shall be on self-learning.
- A record in the form of an activity log-book is to be prepared by each team, wherein the team can record weekly progress of work. The guide/supervisor should verify the recorded notes/comments and approve the same on a weekly basis.
- The design solution is to be validated with proper justification and the report is to be compiled in a standard format and submitted to the department. Efforts are to be made by the students to try and publish a technical paper, either in the institute journal, "Techno Focus: Journal for Budding Engineers" or at a suitable publication, approved by the department research committee/ Head of the department.
- The focus should be on self-learning, capability to design and innovate new products as well as on developing the ability to address societal problems. Advancement of entrepreneurial capabilities and quality development of the students through the year long course should ensure that the design and development of a product of appropriate level and quality is carried out, spread over two semesters, i.e. during the semesters III and IV.

Guidelines for Assessment of the work:

- The review/ progress monitoring committee shall be constituted by the Head of the Department. The progress of design and development of the product is to be evaluated on a continuous basis, holding a minimum of two reviews in the semester.
- In the continuous assessment, focus shall also be on each individual student's contribution to the team activity, their understanding and involvement as well as responses to the questions being raised at all points in time.
- Distribution of marks individually for the both reviews during the semester shall be as given below:
    - Marks awarded by the supervisor based on log-book: 20
    - Marks awarded by review committee: 20
    - Quality of the write-up: 10

A candidate needs to secure a minimum of 50% marks to be declared to have completed the audit course.

Review/progress monitoring committee may consider the following points during the assessment.

- In the semester III, the entire design proposal shall be ready, including components/system selection as well as the cost analysis. Two reviews will be conducted based on the presentation given by the student's team.

- First shall be for finalization of the product selected.
- Second shall be on finalization of the proposed design of the product.

The overall work done by the team shall be assessed based on the following criteria;

1. Quality of survey/ need identification of the product.
2. Clarity of Problem definition (design and development) based on need.
3. Innovativeness in the proposed design.
4. Feasibility of the proposed design and selection of the best solution.
5. Cost effectiveness of the product.
6. Societal impact of the product.
7. Functioning of the working model as per stated requirements.
8. Effective use of standard engineering norms.
9. Contribution of each individual as a member or the team leader.
10. Clarity on the write-up and the technical paper prepared.

• The semester reviews may be based on relevant points listed above, as applicable.

Guidelines for Assessment of Semester Reviews:

- The write-up should be prepared as per the guidelines given by the department.
- The design and the development of the product shall be assessed through a presentation and demonstration of the working model by the student team to a panel of Internal Examiners,

Prepared by Checked by Head of the Department Principal

Program: Computer Engineering	S.Y B. Tech.	Semester: III
Course: : Constitution of India (DJS22A2)		

Objectives:
1. To provide basic information about Indian constitution.
2. To identify individual role and ethical responsibility towards society.
3. To understand human rights and its implications.

Outcomes: On completion of the course, the learner will be able to:
1. Have general knowledge and legal literacy and thereby take up competitive examinations.
2. Understand state and central policies, fundamental duties.
3. Understand Electoral Process, special provisions.
4. Understand powers and functions of Municipalities, Panchayats and Co-operative Societies.
5. Understand Engineering ethics and responsibilities of Engineers.
6. Understand Engineering Integrity & Reliability.

Constitution of India (DJS22A2)		
Unit	**Description**	**Duration**
1	**Introduction to the Constitution of India:** The Making of the Constitution and Salient features of the Constitution. Preamble to the Indian Constitution Fundamental Rights & its limitations.	**2**
2	**Directive Principles of State Policy:** Relevance of Directive Principles State Policy Fundamental Duties. Union Executives – President, Prime Minister Parliament Supreme Court of India.	**2**
3	**State Executives:** Governor, Chief Minister, State Legislature High Court of State. Electoral Process in India, Amendment Procedures, 42nd, 44th, 74th, 76th, 86th & 91stAmendments.	**3**
4	**Special Provisions:** Provisions for Backward class section of society, Provision for Women, Children & Backward Classes Emergency Provisions	**2**
5	**Human Rights:** Meaning and Definitions, Legislation Specific Themes in Human Rights- Working of National Human Rights Commission in India, Powers and functions of Municipalities, Panchayats and Co–Operative Societies.	**2**
6	**Scope & Aims of Engineering Ethics:** Responsibility of Engineers Impediments to Responsibility. Risks, Safety and liability of Engineers, Honesty, Integrity & Reliability in Engineering	**2**
	Total	**13**

Books Recommended:

Text books:

1. Introduction to the Constitution on India, (Students Edn.) by Durga Das Basu: Prentice –Hall EEE, 19th / 20th Edn., 2001.
2. Engineering Ethics, by Charles E. Haries, Michael S Pritchard and Michael J. Robins Thompson Asia, 2003-08-05.

Reference Books:

1. An Introduction to Constitution of India, by M. V. Pylee, Vikas Publishing, 2002.
2. Engineering Ethics, by M. Govindarajan, S. Natarajan, V. S. Senthilkumar, Prentice – Hall of India Pvt. Ltd. New Delhi, 2004.
3. Introduction to the Constitution of India, by Brij Kishore Sharma, PHI Learning Pvt. Ltd., New Delhi, 2011.
4. Latest Publications of Indian Institute of Human Rights, New Delhi

Prepared by Checked by Head of the Department Principal

Program: Computer Engineering	S.Y B.Tech.	Semester: IV
Course: Engineering Mathematics - IV (DJS22CEC401)		
Course: Engineering Mathematics - IV Tutorial (DJS22CET401)		

Pre-requisite: Knowledge of

1. Calculus
2. Descriptive Statistics
3. Basics of probability

Course Objectives:

The objective of this course is to introduce students to the concepts of Eigenvalues and Eigenvectors of Matrices, probability, test of hypothesis and correlation between data.

