



SAI RAJESWARI INSTITUTE OF TECHNOLOGY

(AUTONOMOUS)

Lingapuram (V), Proddatur, Y S R District – 516 362, A.P.

Course Structure

Computer Science and Engineering

B. Tech. (Regular – Full time)

(Effective for the students into II year from the
Academic Year 2023-24 onwards)

&

B. Tech. (Lateral Entry Scheme)

(Effective for the students admitted into II year through Lateral Entry Scheme from
the Academic Year 2024-25 onwards)

B.TECH - COURSE STRUCTURE–R23

(Applicable from the academic year 2023-24 onwards)

B.Tech (CSE) – II Year I Semester

S.No.	Category	Code	Title	L	T	P	Credits	CIA	End Exam Marks	Total Marks
1	Basic Sciences	23MA3BS05	Discrete Mathematics & Graph Theory	3	0	0	3	30	70	100
2	HSMC	23BA3HM01	Universal Human Values– Understanding Harmony & Human Ethical Conduct	2	1	0	3	30	70	100
3	Engineering Science	23CS3ES01	Digital Logic & Computer Organization	3	0	0	3	30	70	100
4	Professional Core	23CS3PC01	Advanced Data Structures & Algorithm Analysis	3	0	0	3	30	70	100
5	Professional Core	23CS3PC02	Object Oriented Programming Through Java	3	0	0	3	30	70	100
6	Professional Core	23CS3PC03	Advanced Data Structures and Algorithm Analysis Lab	0	0	3	1.5	30	70	100
7	Professional Core	23CS3PC04	Object Oriented Programming Through Java Lab	0	0	3	1.5	30	70	100
8	Skill Enhancement Course	23CS3SC01	Python Programming	0	1	2	2	30	70	100
9	Audit Course	23HS3MC01	Environmental Science	2	0	0	-	30	70	100
Total				16	2	8	20			

B.Tech(CSE) – II Year II Semester

S.No.	Category	Code	Title	L	T	P	Credits	CIA	End Exam Marks	Total Marks
1	Management Elective- I	23BA4HM01A/B/C	Managerial Economics and Financial Analysis / Business Environment / Organizational Behavior	2	0	0	2	30	70	100
2	Basic Science	23MA4BS01	Probability & Statistics	3	0	0	3	30	70	100
3	Professional Core	23CS4PC01	Operating Systems	3	0	0	3	30	70	100
4	Professional Core	23CS4PC02	Database Management Systems	3	0	0	3	30	70	100
5	Professional Core	23CS4PC03	Software Engineering	3	0	0	3	30	70	100
6	Professional Core	23CS4PC04	Operating Systems Lab	0	0	3	1.5	30	70	100
7	Professional Core	23CS4PC05	Database Management Systems Lab	0	0	3	1.5	30	70	100
8	Skill Enhancement Course	23CS4SC01	Full Stack Development –I	0	1	2	2	30	70	100
9	BS&H	23BA4HM03	Design Thinking & Innovation	1	0	2	2	30	70	100
Total				15	1	10	21			
Mandatory Community Service Project Internship of 08 weeks duration during summer vacation										



SAI RAJESWARI INSTITUTE OF TECHNOLOGY

(AUTONOMOUS)

Lingapuram (V), Proddatur, Y S R District – 516 362, A.P.

B. Tech. (Regular – Full time)

**(Effective for the students admitted into II year from the
Academic Year 2023-24 onwards)**

**Department of Computer Science & Engineering
SECOND YEAR SYLLABUS**

DISCRETE MATHEMATICS & GRAPH THEORY
(Common to CSE and all CSE allied branches)

II Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23MA3 BS05	Basic Sciences	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		3	0	0	3	30	70	100
Sessional Exam Duration :1Hr 50 Min					End Exam Duration: 3 Hrs			
Course Objectives:								
Course Outcomes: A student after completion of the course will be able to								
CO1	Apply mathematical logic to solve problems.							
CO2	Understand the concepts and perform the operations related to sets, relations and functions. Gain the conceptual background needed and identify structures of algebraic nature.							
CO3	Apply basic counting techniques to solve combinatorial problems.							
CO4	Formulate problems and solve recurrence relations.							
CO5	Apply Graph Theory in solving computer science problems							
UNIT-I	Mathematical Logic							
Introduction, Statements and Notation, Connectives, Well-formed formulas, Tautology, Duality law, Equivalence, Implication, Normal Forms, Functionally complete set of connectives, Inference Theory of Statement Calculus, Predicate Calculus, Inference theory of Predicate Calculus.								
UNIT-II	Set theory							
The Principle of Inclusion- Exclusion, Pigeon hole principle and its application, Functions composition of functions, Inverse Functions, Recursive Functions, Lattices and its properties. Algebraic structures: Algebraic systems-Examples and General Properties, Semi groups and Monoids, groups, sub groups, homomorphism, Isomorphism.								
UNIT-III	Elementary Combinatorics							
Combinations and Permutations, Enumeration of Combinations and Permutations, Enumerating Combinations and Permutations with Repetitions, Enumerating Permutations with Constrained Repetitions, Binomial Coefficients, The Binomial and Multinomial Theorems.								
UNIT-IV	Recurrence Relations							
Generating Functions of Sequences, Calculating Coefficients of Generating Functions, Recurrence relations, Solving Recurrence Relations by Substitution and Generating functions, The Method of Characteristic roots, Solutions of Inhomogeneous , Recurrence Relations.								
UNIT-V	Graphs							
Basic Concepts, Isomorphism and Subgraphs, Trees and their Properties, Spanning Trees, Directed Trees, Binary Trees, Planar Graphs, Euler's Formula, Multigraphs and Euler Circuits, Hamiltonian Graphs.								
Textbooks:								
1. J.P. Tremblay and R. Manohar, Discrete Mathematical Structures with Applications to Computer Science, Tata McGraw Hill, 2002.								
2. Kenneth H. Rosen, Discrete Mathematics and its Applications with Combinatorics and Graph Theory, 7th Edition, McGraw Hill Education (India) Private Limited.								

Reference Books:

1. Joe L. Mott, Abraham Kandel and Theodore P. Baker, Discrete Mathematics for Computer Scientists & Mathematicians, 2nd Edition, Pearson Education.
2. Narsingh Deo, Graph Theory with Applications to Engineering and Computer Science.

Online Learning Resources:

<http://www.cs.yale.edu/homes/aspnes/classes/202/notes.pdf>

SRIT (SR)

UNIVERSAL HUMAN VALUES – UNDERSTANDING HARMONY AND ETHICAL HUMAN CONDUCT

II Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23BA3 HM01	HSMC	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		2	1	0	3	30	70	100
Sessional Exam Duration : 1Hr 50 Min					End Exam Duration: 3 Hrs			
Course Objectives:								
<ul style="list-style-type: none"> To help the students appreciate the essential complementary between 'VALUES' and 'SKILLS' to ensure sustained happiness and prosperity which are the core aspirations of all human beings. To facilitate the development of a Holistic perspective among students towards life and profession as well as towards happiness and prosperity based on a correct understanding of the Human reality and the rest of existence. Such holistic perspective forms the basis of Universal Human Values and movement towards value-based living in a natural way. To highlight plausible implications of such a Holistic understanding in terms of ethical human conduct, trustful and mutually fulfilling human behaviour and mutually enriching interaction with Nature. 								
Course Outcomes: A student after completion of the course will be able to								
CO1	Define the terms like Natural Acceptance, Happiness and Prosperity (L1, L2)							
CO2	Identify one's self, and one's surroundings (family, society nature) (L1, L2)							
CO3	Apply what they have learnt to their own self in different day-to-day settings in real life (L3)							
CO4	Relate human values with human relationship and human society. (L4)							
CO5	Justify the need for universal human values and harmonious existence (L5)							
CO6	Develop as socially and ecologically responsible engineers (L3, L6)							
Course Topics:								
<p>The course has 28 lectures and 14 tutorials in 5 modules. The lectures and tutorials are of 1-hour duration. Tutorial sessions are to be used to explore and practice what has been proposed during the lecture sessions.</p> <p>The Teacher's Manual provides the outline for lectures as well as practice sessions. The teacher is expected to present the issues to be discussed as propositions and encourage the students to have a dialogue.</p>								
UNIT-I	Introduction to Value Education (6 lectures and 3 tutorials for practice session)							
<p>Lecture 1: Right Understanding, Relationship and Physical Facility (Holistic Development and the Role of Education)</p> <p>Lecture 2: Understanding Value Education</p> <p>Tutorial 1: Practice Session PS1 Sharing about Oneself</p> <p>Lecture 3: self-exploration as the Process for Value Education</p> <p>Lecture 4: Continuous Happiness and Prosperity – the Basic Human Aspirations</p> <p>Tutorial 2: Practice Session PS2 Exploring Human Consciousness</p> <p>Lecture 5: Happiness and Prosperity – Current Scenario</p> <p>Lecture 6: Method to Fulfill the Basic Human Aspirations</p> <p>Tutorial 3: Practice Session PS3 Exploring Natural Acceptance</p>								

UNIT-II	Harmony in the Human Being (6 lectures and 3 tutorials for practice session)
	<p>Lecture 7: Understanding Human being as the Co-existence of the self and the body.</p> <p>Lecture 8: Distinguishing between the Needs of the self and the body</p> <p>Tutorial 4: Practice Session PS4 Exploring the difference of Needs of self and body.</p> <p>Lecture 9: The body as an Instrument of the self</p> <p>Lecture 10: Understanding Harmony in the self</p> <p>Tutorial 5: Practice Session PS5 Exploring Sources of Imagination in the self</p> <p>Lecture 11: Harmony of the self with the body</p> <p>Lecture 12: Programme to ensure self-regulation and Health</p> <p>Tutorial 6: Practice Session PS6 Exploring Harmony of self with the body</p>
UNIT-III	Harmony in the Family and Society (6 lectures and 3 tutorials for practice session)
	<p>Lecture 13: Harmony in the Family – the Basic Unit of Human Interaction</p> <p>Lecture 14: 'Trust' – the Foundational Value in Relationship</p> <p>Tutorial 7: Practice Session PS7 Exploring the Feeling of Trust</p> <p>Lecture 15: 'Respect' – as the Right Evaluation</p> <p>Tutorial 8: Practice Session PS8 Exploring the Feeling of Respect</p> <p>Lecture 16: Other Feelings, Justice in Human-to-Human Relationship</p> <p>Lecture 17: Understanding Harmony in the Society</p> <p>Lecture 18: Vision for the Universal Human Order</p> <p>Tutorial 9: Practice Session PS9 Exploring Systems to fulfil Human Goal</p>
UNIT-IV	Harmony in the Nature/Existence (4 lectures and 2 tutorials for practice session)
	<p>Lecture 19: Understanding Harmony in the Nature</p> <p>Lecture 20: Interconnectedness, self-regulation and Mutual Fulfilment among the Four Orders of Nature</p> <p>Tutorial 10: Practice Session PS10 Exploring the Four Orders of Nature</p> <p>Lecture 21: Realizing Existence as Co-existence at All Levels</p> <p>Lecture 22: The Holistic Perception of Harmony in Existence</p> <p>Tutorial 11: Practice Session PS11 Exploring Co-existence in Existence.</p>
UNIT-V	Implications of the Holistic Understanding – a Look at Professional Ethics (6 lectures and 3 tutorials for practice session)
	<p>Lecture 23: Natural Acceptance of Human Values</p> <p>Lecture 24: Definitiveness of (Ethical) Human Conduct</p> <p>Tutorial 12: Practice Session PS12 Exploring Ethical Human Conduct</p> <p>Lecture 25: A Basis for Humanistic Education, Humanistic Constitution and Universal Human Order</p> <p>Lecture 26: Competence in Professional Ethics</p> <p>Tutorial 13: Practice Session PS13 Exploring Humanistic Models in Education</p> <p>Lecture 27: Holistic Technologies, Production Systems and Management Models-Typical Case Studies</p> <p>Lecture 28: Strategies for Transition towards Value-based Life and Profession</p> <p>Tutorial 14: Practice Session PS14 Exploring Steps of Transition towards Universal Human Order</p>

Practice Sessions for UNIT I – Introduction to Value Education

PS1 Sharing about Oneself

PS2 Exploring Human Consciousness

PS3 Exploring Natural Acceptance

Practice Sessions for UNIT II – Harmony in the Human Being

PS4 Exploring the difference of Needs of self and body

PS5 Exploring Sources of Imagination in the self

PS6 Exploring Harmony of self with the body

Practice Sessions for UNIT III – Harmony in the Family and Society

PS7 Exploring the Feeling of Trust

PS8 Exploring the Feeling of Respect

PS9 Exploring Systems to fulfil Human Goal

Practice Sessions for UNIT IV – Harmony in the Nature (Existence)

PS10 Exploring the Four Orders of Nature

PS11 Exploring Co-existence in Existence

Practice Sessions for UNIT V – Implications of the Holistic Understanding – a Look at Professional Ethics

PS12 Exploring Ethical Human Conduct

PS13 Exploring Humanistic Models in Education

PS14 Exploring Steps of Transition towards Universal Human Order

READINGS:

Textbook and Teachers Manual

a. The Textbook

R R Gaur, R Asthana, G P Bagaria, *A Foundation Course in Human Values and Professional Ethics*, 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93-87034-47-1

b. The Teacher's Manual

R R Gaur, R Asthana, G P Bagaria, *Teachers' Manual for A Foundation Course in Human Values and Professional Ethics*, 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93-87034-53-2

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* – Pandit Sunderlal
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)

Mode of Conduct:

Lecture hours are to be used for interactive discussion, placing the proposals about the topics at hand and motivating students to reflect, explore and verify them.