Outcomes: On completion of the course, learner will be able to:

1. Demonstrate ability to manipulate matrices and compute Eigen values and Eigen vectors. Use matrix algebra with its specific rules to solve the system of linear equation, using concept of Eigen value and Eigen vector to the engineering problems.
2. Apply the concept of probability distribution to the engineering problems.
3. Draw conclusions on population based on large and small samples taken and hence use it to understand data science.
4. Apply the concept of Optimization, Correlation and Regression to the engineering problems.

Engineering Mathematics - IV (DJS22CEC401)		
Unit	**Description**	**Duration**
1	**Matrices:** 1.1 Eigen values and Eigen vectors, Cayley-Hamilton theorem (without proof). 1.2 Similar matrices, diagonalizable of matrix. 1.3 Functions of square matrix. 1.4 Matrix decomposition, Singular Value Decomposition of a matrix (SVD).	8
2	**Probability:** 2.1 Random Variables: Discrete & Continuous random variables, expectation, Variance. 2.2 Probability Density Function & Cumulative Density Function. 2.3 Moments, Moment Generating Function. 2.4 Probability distribution: Binomial distribution, Poisson distribution and Normal distribution.	8
3	**Sampling Theory and ANOVA** 3.1 Test of Hypothesis, Level of significance, Critical region, One Tailed and Two Tailed test, Interval Estimation of population parameters. Large and small sample.	12

	3.2 Test of significant for Large Samples: Test for significance of the difference between sample mean and population means, Test for significance of the difference between the means of two samples. 3.3 Test of significant for small samples: Student's t-distribution and its properties. Test for significance of the difference between sample mean and population mean, Test for significance of the difference between the means of two samples, paired t-test. 3.4 Chi square test: Test of goodness of fit and independence of attributes, Contingency table. Association of attributes and Yate's correction. 3.5 Analysis of Variance(F-Test): One-way classification, Two-way classification (short-cut method).	
4	**Mathematical Programming** 4.1 Types of solution, Standard and Canonical form of LPP, Basic and feasible solutions. 4.2 Simplex method, Big –M method (method of penalty). 4.3 Duality of Linear Programming Problem. 4.4 Dual simplex method. 4.5 Non Linear Programming:-Problems with equality constrains and inequality constrains (No formulation, No Graphical method).	11
5	**Correlation & regression. (Flipped Classroom)** 5.1 Scattered diagrams, Karl Pearson's coefficient of correlation. 5.2 Spearman's Rank correlation (non-repeated and repeated ranks). 5.3 Regression coefficient & Lines of Regression.	--
	Total	39

Engineering Mathematics - IV Tutorial (DJS22CET401)	
Sr. No.	**Suggested Tutorials**
1	Matrices: Eigenvalues & Eigenvectors, Cayley Hamilton Theorem.
2	Matrices: Diagonalization and functions of square matrix.
3	Probability and Random variable
4	Probability Distribution
5	Sampling: Large Sample Test
6	Sampling: Small Sample Test
7	Sampling: Chi Square Test, ANOVA
8	LPP: Simplex Method, Big M Method
9	LPP: Duality and Dual Simplex Method
10	NLPP
11	Correlation and Regression

Minimum eight tutorials from the above suggested list or any other experiment based on syllabus will be included, which would help the learner to apply the concept learnt.

Books Recommended:

Text books:

1. Higher Engineering Mathematics by Dr. B. S. Grewal 42th edition, Khanna Publication.
2. Advanced Engineering Mathematics –Fourth Edition, Dennis G Zill & Warren S Wright.

3. Operation Research by Hira&Gupta,S Chand.
4. Probability and Statistics for Engineering, Dr. J Ravichandran, Wiley-India.

Reference Books:

1. Mathematical Methods of Science and Engineering by Kanti B. Datta, Cengage Learning.
2. Fundamentals Of Mathematical Statistics by S. C. Gupta, V. K. Kapoor, Sultan Chand & Sons - 2003
3. Probability & Statistics with reliability by Kishor s. Trivedi, Wiley India
4. Operations Research by S.D. Sharma KedarNath, Ram Nath& Co. Meerat.
5. Engineering optimization (Theory and Practice) by SingiresuS.Rao, New Age International publication

Prepared by Checked by Head of the Department Principal

Program: Computer Engineering	S.Y B.Tech.	Semester: IV
Course: Operating Systems (DJS22CEC402)		
Course: Operating Systems Laboratory (DJS22CEL402)		

Pre-requisite: Digital Electronics

Course Objectives:

1. To introduce basic concepts and functions of different operating systems.
2. To understand the concept of process, thread and resource management.
3. To understand the concepts of process synchronization and deadlock.
4. To understand various Memory, I/O and File management techniques.

Outcomes:

On successful completion of course, learner will be able to:

1. Understand role of Operating System in terms of process, memory, file and I/O management.
2. Apply appropriate process scheduling, memory mapping and disk scheduling methods.
3. Identify the need of concurrency and apply appropriate method to solve the concurrency or deadlock problem.
4. Apply and analyze different techniques of file and I/O management.