Tutorial hours are to be used for practice sessions.

While analyzing and discussing the topic, the faculty mentor's role is in pointing to essential elements to help in sorting them out from the surface elements. In other words, help the students explore the important or critical elements.

In the discussions, particularly during practice sessions (tutorials), the mentor encourages the student to connect with one's own self and do self-observation, self-reflection and self-exploration.

Scenarios may be used to initiate discussion. The student is encouraged to take up "ordinary" situations rather than "extra-ordinary" situations. Such observations and their analyses are shared and discussed with other students and faculty mentor, in a group sitting.

Tutorials (experiments or practical) are important for the course. The difference is that the laboratory is everyday life, and practical are how you behave and work in real life. Depending on the nature of topics, worksheets, home assignment and/or activity are included. The practice sessions (tutorials) would also provide support to a student in performing actions commensurate to his/her beliefs. It is intended that this would lead to development of commitment, namely behaving and working based on basic human values.

It is recommended that this content be placed before the student as it is, in the form of a basic foundation course, without including anything else or excluding any part of this content. Additional content may be offered in separate, higher courses. This course is to be taught by faculty from every teaching department, not exclusively by any one department.

Teacher preparation with a minimum exposure to at least one 8-day Faculty Development Program on Universal Human Values is deemed essential.

Online Resources:

1. <https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%201-Introduction%20to%20Value%20Education.pdf>
2. <https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%202-Harmony%20in%20the%20Human%20Being.pdf>
3. <https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%203-Harmony%20in%20the%20Family.pdf>
4. <https://fdp-si.aicte-india.org/UHV%201%20Teaching%20Material/D3-S2%20Respect%20July%202023.pdf>
5. <https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%205-Harmony%20in%20the%20Nature%20and%20Existence.pdf>
6. <https://fdp-si.aicte-india.org/download/FDPTeachingMaterial/3-days%20FDP-SI%20UHV%20Teaching%20Material/Day%203%20Handouts/UHV%203D%20D3-S2A%20Und%20Nature-Existence.pdf>
7. <https://fdp-si.aicte-india.org/UHV%20II%20Teaching%20Material/UHV%20II%20Lecture%202023-25%20Ethics%20v1.pdf>
8. <https://www.studocu.com/in/document/kiet-group-of-institutions/universal-human-values/chapter-5-holistic-understanding-of-harmony-on-professional-ethics/62490385>
9. https://onlinecourses.swayam2.ac.in/aic22_ge23/preview

Digital Logic & Computer Organization

II Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23CS3E S01	Engineering Science	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		3	0	0	3	30	70	100
Sessional Exam Duration :1Hr 50 Min					End Exam Duration: 3 Hrs			
Course Objectives: The main objectives of the course is to <ul style="list-style-type: none"> • provide students with a comprehensive understanding of digital logic design principles and computer organization fundamentals • Describe memory hierarchy concepts • Explain input/output (I/O) systems and their interaction with the CPU, memory, and peripheral devices. 								
Course Outcomes: A student after completion of the course will be able to								
CO1	Differentiate between combinational and sequential circuits based on their characteristics and functionalities. (L2)							
CO2	Demonstrate an understanding of computer functional units. (L2)							
CO3	Analyze the design and operation of processors, including instruction execution, pipelining, and control unit mechanisms, to comprehend their role in computer systems.(L3)							
CO4	Describe memory hierarchy concepts, including cache memory, virtual memory, and secondary storage, and evaluate their impact on system performance and scalability. (L3)							
CO5	Explain input/output (I/O) systems and their interaction with the CPU, memory, and peripheral devices, including interrupts, DMA, and I/O mapping techniques. (L3)							
UNIT-I								
Data Representation: Binary Numbers, Fixed Point Representation. Floating Point Representation. Number base conversions, Octal and Hexadecimal Numbers, components, Signed binary numbers, Binary codes Digital Logic Circuits-I: Basic Logic Functions, Logic gates, universal logic gates, Minimization of Logic expressions. K-Map Simplification, Combinational Circuits, Decoders, Multiplexers								
UNIT-II								
Digital Logic Circuits-II: Sequential Circuits, Flip-Flops, Binary counters, Registers, Shift Registers, Ripple counters Basic Structure of Computers: Computer Types, Functional units, Basic operational concepts, Bus structures, Software, Performance, multiprocessors and multi computers, Computer Generations, Von- Neumann Architecture								
UNIT-III								
Computer Arithmetic : Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed-operand Multiplication, Fast Multiplication, Integer Division, Floating-Point Numbers and Operations								

Processor Organization: Fundamental Concepts, Execution of a Complete Instruction, Multiple-Bus Organization, Hardwired Control and Multi programmed Control	
UNIT-IV	
The Memory Organization: Basic Concepts, Semiconductor RAM Memories, Read-Only Memories, Speed, Size and Cost, Cache Memories, Performance Considerations, Virtual Memories, Memory Management Requirements, Secondary Storage	
UNIT-V	
Input/Output Organization: Accessing I/O Devices, Interrupts, Processor Examples, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces	
<p>Textbooks:</p> <ol style="list-style-type: none"> 1. Computer Organization, Carl Hamacher, Zvonko Vranesic, Safwat Zaky, 6th edition, McGraw Hill, 2023. 2. Digital Design, 6th Edition, M. Morris Mano, Pearson Education, 2018. 3. Computer Organization and Architecture, William Stallings, 11th Edition, Pearson, 2022. <p>Reference Books:</p> <ol style="list-style-type: none"> 1. Computer Systems Architecture, M. Morris Mano, 3rd Edition, Pearson, 2017. 2. Computer Organization and Design, David A. Paterson, John L. Hennessy, Elsevier, 2004. 3. Fundamentals of Logic Design, Roth, 5th Edition, Thomson, 2003. <p>Online Learning Resources: https://nptel.ac.in/courses/106/103/106103068/</p>	

Advanced Data Structures & Algorithm Analysis

I Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23CS3P C01	Professional Core	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		3	0	0	3	30	70	100
Sessional Exam Duration : 1Hr 50 Min					End Exam Duration: 3 Hrs			
<p>Course Objectives: The main objectives of the course is to</p> <ul style="list-style-type: none"> • provide knowledge on advance data structures frequently used in Computer Science domain • Develop skills in algorithm design techniques popularly used • Understand the use of various data structures in the algorithm design 								
Course Outcomes: A student after completion of the course will be able to								
CO1	Illustrate the working of the advanced tree data structures and their applications (L2)							
CO2	Understand the Graph data structure, traversals and apply them in various contexts. (L2)							
CO3	Use various data structures in the design of algorithms (L3)							
CO4	Recommend appropriate data structures based on the problem being solved (L5)							
CO5	Analyze algorithms with respect to space and time complexities (L4)							
UNIT – I								
Introduction to Algorithm Analysis, Space and Time Complexity analysis, Asymptotic Notations AVL Trees – Creation, Insertion, Deletion operations and Applications B-Trees – Creation, Insertion, Deletion operations and Applications								
UNIT – II								
Heap Trees (Priority Queues) – Min and Max Heaps, Operations and Applications Graphs – Terminology, Representations, Basic Search and Traversals, Connected Components and Biconnected Components, applications Divide and Conquer: The General Method, Quick Sort, Merge Sort, Strassen’s matrix multiplication, Convex Hull								
UNIT – III								
Greedy Method: General Method, Job Sequencing with deadlines, Knapsack Problem, Minimum cost spanning trees, Single Source Shortest Paths Dynamic Programming: General Method, All pairs shortest paths, Single Source Shortest Paths – General Weights (Bellman Ford Algorithm), Optimal Binary Search Trees, 0/1 Knapsack, String Editing, Travelling Salesperson problem								
UNIT – IV								
Backtracking: General Method, 8-Queens Problem, Sum of Subsets problem, Graph Coloring, 0/1 Knapsack Problem Branch and Bound: The General Method, 0/1 Knapsack Problem, Travelling Salesperson problem								
UNIT – V								
NP Hard and NP Complete Problems: Basic Concepts, Cook’s theorem NP Hard Graph Problems: Clique Decision Problem (CDP), Chromatic Number Decision Problem (CNDP), Traveling Salesperson Decision Problem (TSP) NP Hard Scheduling Problems: Scheduling Identical Processors, Job Shop Scheduling								
Textbooks:								

1. Fundamentals of Data Structures in C++, Horowitz, Ellis; Sahni, Sartaj; Mehta, Dinesh 2nd Edition Universities Press
2. Computer Algorithms/C++ Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran 2nd Edition University Press

Reference Books:

1. Data Structures and program design in C, Robert Kruse, Pearson Education Asia
2. An introduction to Data Structures with applications, Trembley & Sorenson, McGraw Hill
3. The Art of Computer Programming, Vol.1: Fundamental Algorithms, Donald E Knuth, Addison-Wesley, 1997.
4. Data Structures using C & C++: Langsam, Augenstein & Tanenbaum, Pearson, 1995
5. Algorithms + Data Structures & Programs:, N.Wirth, PHI
6. Fundamentals of Data Structures in C++: Horowitz Sahni & Mehta, Galgottia Pub.
7. Data structures in Java:, Thomas Standish, Pearson Education Asia

Online Learning Resources:

1. https://www.tutorialspoint.com/advanced_data_structures/index.asp
2. <http://peterindia.net/Algorithms.html>
3. Abdul Bari, [1. Introduction to Algorithms \(youtube.com\)](https://www.youtube.com/watch?v=1. Introduction to Algorithms)

Object-Oriented Programming through JAVA

I Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23CS3P C02	Professional Core	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		3	0	0	3	30	70	100
Sessional Exam Duration : 1Hr 50 Min					End Exam Duration: 3 Hrs			
<p>Course Objectives: The learning objectives of this course are to:</p> <ul style="list-style-type: none"> identify Java language components and how they work together in applications learn the fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries. learn how to extend Java classes with inheritance and dynamic binding and how to use exception handling in Java applications understand how to design applications with threads in Java understand how to use Java APIs for program development 								
Course Outcomes: A student after completion of the course will be able to								
CO1	Analyze problems, design solutions using OOP principles, and implement them efficiently in Java. (L4)							
CO2	Design and implement classes to model real-world entities, with a focus on attributes, behaviours, and relationships between objects (L4)							
CO3	Demonstrate an understanding of inheritance hierarchies and polymorphic behaviour, including method overriding and dynamic method dispatch. (L3)							
CO4	Apply Competence in handling exceptions and errors to write robust and fault-tolerant code. (L3)							
CO5	Perform file input/output operations, including reading from and writing to files using Java I/O classes, graphical user interface (GUI) programming using JavaFX.(L3)							
UNIT-I	Object Oriented Programming							
<p>Basic concepts, Principles, Program Structure in Java: Introduction, Writing Simple Java Programs, Elements or Tokens in Java Programs, Java Statements, Command Line Arguments, User Input to Programs, Escape Sequences Comments, Programming Style.</p> <p>Data Types, Variables, and Operators :Introduction, Data Types in Java, Declaration of Variables, Data Types, Type Casting, Scope of Variable Identifier, Literal Constants, Symbolic Constants, Formatted Output with printf() Method, Static Variables and Methods, Attribute Final,</p> <p>Introduction to Operators, Precedence and Associativity of Operators, Assignment Operator (=), Basic Arithmetic Operators, Increment (++) and Decrement (- -) Operators, Ternary Operator, Relational Operators, Boolean Logical Operators, Bitwise Logical Operators.</p> <p>Control Statements: Introduction, if Expression, Nested if Expressions, if–else Expressions, Ternary Operator?:, Switch Statement, Iteration Statements, while Expression, do–while Loop, for Loop, Nested for Loop, For–Each for Loop, Break Statement, Continue Statement.</p>								
UNIT-II	Classes and Objects							
<p>Introduction, Class Declaration and Modifiers, Class Members, Declaration of Class Objects, Assigning One Object to Another, Access Control for Class Members, Accessing Private Members of Class, Constructor Methods for Class, Overloaded Constructor Methods, Nested Classes, Final Class and Methods, Passing Arguments by Value and by Reference, Keyword this.</p>								