Operating Systems (DJS22CEC402)		
Unit	**Description**	**Duration**
1	**Introduction to Operating System:** Operating System Objectives, basic functions and services, Evolution of operating system, Operating System structures (monolithic, microkernel), Types of Operating Systems: Batch, multiprogramming. Multitasking, time sharing, parallel, distributed & real-time O.S., Linux OS, Android OS, Windows OS, System calls	04
2	**Process Management:** Concept of a Process, Process States, Process Description, Process Control Block, Operations on Processes. **Threads**: Definition and Types, Concept of Multithreading, **Scheduling**: Types of Scheduling: Preemptive and, Non-preemptive, Scheduling algorithms and their performance evaluation: FCFS, SJF, SRTF, Priority based, Round Robin,	07
3	**Process Synchronization:**	

	Concurrency: Principles of Concurrency, Inter-Process Communication, Process/Thread Synchronization. **Mutual Exclusion:** Requirements, Hardware and Software Support, Semaphores and Mutex, Monitors, Classical synchronization problems: Producer and Consumer problem, Readers/Writers Problem,	07
4	**Deadlock:** Principles of deadlock, Conditions for deadlock, Resource Allocation Graph, Deadlock Prevention, Deadlock Avoidance: Banker's Algorithm for Single & Multiple Resources, Deadlock Detection and Recovery. Dining Philosophers Problem.	07
5	**Memory Management** Memory Management Requirements, Memory Partitioning: Fixed Partitioning, Dynamic Partitioning, Memory Allocation Strategies: Best-Fit, First Fit, Worst Fit, Next Fit, Relocation, Paging, Segmentation. Virtual Memory: Demand Paging, Structure of Page Tables, Page Replacement Strategies: FIFO, Optimal, LRU, LFU, Thrashing.	07
6	**File System and I/O Management** **File Management**: Overview, File Organization and Access, Secondary Storage Management: File Allocation Methods **Input /Output Management** **I/O Management and Disk Scheduling:** I/O Devices, I/O Buffering, Disk Scheduling algorithm: FCFS, SSTF, SCAN, CSCAN, LOOK, C-LOOK. RAID	07
	Total	39

Operating Systems Laboratory (DJS22CEL402)	
Sr. No.	**Suggested List of Practical:**
1	Explore the internal commands of linux and Write shell scripts to do the following: Display top 10 processes in descending order Display processes with highest memory usage. Display current logged in user and logname. Display current shell, home directory, operating system type, current path setting, current working directory. Display OS version, release number, kernel version. Illustrate the use of sort, grep, awk, etc.
2	System calls for file manipulation.
3	CPU scheduling algorithms like FCFS, SJF, Round Robin etc.

4	There is a service counter which has a limited waiting queue outside it. It works as follows: • The counter remains open till the waiting queue is not empty • If the queue is already full, the new customer simply leaves • If the queue becomes empty, the outlet doors will be closed (service personnel sleep) • Whenever a customer arrives at the closed outlet, he/she needs to wake the person at the counter with a wake-up call Implement the above-described problem using semaphores or mutexes along with threads. Also show how it works, if there are 2 service personnel, and a single queue. Try to simulate all possible events that can take place, in the above scenario.
5	Implement Banker's Algorithm for deadlock avoidance
6	Implement Placement algorithms (Best, First, Worst fit)
7	Implement various page replacement policies (LRU, FIFO, Optimal)
8	Implement File allocation techniques (Sequential, Indexed, Linked)
9	Implement disk scheduling algorithm FCFS, SSTF, SCAN, CSCAN etc.
10	Using the CPU-OS simulator analyze and synthesize the following: a. Process Scheduling algorithms. b. Thread creation and synchronization. c. Deadlock prevention and avoidance.
11	Building a scheduler in XV6

Minimum eight experiments from the above suggested list or any other experiment based on syllabus will be included, which would help the learner to apply the concept learnt.

Books Recommended:

Text books:

1. William Stallings, Operating System: Internals and Design Principles, Prentice Hall, 8th Edition, 2014, ISBN-10: 0133805913 • ISBN-13: 9780133805918.
2. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, Operating System Concepts, John Wiley & Sons , Inc., 9th Edition, 2016, ISBN 978-81-265-5427-0
3. Andrew Tannenbaum, Operating System Design and Implementation, Pearson, 3rd Edition.

Reference Books:

1. Maurice J. Bach, "Design of UNIX Operating System", PHI
2. Achyut Godbole and Atul Kahate, Operating Systems, Mc Graw Hill Education, 3rd Edition
3. The Linux Kernel Book, Remy Card, Eric Dumas, Frank Mevel, Wiley Publications.

Prepared by Checked by Head of the Department Principal

Program: Computer Engineering	S.Y B.Tech.	Semester: IV
Course: Design and Analysis of Algorithms (DJS22CEC403)		
Course Design and Analysis of Algorithms Laboratory (DJS22CEL403)		

Pre-requisite: Computer Programming, Data structure

Course Objectives: The objective of the course is to introduce important algorithmic design paradigms and approaches for effective problem solving. To analyze the algorithm for its efficiency to show its effectiveness over the others. In addition, the concepts of tractable and intractable problems and the classes P, NP and NP-complete problems will be introduced.

Outcomes:

On successful completion of course, learner will be able to:

1. Analyze the performance of algorithms using asymptotic analysis.
2. Solve the problem using appropriate algorithmic design techniques.
3. Able to prove that certain problems are NP-Complete.

Design and Analysis of Algorithms (DJS22CEC403)		
Unit	**Description**	**Duration**
1	**Introduction:** Introduction to Asymptotic Analysis, Analysis of control statements and loops, solving recurrence relations using tree, substitution, master method, analysis of quick sort and merge sort Problem Solving using divide and conquer algorithm - Max-Min problem, Strassen's Matrix Multiplication.	08
2	**Greedy Method:** Introduction, control abstraction Problem solving using - fractional knapsack problem, activity selection problem, job sequencing with deadline, find a union, Minimum Spanning trees (Kruskal's algorithm, Prim's algorithm) Graphs: Single source shortest path (Dijkstra's algorithm), coin change problem.	07
3	**Dynamic Programming:** Introduction, principle of optimality, Components of dynamic programming, characteristics of dynamic programming, Fibonacci problem, Coin Changing	

	problem, 0/1 knapsack (table and set method), Multistage graphs, All pairs shortest paths (Floyd Warshall Algorithm), Single source shortest path (Bellman-Ford Algorithm), Matrix Chain Multiplication, Optimal binary search tree (OBST-successful and unsuccessful search), Travelling salesperson problem, Johnson' algorithm for Flow shop scheduling, Longest Common Subsequence (LCS), analysis of all algorithms.	10
4	**Backtracking:** Introduction, Basics of backtracking, N–queen problem, Sum of subsets, Graph coloring, Hamiltonian cycles Generating permutation, Analysis of all algorithms. **Branch-and-Bound:** Introduction, Control abstraction-LC BB, FIFO BB, LIFO BB, Properties, FIFO BB, LIFO BB, LC BB, Fifteen Puzzle problem, 0/1 Knapsack problem, Travelling Salesman problem, Job Sequencing with Deadline	07
5	**String Matching Algorithms** Introduction, The naive string-matching algorithm, The Rabin Karp algorithm, String matching with finite automata, The Knuth Morris Pratt algorithm	03
6	**Basics of Computational Complexity** Complexity classes: The class P and NP, Polynomial reduction, NP Completeness Problem, NP-Hard Problems, NP Completeness problem using Travelling Salesman problem (TSP), Approximation algorithm using TSP	04
	Total	39

Design and Analysis of Algorithms Laboratory (DJS22CEL403)	
Sr. No.	**Suggested List of Practical:**
1.	Implementation of Min Max algorithm
2.	Implementation of Strassen's Matrix Multiplication.
3.	Implementation of Karatsuba algorithm for long integer multiplication
4.	Fractional Knapsack implementation using greedy approach
5.	Implementation of Activity selection using greedy approach
6.	Implementation of Kruskal's/ Prim's algorithm using greedy approach
7.	Implementation of job sequencing with deadline using greedy approach

8.	Implementation of other greedy algorithms eg: tree vertex split, subset cover, container loading, coin changing, optimal; merge patterns (Huffman tree)
9.	Implementation of Single source shortest path (Dijkstra's algorithm)
10.	Implementation of Bellman Ford algorithm using Dynamic programming
11.	Implementation of Longest Common Subsequence algorithm using Dynamic programming
12.	Implementation of Travelling Salesperson problem using Dynamic programming
13.	Implementation of multistage graphs/ all pair shortest path using dynamic programming.
14.	Implementation of N-queen problem using Backtracking
15.	Implementation of 15 Puzzle problem using Backtracking
16.	Implementation of Knuth Morris Pratt string matching algorithm

Minimum ten experiments from the above suggested list or any other experiment based on syllabus will be included, which would help the learner to apply the concept learnt.

Books Recommended:

Textbooks:

1. T.H.Coreman , C.E. Leiserson,R.L. Rivest, and C. Stein, "Introduction to algorithms", 2nd edition, PHI publication 2005.

2. Ellis horowitz , Sartaj Sahni, S. Rajsekaran. "Fundamentals of computer algorithms" University Press

Reference Books:

1. Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani, "Algorithms", Tata McGraw- Hill Edition.

2. S. K. Basu, "Design Methods and Analysis of Algorithm", PHI.

3. John Kleinberg, Eva Tardos, "Algorithm Design", Pearson.

4. Michael T. Goodrich, Roberto Tamassia, "Algorithm Design", Wiley Publication.

Prepared by Checked by Head of the Department Principal

Program: Computer Engineering	S.Y B.Tech.	Semester: IV
Course: Processor Organization and Architecture (DJS22CEC404)		
Course: Processor Organization and Architecture Laboratory (DJS22CEL404)		

Pre-requisite: Digital Electronics, Operating systems

Course Objectives:

1. To have a thorough understanding of the basic structure and operations of a computer system.
2. To study the hierarchical memory system including cache memories and virtual memory.
3. To prepare students for higher processor architectures and embedded systems.
4. To apply innovative solutions and make progress in the knowledge to exploit the new paradigms of computing, particularly in distributed environments.

Outcomes: On successful completion of course, learner will be able to:

1. Understand the arithmetic and logic algorithms for processors.
2. Understand the concepts of memory organization and mapping techniques.
3. Explain, Interpret and implement the instructions and addressing modes of 8086 microprocessor and write assembly and mixed language programs.
4. Understand various parameters to evaluate processor performance.
5. Understand advanced trends and technologies in processor architectures.

Processor Organization and Architecture (DJS22CEC404)		
Unit	**Description**	**Duration**
1	**Introduction to Computer Architecture & Organization**: Introduction, Basic organization of computer architecture; Von Neumann model and Harvard architecture; Data Representation and Arithmetic Algorithms- Addition, Subtraction, Multiplication - unsigned multiplication, Booth's algorithm (Signed multiplication), Division of integers - restoring division, non-restoring division.	6
2	**Memory organization:** Types of RAM (SRAM, DRAM, SDRAM, DDR, SSD) and ROM; Characteristics of memory; Memory hierarchy- cost and performance measurement; Virtual Memory: Concept, Segmentation and Paging; Address translation mechanism; Interleaved and Associative memory; Cache memory Concepts, Cache Coherency	8
3	**Intel 8086 architecture and addressing modes:**	