Methods: Introduction, Defining Methods, Overloaded Methods, Overloaded Constructor Methods, Class Objects as Parameters in Methods, Access Control, Recursive Methods, Nesting of Methods, Overriding Methods, Attributes Final and Static.	
UNIT-III	Arrays
Introduction, Declaration and Initialization of Arrays, Storage of Array in Computer Memory, Accessing Elements of Arrays, Operations on Array Elements, Assigning Array to Another Array, Dynamic Change of Array Size, Sorting of Arrays, Search for Values in Arrays, Class Arrays, Two-dimensional Arrays, Arrays of Varying Lengths, Three-dimensional Arrays, Arrays as Vectors. Inheritance: Introduction, Process of Inheritance, Types of Inheritances, Universal Super Class-Object Class, Inhibiting Inheritance of Class Using Final, Access Control and Inheritance, Multilevel Inheritance, Application of Keyword Super, Constructor Method and Inheritance, Method Overriding, Dynamic Method Dispatch, Abstract Classes, Interfaces and Inheritance. Interfaces: Introduction, Declaration of Interface, Implementation of Interface, Multiple Interfaces, Nested Interfaces, Inheritance of Interfaces, Default Methods in Interfaces, Static Methods in Interface, Functional Interfaces, Annotations.	
UNIT-IV	Packages and Java Library
Introduction, Defining Package, Importing Packages and Classes into Programs, Path and Class Path, Access Control, Packages in Java SE, Java.lang Package and its Classes, Class Object, Enumeration, class Math, Wrapper Classes, Auto-boxing and Auto-unboxing, Java util Classes and Interfaces, Formatter Class, Random Class, Time Package, Class Instant (java.time.Instant), Formatting for Date/Time in Java, Temporal Adjusters Class, Temporal Adjusters Class. Exception Handling: Introduction, Hierarchy of Standard Exception Classes, Keywords throws and throw, try, catch, and finally Blocks, Multiple Catch Clauses, Class Throwable, Unchecked Exceptions, Checked Exceptions. Java I/O and File: Java I/O API, standard I/O streams, types, Byte streams, Character streams, Scanner class, Files in Java(Text Book 2)	
UNIT-V	String Handling in Java
Introduction, Interface Char Sequence, Class String, Methods for Extracting Characters from Strings, Comparison, Modifying, Searching; Class String Buffer. Multithreaded Programming: Introduction, Need for Multiple Threads Multithreaded Programming for Multi-core Processor, Thread Class, Main Thread-Creation of New Threads, Thread States, Thread Priority-Synchronization, Deadlock and Race Situations, Inter-thread Communication - Suspending, Resuming, and Stopping of Threads. Java Database Connectivity: Introduction, JDBC Architecture, Installing MySQL and MySQL Connector/J, JDBC Environment Setup, Establishing JDBC Database Connections, ResultSet Interface Java FX GUI: Java FX Scene Builder, Java FX App Window Structure, displaying text and image, event handling, laying out nodes in scene graph, mouse events (Text Book 3)	
<p>Text Books:</p> <ol style="list-style-type: none"> JAVA one step ahead, Anitha Seth, B.L.Juneja, Oxford. Joy with JAVA, Fundamentals of Object Oriented Programming, DebasisSamanta, MonalisaSarma, Cambridge, 2023. JAVA 9 for Programmers, Paul Deitel, Harvey Deitel, 4th Edition, Pearson. <p>References Books:</p> <ol style="list-style-type: none"> The complete Reference Java, 11th edition, Herbert Schildt, TMH Introduction to Java programming, 7th Edition, Y Daniel Liang, Pearson <p>Online Resources:</p> <ol style="list-style-type: none"> https://nptel.ac.in/courses/106/105/106105191/ https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012880464547618816347_shared/overview 	

Advanced Data Structures & Algorithm Analysis Lab

I Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23CS3P C03	Professional Core	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		0	0	3	1.5	30	70	100
Sessional Exam Duration : 1Hr 50 Min					End Exam Duration: 3 Hrs			
PART-A: BASIC CIVIL ENGINEERING								
Course Objectives:								
The objectives of the course is to								
<ul style="list-style-type: none"> acquire practical skills in constructing and managing Data structures apply the popular algorithm design methods in problem-solving scenarios 								
Course Outcomes: A student after completion of the course will be able to								
CO1	Design and develop programs to solve real world problems with the popular algorithm design methods. (L5)							
CO2	Demonstrate an understanding of Non-Linear data structures by developing implementing the operations on AVL Trees, B-Trees, Heaps and Graphs.(L2)							
CO3	Critically assess the design choices and implementation strategies of algorithms and data structures in complex applications. (L5)							
CO4	Utilize appropriate data structures and algorithms to optimize solutions for specific computational problems.(L3)							
CO5	Compare the performance of different of algorithm design strategies (L4)							
List of topics								
<ul style="list-style-type: none"> Operations on AVL trees, B-Trees, Heap Trees Graph Traversals Sorting techniques Minimum cost spanning trees Shortest path algorithms 0/1 Knapsack Problem Travelling Salesperson problem Optimal Binary Search Trees N-Queens Problem Job Sequencing 								
Sample Programs								
<ol style="list-style-type: none"> 1. Construct an AVL tree for a given set of elements which are stored in a file. And implement insert and delete operation on the constructed tree. Write contents of tree into a new file using in-order. 2. Construct B-Tree an order of 5 with a set of 100 random elements stored in array. Implement searching, insertion and deletion operations. 3. Construct Min and Max Heap using arrays, delete any element and display the content of the Heap. 4. Implement BFT and DFT for given graph, when graph is represented by <ol style="list-style-type: none"> a) Adjacency Matrix b) Adjacency Lists 5. Write a program for finding the bi-connected components in a given graph. 								

6. Implement Quick sort and Merge sort and observe the execution time for various input sizes (Average, Worst and Best cases).
7. Compare the performance of Single Source Shortest Paths using Greedy method when the graph is represented by adjacency matrix and adjacency lists.
8. Implement Job sequencing with deadlines using Greedy strategy.
9. Write a program to solve 0/1 Knapsack problem Using Dynamic Programming.
10. Implement N-Queens Problem Using Backtracking.
11. Use Backtracking strategy to solve 0/1 Knapsack problem.
12. Implement Travelling Sales Person problem using Branch and Bound approach.