	Major features of 8086 processor; 8086 CPU architecture and pipelined operations; programmer's model and 8086 pin description; 8086 addressing modes.	5
4	**8086 Instruction set, Interrupts and Programming:** Instruction set of 8086 microprocessor ; assembler directives; procedure and macros; Interrupts in 8086 microprocessors: Dedicated interrupts, Software interrupts, DOS interrupts (Programming examples); Assembly language programming for 8086 microprocessors.	8
5	**Introduction to Processor Performance Evaluation:** Performance metrics and benchmarks; Clock Cycle Time (CCT), Clock Rate, and Instructions Per Second (IPS), Cycles Per Instruction (CPI), Million Instructions Per Second (MIPS), Floating-Point Operations Per Second (FLOPS), Performance per Watt, Performance Evaluation with Benchmarks: Overview of popular benchmarks like SPEC, Linpack, TPC, and Stream	6
6	**Pentium Processor Architecture**: Features of Intel Pentium processor, Pentium Superscalar architecture, Pipelining, Branch prediction, Instruction and data cache concept.	6
	Total	39

Processor Organization and Architecture Laboratory (DJS22CEL404)	
Sr. No.	**Suggested List of Practical:**
1	To implement shift and add method of multiplication algorithm.
2	To implement Booth's multiplication algorithm.
3	To study and implement Restoring division algorithm.
4	To study and implement Non- Restoring Division algorithm.
5	To implement Memory allocation policies such as First Fit, Best Fit etc.
6	To study and implement page replacement policies such as FIFO, LRU etc.
7	Assembly program for 16-bit addition.
8	Assembly Program to transfer n block of data from one segment to another segment.
9	Assembly program to sort numbers in ascending/ descending order
10	Assembly program to find minimum/ maximum no. from a given array.
11	Assembly language program using Procedures and Macros.
12	Case study to evaluate the Performance of different Processors.

Minimum ten experiments from the above suggested list or any other experiment based on syllabus will be included, which would help the learner to apply the concept learnt.

Books Recommended:

1. William Stallings- "Computer Organization and Architecture: Designing for Performance", Pearson Publication, 10th Edition, 2013.
2. John P. Hayes- "Computer Architecture and Organization", McGraw-Hill, 1988.
3. John Uffenbeck – "8086/8088 family: Design Programming and Interfacing", PHI.
4. Douglas Hall- "Microprocessor and Interfacing", Tata McGraw Hill.
5. M. A. Mazidi, J. C. Mazidi, Rolin D. McKinlay- "The 8051 Microcontroller and Embedded Systems Using Assembly and C", Pearson Education, 2ndEdition.
6. Kenneth J. Ayala- "The 8051 Microcontroller", Cengage Learning India Pvt. Ltd, 3rdEdition
7. James L. Antonakos- "The Intel Microprocessor family: Hardware and Software principles and Applications", Cengage Learning.

Prepared by Checked by Head of the Department Principal

Program: Computer Engineering	**S.Y B.Tech.**	**Semester: IV**
Course: Programming Laboratory-II (Web Design) (DJS22CEL405)		

Pre-requisite: Basics of programming

Objectives:

1. To get familiar with the basics of Web Programming.
2. To acquire knowledge and skills for creation of web site considering both client and server-side programming.
3. To gain ability to develop responsive web applications.
4. To understand REST API and MongoDB for Frontend and Backend Connectivity.

Outcomes:

1. Implement interactive web page(s) using HTML5, CSS3 and JavaScript.
2. Design Web Applications using Typescript.
3. Implement Single Page Applications using Angular, React.js and Node.js Framework.
4. Create REST Web services using MongoDB.
5.

Course: Programming Laboratory-II (Web Design) (DJS22CEL405)		
1	**HTML5, CSS3 and Bootstrap:** **HTML5**: Introduction and Advantages of HTML5, Overview of New Features of HTML5, List of HTML4 elements removed from HTML5, Page Layout Semantic Elements, HTML5 Web Forms, Canvas API, SVG API, HTML5 Media (Video & Audio), Web Storage (DOM) API, Geolocation. **CSS3**: Introducing CSS3, Border and Box Effects, Background Images and Other Decorative, 2D and 3D Transformations, Transitions and Animations, Tailwind CSS, CSS3-Multi Column Layout, Media Queries **Bootstrap**: Introduction to Bootstrap, Bootstrap Grids, Bootstrap Themes, Bootstrap CSS, Bootstrap JS, Introduction to Apache Tomcat Server.	**08**
2	**JavaScript** Introduction to JavaScript, JavaScript DOM Model, Date and Objects, Regular Expressions, Exception Handling, Validation, Built-in objects, Event Handling, DHTML with JavaScript, JSON Introduction, Syntax, Function Files, Http Request, SQL.	**06**
3	**TypeScript** Overview, TypeScript Internal Architecture, TypeScript Environment Setup, TypeScript Types, variables and operators, Decision Making and loops, TypeScript Functions, TypeScript Classes and Objects, TypeScript Modules	**04**

4	**Angular** Introduction to Angular, Angular Application Architecture, what is Ng Module?, Angular Components, Angular Templates, Data Binding, Types of Data Binding Modules Component Working, Directives, Structure Directives, Template Routing, Theme Implementation in Angular Framework, Angular Forms, Services, Inject Services, Angular Server Communication with Backend Server, Working of API's (GET, POST, PUT, DELETE), Complete Web application in Angular Framework	**12**
5	**Node.js and React.js** **Node.js** : Introduction to Node.js, Installing Node.js, Node.js modules, File I/O, **Web development with Node.js**: Creating a basic web server, HTTP requests and responses, Handling dynamic content **Advanced Node.js**: Asynchronous programming with callbacks, promises, and async/await, Streams and buffers, Security and authentication **React.js**: Introduction to React.js, Installing React.js, React.js basics like JSX syntax, Components and props, State and lifecycle, Event handling **Advanced React.js**: Hooks (useState, useEffect, useContext, etc.), Redux for state management, Routing with React Router **React.js and other libraries**: Using third-party libraries with React.js (such as Material-UI or Bootstrap), Deploying a React.js app to a web server, Integration with CI/CD pipelines	**14**
6	**MongoDB** Understanding MongoDB, MongoDB Data Types, Administering User Accounts, Configuring Access Control, Adding the MongoDB Driver to Node.js, Connecting to MongoDB from Node.js, Accessing and Manipulating Databases, Manipulating MongoDB Documents from Node.js, Accessing MongoDB from Node.js, Using Mongoose for Structured Schema and Validation. REST API: Examining the rules of REST APIs, Evaluating API patterns, Handling typical CRUD functions (create, read, update, delete), Using Express and Mongoose to interact with MongoDB, Testing API endpoints	**08**
	Total	**52**