Reference Books:

1. Fundamentals of Data Structures in C++, Horowitz Ellis, SahniSartaj, Mehta, Dinesh, 2ndEdition, Universities Press
2. Computer Algorithms/C++ Ellis Horowitz, SartajSahni, SanguthevarRajasekaran, 2ndEdition, University Press
3. Data Structures and program design in C, Robert Kruse, Pearson Education Asia
4. An introduction to Data Structures with applications, Trembley& Sorenson, McGraw Hill

Online Learning Resources:

1. <http://cse01-iiith.vlabs.ac.in/>
2. <http://peterindia.net/Algorithms.html>

Object-Oriented Programming through JAVA Lab

I Semester :						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23CS3P C04	Professional Core	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		0	0	3	1.5	30	70	100
Sessional Exam Duration : 1Hr 50 Min					End Exam Duration: 3 Hrs			

Course Objectives:

The aim of this course is to

- Practice object oriented programming in the Java programming language
- implement Classes, Objects, Methods, Inheritance, Exception, Runtime Polymorphism, User defined Exception handling mechanism
- Illustrate inheritance, Exception handling mechanism, JDBC connectivity
- Construct Threads, Event Handling, implement packages, Java FX GUI

Course Outcomes: A student after completion of the course will be able to

CO1	Demonstrate a solid understanding of Java syntax, including data types, control structures, methods, classes, objects, inheritance, polymorphism, and exception handling. (L2)
CO2	Apply fundamental OOP principles such as encapsulation, inheritance, polymorphism, and abstraction to solve programming problems effectively. (L3)
CO3	Familiar with commonly used Java libraries and APIs, including the Collections Framework, Java I/O, JDBC, and other utility classes. (L2)
CO4	Develop problem-solving skills and algorithmic thinking, applying OOP concepts to design efficient solutions to various programming challenges. (L3)
CO5	Proficiently construct graphical user interface (GUI) applications using JavaFX (L4)

List of topics

- Object Oriented Programming fundamentals- data types, control structures
- Classes, methods, objects, Inheritance, polymorphism,
- Exception handling, Threads, Packages, Interfaces
- Files, I/O streams, JavaFX GUI

Sample Programs

Exercise – 1:

- a) Write a JAVA program to display default value of all primitive data type of JAVA
- b) Write a java program that display the roots of a quadratic equation $ax^2+bx=0$. Calculate the discriminate D and basing on value of D, describe the nature of root.

Exercise - 2

- a) Write a JAVA program to search for an element in a given list of elements using binary search mechanism.
- b) Write a JAVA program to sort for an element in a given list of elements using bubble sort
- c) Write a JAVA program using StringBuffer to delete, remove character.

Exercise - 3

- a) Write a JAVA program to implement class mechanism. Create a class, methods and invoke them inside main method.
- b) Write a JAVA program implement method overloading.

- c) Write a JAVA program to implement constructor.
- d) Write a JAVA program to implement constructor overloading.

Exercise - 4

- a) Write a JAVA program to implement Single Inheritance
- b) Write a JAVA program to implement multi level Inheritance
- c) Write a JAVA program for abstract class to find areas of different shapes

Exercise - 5

- a) Write a JAVA program give example for “super” keyword.
- b) Write a JAVA program to implement Interface. What kind of Inheritance can be achieved?
- c) Write a JAVA program that implements Runtime polymorphism

Exercise - 6

- a) Write a JAVA program that describes exception handling mechanism
- b) Write a JAVA program Illustrating Multiple catch clauses
 - Write a JAVA program for creation of Java Built-in Exceptions
 - Write a JAVA program for creation of User Defined Exception

Exercise - 7

- a) Write a JAVA program that creates threads by extending Thread class. First thread display “Good Morning “every 1 sec, the second thread displays “Hello “every 2 seconds and the third display “Welcome” every 3 seconds, (Repeat the same by implementing Runnable)
- b) Write a program illustrating **is Alive** and **join ()**
- c) Write a Program illustrating Daemon Threads.
- d) Write a JAVA program Producer Consumer Problem

Exercise – 8

- 8. Write a JAVA program that import and use the user defined packages
- 9. Without writing any code, build a GUI that display text in label and image in an ImageView (use JavaFX)
- 10. Build a Tip Calculator app using several JavaFX components and learn how to respond to user interactions with the GUI

Exercise – 9

- 4. Write a java program that connects to a database using JDBC
- b) Write a java program to connect to a database using JDBC and insert values into it.
- c) Write a java program to connect to a database using JDBC and delete values from it

Textbooks:

1. JAVA one step ahead, Anitha Seth, B.L.Juneja, Oxford.
2. Joy with JAVA, Fundamentals of Object Oriented Programming, DebasisSamanta, MonalisaSarma, Cambridge, 2023.
3. JAVA 9 for Programmers, Paul Deitel, Harvey Deitel, 4th Edition, Pearson.

References Books:

1. The complete Reference Java, 11th edition, Herbert Schildt, TMH
2. Introduction to Java programming, 7th Edition, Y Daniel Liang, Pearson

Online Resources:

1. <https://nptel.ac.in/courses/106/105/106105191/>
2. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012880464547618816347_shared/overview

Python Programming

I Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23CS3S C01	Skill Enhancement Course	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		0	1	2	2	30	70	100
						End Exam Duration: 3 Hrs		
<p>Course Objectives: The main objectives of the course are to</p> <ul style="list-style-type: none"> Introduce core programming concepts of Python programming language. Demonstrate about Python data structures like Lists, Tuples, Sets and dictionaries Implement Functions, Modules and Regular Expressions in Python Programming and to create practical and contemporary applications using these 								
<p>Course Outcomes: A student after completion of the course will be able to</p>								
CO1	Show case a dept command of Python syntax, deftly utilizing variables, data types, control structures, functions, modules, and exception handling to engineer robust and efficient code solutions. (L4)							
CO2	apply Python programming concepts to solve a variety of computational problems (L3)							
CO3	understand the principles of object-oriented programming (OOP) in Python, including classes, objects, inheritance, polymorphism, and encapsulation, and apply them to design and implement Python programs (L3)							
CO4	become proficient in using commonly used Python libraries and frameworks such as JSON, XML, NumPy, pandas (L2)							
CO5	exhibit competence in implementing and manipulating fundamental data structures such as lists, tuples, sets, dictionaries (L3)							
<p>List of Topics:</p> <p>UNTI-I: History of Python Programming Language, Thrust Areas of Python, Installing Anaconda Python Distribution, Installing and Using Jupyter Notebook.</p> <p>Parts of Python Programming Language: Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input, Print Output, Type Conversions, the type () Function and Is Operator, Dynamic and Strongly Typed Language.</p> <p>Control Flow Statements: if statement, if-else statement, if...elif...else, Nested if statement, while Loop, for Loop, continue and break Statements, Catching Exceptions Using try and except Statement.</p> <p>Sample Experiments:</p> <ol style="list-style-type: none"> 1. Write a program to find the largest element among three Numbers. 2. Write a Program to display all prime numbers within an interval 3. Write a program to swap two numbers without using a temporary variable. 4. Demonstrate the following Operators in Python with suitable examples. <ol style="list-style-type: none"> i) Arithmetic Operators ii) Relational Operators iii) Assignment Operatorsiv) Logical Operators v) Bit wise Operators vi) Ternary Operator vii) Membership Operators viii) 								

Identity Operators

5. Write a program to add and multiply complex numbers
6. Write a program to print multiplication table of a given number.