Programming Laboratory-II (Web Design) (DJS22CEL405)	
Sr. No.	**Suggested Experiment List**
1	Using HTML5 layout tags develop informative page with sections which include various images, links to other pages for navigation, make use of all possible formatting (for example font, color etc.).
2	Apply CSS properties, Border, margins, Padding, Navigation, dropdown list to page created in First Experiment.
3	Create form in HTML5 with all form elements. Apply form validations (e.g., Email, mobile, Pin code, Password) using JavaScript.
4	Inheritance, Access Modifiers example using TypeScript.

5	Building a Simple Website with TypeScript.
6	Building a simple to-do list app: Students can learn the basics of Angular.js by building a simple to-do list application. They can learn how to use directives, controllers, and services to create a functional app.
7	Use Angular.js to create a dynamic shopping cart that updates in real-time. They can learn how to use filters, directives, and controllers to add, remove, and update items in the cart.
8	Build a single-page application (SPA) using Angular.js. They can learn how to use Angular.js routing to create a navigation system that allows users to move between different views without the need for page reloads.
9	Building a simple web server: Students can learn the basics of Node.js by building a simple web server that serves static content. They can learn how to use the http module to create a server, and how to handle requests and responses.
10	Creating a real-time dashboard: Students can learn how to use Node.js to create a real-time dashboard that displays data in real-time. They can learn how to use websockets to create a two-way communication channel between the client and the server, and how to create interactive visualizations using libraries like D3.js.
11	Building a simple Blog App which has features like articles list based on most commented or most liked, author profile page and article page with comment section and reaction button. Basically, this app must have 3 different filters, based on filter chosen, the listing component must render 3 different pages. Develop using React Hooks, React Routing, Pagination and other sorting techniques.
12	Build a RESTful API using MongoDB.

Minimum ten experiments from the above suggested list or any other experiment based on syllabus will be included, which would help the learner to apply the concept learnt.

Mini Project:

Develop website using any CMS tool which falls into one of the categories blog, social networking, News updates, Wikipedia, E-commerce store. Website must include home page, and at least 3 forms (with Validation), use at least HTML5, CSS/Bootstrap, JavaScript, Angular.js, React.js web technologies. Database support is needed. Deploy website on live webserver and access through URL.

Text Books:

1. John Dean, "Web Programming with HTML5, CSS3 and JavaScript", Jones & Bartlett Learning, 2019 Edition.
2. Boris Cherny, "Programming TypeScript- Making Your Javascript Application Scale", O'Reilly Media Inc., 2019 Edition.
3. Adam Bretz and Colin J. Ihrig, "Full Stack JavaScript Development with MEAN", SitePoint Pty. Ltd., 2015 Edition.

4. Simon Holmes Clive Harber, "Getting MEAN with Mongo, Express, Angular, and Node", Manning Publications, 2019 Edition.

References:

1. Glenn Johnson, "Programming in HTML5 with JavaScript and CSS3", Microsoft Press, 2013 Edition.
2. Yakov Fain and Anton Moiseev, "TypeScript Quickly", Manning Publications, 2020 Edition.
3. Steve Fenton, "Pro TypeScript: Application - Scale Javascript Development", Apress, 2014 Edition.
4. Brad Dayley, Brendan Dayley, Caleb Dayley, "Node.js, MongoDB and Angular Web Development: The definitive guide to using the MEAN stack to build web applications", 2nd Edition, AddisonWesley Professional, 2018 Edition.

Online Resources:

1. https://www.udemy.com/course/crash-course-html-and-css/
2. https://nptel.ac.in/courses/106106156
3. https://www.udemy.com/course/reactjs-training/
4. https://www.udemy.com/course/mern-stack-course-mongodb-express-react-and-nodejs/
5. https://www.classcentral.com/course/skillshare-create-a-web-app-with-react-mongodb-express-and-nodejs-84146

Prepared by Checked by Head of the Department Principal

Program: Computer Engineering	S.Y B.Tech.	Semester: IV
Course: Universal Human Values (DJS22IHC1)		
Course: Universal Human Values Tutorial (DJS22IHT1)		

Objectives:

1. Development of a holistic perspective based on self-exploration about themselves (human being), family, society, and nature/existence.
2. Understanding (or developing clarity) of the harmony in the human being, family, society, and nature/existence.
3. Strengthening of self-reflection.
4. Development of commitment and courage to act.

Outcomes: On completion of the course, the learner will be able to:

1. Become more aware of themselves, and their surroundings (family, society, nature); they would become more responsible in life, and in handling problems with sustainable solutions, while keeping human relationships and human nature in mind. They would have better critical ability.
2. Become sensitive to their commitment towards what they have understood (human values, human relationships, and human society).
3. Apply what they have learnt to their own self in different day-to-day settings in real life, at least a beginning would be made in this direction.

Universal Human Values (DJS22IHC1)		
Unit	**Description**	**Duration**
1	**Introduction:** Need, Basic Guidelines, Content and Process for Value Education Purpose and motivation for the course. Self-Exploration–what is it? - Its content and process; 'Natural Acceptance' and Experiential Validation- as the process for self-exploration. Continuous Happiness and Prosperity- A look at basic Human Aspirations. Right understanding, Relationship and Physical Facility- the basic requirements for fulfilment of aspirations of every human being with their correct priority. Understanding Happiness and Prosperity correctly- A critical appraisal of the current scenario. Method to fulfil the above human aspirations: understanding and living in harmony at various levels.	**5**
2	**Understanding Harmony in the Human Being** - Harmony in Myself! Understanding human being as a co-existence of the sentient 'I' and the material 'Body'. Understanding the needs of Self ('I') and 'Body' - happiness and physical facility. Understanding the Body as an instrument of 'I' (I am being the doer, seer and	**5**

	enjoyer). Understanding the characteristics and activities of 'I' and harmony in 'I'. Understanding the harmony of I with the Body: Sanyam and Health; correct appraisal of Physical needs, meaning of Prosperity in detail. Programs to ensure Sanyam and Health.	
3	**Understanding Harmony in the Family**: Harmony in Human-Human Relationship. Understanding values in human-human relationship; meaning of Justice (nine universal values in relationships) and program for its fulfilment to ensure mutual happiness; Trust and Respect as the foundational values of relationship. Understanding the meaning of Trust; Difference between intention and competence. Understanding the meaning of Respect, Difference between respect and differentiation; the other salient values in relationship	**3**
4	**Understanding the harmony in society (society being an extension of family):** Resolution, Prosperity, fearlessness (trust) and co-existence as comprehensive Human Goals. Visualizing a universal harmonious order in society- Undivided Society, Universal Order- from family to world family.	**3**
5	**Understanding Harmony in Nature and Existence:** Whole existence as Coexistence Understanding the harmony in the Nature. Interconnectedness and mutual fulfilment among the four orders of nature recyclability and self-regulation in nature. Understanding Existence as Co-existence of mutually interacting units in all pervasive space. Holistic perception of harmony at all levels of existence.	**5**
6	**Implications of the above Holistic Understanding of Harmony on Professional Ethics:** Natural acceptance of human values. Definitiveness of Ethical Human Conduct. Basis for Humanistic Education, Humanistic Constitution and Humanistic Universal Order. Competence in professional ethics: a. Ability to utilize the professional competence for augmenting universal human order, b. Ability to identify the scope and characteristics of people friendly and eco-friendly production systems, c. Ability to identify and develop appropriate technologies and management patterns for above production systems. Case studies of typical holistic technologies, management models and production systems. Strategy for transition from the present state to Universal Human Order: a. At the level of individual: as socially and ecologically responsible engineers, technologists, and managers, b. At the level of society: as mutually enriching institutions and organizations.	**5**
	Total	**26**

Tutorials: (Term work)

Term work shall consist of minimum 5 activities based on activities conducted.
The tutorials could be conducted as per the following topics: -

Universal Human Values Tutorial (DJS22IHT1)	
Activity No 1	Practice sessions to discuss natural acceptance in human being as the innate acceptance for living with responsibility (living in relationship, harmony, and co-existence) rather than as arbitrariness in choice based on liking-disliking.
Activity No 2	Practice sessions to discuss the role others have played in making material goods available to me. Identifying from one's own life. Differentiate between prosperity and accumulation. Discuss program for ensuring health vs dealing with disease.
Activity No 3	Practice sessions to reflect on relationships in family, hostel and institute as extended family, real life examples, teacher-student relationship, goal of education etc. Gratitude as a universal value in relationships. Discuss with scenarios. Elicit examples from students' lives.
Activity No 4	Practice sessions to discuss human being as cause of imbalance in nature (film "Home" can be used), pollution, depletion of resources and role of technology etc.
Activity No 5	Practice Exercises and Case Studies will be taken up in Practice (tutorial) Sessions e.g. To discuss the conduct as an engineer or scientist etc.

The final certification and acceptance of term work will be subject to satisfactory performance of activities and upon fulfilling minimum passing criteria in the term work.

Books Recommended:

Text books:

1. Human Values and Professional Ethics by R R Gaur, R Sangal, G P Bagaria, Excel Books, New Delhi, 2010.

Reference Books:

1. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, Amarkantak, 1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi.
5. Small is Beautiful - E. F Schumacher. 6. Slow is Beautiful - Cecile Andrews.
7. Economy of Permanence - J C Kumarappa.
8. Bharat Mein Angreji Raj – PanditSunderlal.
9. Rediscovering India - by Dharampal.
10. Hind Swaraj or Indian Home Rule - by Mohandas K. Gandhi.

11.India Wins Freedom - Maulana Abdul Kalam Azad.
12.Vivekananda - Romain Rolland. (English)
13.Gandhi - Romain Rolland. (English)

Program: Computer Engineering	S.Y B.Tech.	Semester: IV
Course: Innovative Product Development-II (DJS22ILLA2)		

Objectives:

1. To acquaint the students with the process of identifying the need (considering a societal requirement) and ensuring that a solution is found out to address the same by designing and developing an innovative product.
2. To familiarize the students with the process of designing and developing a product, while they work as part of a team.
3. To acquaint the students with the process of applying basic engineering fundamentals, so as to attempt at the design and development of a successful value-added product.
4. To inculcate the basic concepts of entrepreneurship and the process of self-learning and research required to conceptualize and create a successful product.

Outcomes: On completion of the course, the learner will be able to:

1. Identify the requirement for a product based on societal/research needs.
2. Apply knowledge and skills required to solve a societal need by conceptualizing a product, especially while working in a team.
3. Use standard norms of engineering concepts/practices in the design and development of an innovative product.
4. Draw proper inferences through theoretical/ experimental/simulations and analyze the impact of the proposed method of design and development of the product.
5. Develop interpersonal skills, while working as a member of the team or as the leader.
6. Demonstrate capabilities of self-learning as part of the team, leading to life-long learning, which could eventually prepare themselves to be successful entrepreneurs.
7. Demonstrate product/project management principles during the design and development work and also excel in written (Technical paper preparation) as well as oral communication.