UNIT-II: Functions: Built-In Functions, Commonly Used Modules, Function Definition and Calling the function, return Statement and void Function, Scope and Lifetime of Variables, Default Parameters, Keyword Arguments, *args and **kwargs, Command Line Arguments.

Strings: Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings.

Lists: Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods, del Statement.

Sample Experiments:

7. Write a program to define a function with multiple return values.
8. Write a program to define a function using default arguments.
9. Write a program to find the length of the string without using any library functions.
10. Write a program to check if the substring is present in a given string or not.
11. Write a program to perform the given operations on a list:
 - i. Addition
 - ii. Insertion
 - iii. slicing
12. Write a program to perform any 5 built-in functions by taking any list.

UNIT-III: Dictionaries: Creating Dictionary, Accessing and Modifying key:value Pairs in Dictionaries, Built-In Functions Used on Dictionaries, Dictionary Methods, del Statement.

Tuples and Sets: Creating Tuples, Basic Tuple Operations, tuple() Function, Indexing and Slicing in Tuples, Built-In Functions Used on Tuples, Relation between Tuples and Lists, Relation between Tuples and Dictionaries, Using zip() Function, Sets, Set Methods, Frozenset.

Sample Experiments:

13. Write a program to create tuples (name, age, address, college) for at least two members and concatenate the tuples and print the concatenated tuples.
14. Write a program to count the number of vowels in a string (No control flow allowed).
15. Write a program to check if a given key exists in a dictionary or not.
16. Write a program to add a new key-value pair to an existing dictionary.
17. Write a program to sum all the items in a given dictionary.

UNIT-IV:Files: Types of Files, Creating and Reading Text Data, File Methods to Read and Write Data, Reading and Writing Binary Files, Pickle Module, Reading and Writing CSV Files, Python os and os.path Modules.

Object-Oriented Programming: Classes and Objects, Creating Classes in Python, Creating Objects in Python, Constructor Method, Classes with Multiple Objects, Class Attributes Vs Data Attributes, Encapsulation, Inheritance, Polymorphism.

Sample Experiments:

18. Write a program to sort words in a file and put them in another file. The output file should have only lower-case words, so any upper-case words from source must be lowered.
19. Python program to print each line of a file in reverse order.
20. Python program to compute the number of characters, words and lines in a file.

21. Write a program to create, display, append, insert and reverse the order of the items in the array.
22. Write a program to add, transpose and multiply two matrices.
23. Write a Python program to create a class that represents a shape. Include methods to calculate its area and perimeter. Implement subclasses for different shapes like circle, triangle, and square.

UNIT-V: Introduction to Data Science: Functional Programming, JSON and XML in Python, NumPy with Python, Pandas.

Sample Experiments:

24. Python program to check whether a JSON string contains complex object or not.
25. Python Program to demonstrate NumPy arrays creation using array () function.
26. Python program to demonstrate use of ndim, shape, size, dtype.
27. Python program to demonstrate basic slicing, integer and Boolean indexing.
28. Python program to find min, max, sum, cumulative sum of array
29. Create a dictionary with at least five keys and each key represent value as a list where this list contains at least ten values and convert this dictionary as a pandas data frame and explore the data through the data frame as follows:
 - a) Apply head () function to the pandas data frame
 - b) Perform various data selection operations on Data Frame
30. Select any two columns from the above data frame, and observe the change in one attribute with respect to other attribute with scatter and plot operations in matplotlib

Reference Books:

1. Gowrishankar S, Veena A., Introduction to Python Programming, CRC Press.
2. Python Programming, S Sridhar, J Indumathi, V M Hariharan, 2nd Edition, Pearson, 2024
3. Introduction to Programming Using Python, Y. Daniel Liang, Pearson.

Online Learning Resources/Virtual Labs:

1. <https://www.coursera.org/learn/python-for-applied-data-science-ai>
2. <https://www.coursera.org/learn/python?specialization=python#syllabus>

ENVIRONMENTAL SCIENCE

I Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23HS3 MC01	Audit Course	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		2	0	0	-	30	70	100
Sessional Exam Duration :1Hr 50 Min					End Exam Duration: 3 Hrs			
Course Objectives: The learning objectives of this course are to: <ul style="list-style-type: none"> To make the students to get awareness on environment. To understand the importance of protecting natural resources, ecosystems for future generations and pollution causes due to the day to day activities of human life To save earth from the inventions by the engineers. 								
Course Outcomes: A student after completion of the course will be able to								
CO1	Grasp multidisciplinary nature of environmental studies and various renewable and nonrenewable resources.							
CO2	Understand flow and bio-geo- chemical cycles and ecological pyramids.							
CO3	Understand various causes of pollution and solid waste management and related preventive measures.							
CO4	About the rainwater harvesting, watershed management, ozone layer depletion and waste land reclamation.							
CO5	Causes of population explosion, value education and welfare programmes.							
UNIT-I								
Multidisciplinary Nature of Environmental Studies: – Definition, Scope and Importance – Need for Public Awareness. Natural Resources : Renewable and non-renewable resources – Natural resources and associated problems – Forest resources – Use and over – exploitation, deforestation, case studies – Timber extraction – Mining, dams and other effects on forest and tribal people – Water resources – Use and over utilization of surface and ground water – Floods, drought, conflicts over water, dams – benefits and problems – Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies – Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies. – Energy resources:								
UNIT-II								
Ecosystems: Concept of an ecosystem. – Structure and function of an ecosystem – Producers, consumers and decomposers – Energy flow in the ecosystem – Ecological succession – Food chains, food webs and ecological pyramids – Introduction, types, characteristic features, structure and function of the following ecosystem: <ol style="list-style-type: none"> Forest ecosystem. Grassland ecosystem Desert ecosystem. Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries) 								
Biodiversity and its Conservation : Introduction 0 Definition: genetic, species and ecosystem diversity								

– Bio-geographical classification of India – Value of biodiversity: consumptive use, Productive use, social, ethical, aesthetic and option values – Biodiversity at global, National and local levels – India as a mega-diversity nation – Hot-spots of biodiversity – Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts – Endangered and endemic species of India – Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

UNIT-III

Environmental Pollution: Definition, Cause, effects and control measures of :

- a. Air Pollution.
- b. Water pollution
- c. Soil pollution
- d. Marine pollution
- e. Noise pollution
- f. Thermal pollution
- g. Nuclear hazards

Solid Waste Management: Causes, effects and control measures of urban and industrial wastes – Role of an individual in prevention of pollution – Pollution case studies – Disaster management: floods, earthquake, cyclone and landslides.