Guidelines for the proposed product design and development:

- Students shall form a team of 3 to 4 students (max allowed: 5-6 in extraordinary cases, subject to the approval of the department review committee and the Head of the department).
- Students should carry out a survey and identify the need, which shall be converted into conceptualization of a product, in consultation with the faculty supervisor/head of department/internal committee of faculty members.
- Students in the team shall understand the effective need for product development and accordingly select the best possible design in consultation with the faculty supervisor.

- Students shall convert the best design solution into a working model, using various components drawn from their domain as well as related interdisciplinary areas.
- Faculty supervisor may provide inputs to students during the entire span of the activity, spread over 2 semesters, wherein the main focus shall be on self-learning.
- A record in the form of an activity log-book is to be prepared by each team, wherein the team can record weekly progress of work. The guide/supervisor should verify the recorded notes/comments and approve the same on a weekly basis.
- The design solution is to be validated with proper justification and the report is to be compiled in a standard format and submitted to the department. Efforts are to be made by the students to try and publish a technical paper, either in the institute journal, "Techno Focus: Journal for Budding Engineers" or at a suitable publication, approved by the department research committee/ Head of the department.
- The focus should be on self-learning, capability to design and innovate new products as well as on developing the ability to address societal problems. Advancement of entrepreneurial capabilities and quality development of the students through the year long course should ensure that the design and development of a product of appropriate level and quality is carried out, spread over two semesters, i.e. during the semesters III and IV.

Guidelines for Assessment of the work:

- The review/ progress monitoring committee shall be constituted by the Head of the Department. The progress of design and development of the product is to be evaluated on a continuous basis, holding a minimum of two reviews in each semester.
- In the continuous assessment, focus shall also be on each individual student's contribution to the team activity, their understanding and involvement as well as responses to the questions being raised at all points in time.
- Distribution of marks individually for the both reviews as well as for the first review during the subsequent semester shall be as given below:
    - Marks awarded by the supervisor based on log-book: 20
    - Marks awarded by review committee: 20
    - Quality of the write-up: 10

In the last review of the semester IV, the marks will be awarded as follows.

- Marks awarded by the supervisor (Considering technical paper writing): 30
- Marks awarded by the review committee : 20

A candidate needs to secure a minimum of 50% marks to be declared to have completed the audit course.

Review/progress monitoring committee may consider the following points during the assessment.

- In the semester IV, the expected work shall be procurement of components/systems, building of the working prototype, testing and validation of the results based on work completed in semester III.
  - First review is based on readiness of building the working prototype.
  - Second review shall be based on a presentation as well as the demonstration of the working model, during the last month of semester IV. This review will also look at the readiness of the proposed technical paper presentation of the team.

The overall work done by the team shall be assessed based on the following criteria;

1. Quality of survey/ need identification of the product.
2. Clarity of Problem definition (design and development) based on need.
3. Innovativeness in the proposed design.
4. Feasibility of the proposed design and selection of the best solution.
5. Cost effectiveness of the product.
6. Societal impact of the product.
7. Functioning of the working model as per stated requirements.
8. Effective use of standard engineering norms.
9. Contribution of each individual as a member or the team leader.
10. Clarity on the write-up and the technical paper prepared.

• The semester reviews may be based on relevant points listed above, as applicable.

Guidelines for Assessment of Semester Reviews:

- The write-up should be prepared as per the guidelines given by the department.
- The design and the development of the product shall be assessed through a presentation and demonstration of the working model by the student team to a panel of Internal and External Examiners, preferably from industry or any research organizations having an experience of more than five years, approved by the Head of the Institution. The presence of the external examiner is desirable only for the 2nd presentation in semester IV. Students are required to present the outline of the technical paper prepared by them during the final review in semester IV.

Prepared by Checked by Head of the Department Principal

Program: Computer Engineering	S.Y B.Tech.	Semester: IV
Course: Environmental Studies (DJS22A3)		

Pre-requisite: Interest in Environment and its impact on Human

Objectives:

1. Understand environmental issues such as depleting resources, pollution, ecological problems and the renewable energy scenario.
2. Familiarise environment related legislation

Outcomes: On completion of the course, learner will be able to:

1. Understand how human activities affect environment
2. Understand the various technology options that can make aa difference

Environmental Studies (DJS22A3)		
Unit	**Description**	**Duration**
1	**Social Issues and Environment:** Ecological footprint and Carrying Capacity, Depleting nature of Environmental resources such as soil, water minerals and forests, Carbon emissions and Global Warming.	4
2	**Technological Growth for Sustainable Development:** Social, Economical and Environmental aspects of Sustainable Development, Renewable Energy Harvesting, Concept of Carbon credit, Green Building, Power and functions of Central Pollution Control Board and State Pollution Control Board .	4
3	**Green Technology:** History, Agenda, and Challenges Ahead. Sustainable Cloud Computing, and Risk Management, Sustainable Software Design, Data Center Energy Efficiency, Thin-Client and Energy Efficiency.	5
	Total	13

Books Recommended:

Text books:

1. Environmental Studies From Crisis to Cure, R. Rajagopalan, 2012
2. Textbook for Environmental Studies For Undergraduate Courses of all Branches of Higher Education, Erach Bharucha
3. Green Information Technology A Sustainable Approach, Mohammad Dastbaz, Colin Pattinson, Babak Akhgar, Morgan and Kaufman, Elsevier, 2015.

Reference Books:

1. Information Technologies in Environmental Engineering: New Trends and Challenges, Paulina Golinska, Marek Fortsch, Jorge Marx-Gómez, Springer, 2011.

Prepared by Checked by Head of the Department Principal