UNIT-IV

Social Issues and the Environment: From Unsustainable to Sustainable development – Urban problems related to energy – Water conservation, rain water harvesting, watershed management – Resettlement and rehabilitation of people; its problems and concerns. Case studies – Environmental ethics: Issues and possible solutions – Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Case Studies – Wasteland reclamation. – Consumerism and waste products. – Environment Protection Act. – Air (Prevention and Control of Pollution) Act. – Water (Prevention and control of Pollution) Act – Wildlife Protection Act – Forest Conservation Act – Issues involved in enforcement of environmental legislation – Public awareness.

UNIT-V

Human Population and the Environment: Population growth, variation among nations. Population explosion – Family Welfare Programmes. – Environment and human health – Human Rights – Value Education – HIV/AIDS – Women and Child Welfare – Role of information Technology in Environment and human health – Case studies.

Field Work: Visit to a local area to document environmental assets River/forest grassland/hill/mountain – Visit to a local polluted site-Urban/Rural/Industrial/Agricultural Study of common plants, insects, and birds – river, hill slopes, etc..

Textbooks:

1. Textbook of Environmental Studies for Undergraduate Courses Erach Bharucha for University Grants Commission, Universities Press.
2. Palaniswamy, “Environmental Studies”, Pearson education
3. S.Azeem Unnisa, “Environmental Studies” Academic Publishing Company
4. K.Raghavan Nambiar, “Text book of Environmental Studies for Undergraduate Courses as per UGC model syllabus”, Scitech Publications (India), Pvt. Ltd.

References:

1. Deeksha Dave and E.Sai Baba Reddy, “Textbook of Environmental Science”, Cengage Publications.
2. M.Anji Reddy, “Text book of Environmental Sciences and Technology”, BS Publication.
3. J.P.Sharma, Comprehensive Environmental studies, Laxmi publications.
4. J. Glynn Henry and Gary W. Heinke, “Environmental Sciences and Engineering”, Prentice hall of India Private limited

5. G.R.Chatwal, “A Text Book of Environmental Studies” Himalaya Publishing House
Gilbert M. Masters and Wendell P. Ela, “Introduction to Environmental Engineering and Science, Prentice hall of India Private limited.

MANAGERIAL ECONOMICS AND FINANCIAL ANALYSIS

II Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23BA4 HM01A	Management Elective- I	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		2	0	0	2	30	70	100
Sessional Exam Duration :1Hr 50 Min					End Exam Duration: 3 Hrs			
Course Objectives:								
The main objectives of the course is to make student								
<ul style="list-style-type: none"> • To inculcate the basic knowledge of microeconomics and financial accounting • To make the students learn how demand is estimated for different products, input-output relationship for optimizing production and cost • To Know the Various types of market structure and pricing methods and strategy • To give an overview on investment appraisal methods to promote the students to learn how to plan long-term investment decisions. • To provide fundamental skills on accounting and to explain the process of preparing financial statements. 								
Course Outcomes: A student after completion of the course will be able to								
CO1	Define the concepts related to Managerial Economics, financial accounting and management(L2)							
CO2	Understand the fundamentals of Economics viz., Demand, Production, cost, revenue and markets (L2)							
CO3	Apply the Concept of Production cost and revenues for effective Business decision (L3)							
CO4	Analyze how to invest their capital and maximize returns (L4)							
CO5	Evaluate the capital budgeting techniques. (L5)							
CO6	Develop the accounting statements and evaluate the financial performance of business entity (L5)							
UNIT-I	Managerial Economics							
Introduction – Nature, meaning, significance, functions, and advantages. Demand-Concept, Function, Law of Demand - Demand Elasticity- Types – Measurement. Demand Forecasting- Factors governing Forecasting, Methods. Managerial Economics and Financial Accounting and Management.								
UNIT-II	Production and Cost Analysis							
Introduction – Nature, meaning, significance, functions and advantages. Production Function– Least- cost combination– Short run and long run Production Function- Isoquants and Is costs, Cost & Break-Even Analysis - Cost concepts and Cost behaviour- Break-Even Analysis (BEA) - Determination of Break-Even Point (Simple Problems).								
UNIT-III	Business Organizations and Markets							
Introduction – Forms of Business Organizations- Sole Proprietary - Partnership - Joint Stock Companies - Public Sector Enterprises. Types of Markets - Perfect and Imperfect Competition - Features of Perfect Competition Monopoly- Monopolistic Competition–Oligopoly-Price-Output								

Determination - Pricing Methods and Strategies	
UNIT-IV	Capital Budgeting
Introduction – Nature, meaning, significance. Types of Working Capital, Components, Sources of Short-term and Long-term Capital, Estimating Working capital requirements. Capital Budgeting– Features, Proposals, Methods and Evaluation. Projects – Pay Back Method, Accounting Rate of Return (ARR) Net Present Value (NPV) Internal Rate Return (IRR) Method (sample problems)	
UNIT-V	Financial Accounting and Analysis
Introduction – Concepts and Conventions- Double-Entry Bookkeeping, Journal, Ledger, Trial Balance- Final Accounts (Trading Account, Profit and Loss Account and Balance Sheet with simple adjustments). Introduction to Financial Analysis - Analysis and Interpretation of Liquidity Ratios, Activity Ratios, and Capital structure Ratios and Profitability.	
Textbooks:	
<ol style="list-style-type: none"> 1. Varshney & Maheswari: Managerial Economics, Sultan Chand. 2. Aryasri: Business Economics and Financial Analysis, 4/e, MGH. 	
Reference Books:	
<ol style="list-style-type: none"> 1. Ahuja HI Managerial economics Schand. 2. S.A. Siddiqui and A.S. Siddiqui: Managerial Economics and Financial Analysis, New Age International. 3. Joseph G. Nellis and David Parker: Principles of Business Economics, Pearson, 2/e, New Delhi. 4. Domnick Salvatore: Managerial Economics in a Global Economy, Cengage. 	
Online Learning Resources:	
https://www.slideshare.net/123ps/managerial-economics-ppt https://www.slideshare.net/rossanz/production-and-cost-45827016 https://www.slideshare.net/darkyla/business-organizations-19917607 https://www.slideshare.net/balarajbl/market-and-classification-of-market https://www.slideshare.net/ruchi101/capital-budgeting-ppt-59565396 https://www.slideshare.net/ashu1983/financial-accounting	

PROBABILITY & STATISTICS
(Common to CSE, CSE (AI &ML), CSE(IoT), CSE(AI), AI &ML, CS, IT)

II Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23MA4 BS01	Basic Science	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		3	0	0	3	30	70	100
Sessional Exam Duration :1Hr 50 Min					End Exam Duration: 3 Hrs			
Course Objectives: The main objectives of the course is to make student								
<ul style="list-style-type: none"> • 								
Course Outcomes: A student after completion of the course will be able to								
CO1	Acquire knowledge in finding the analysis of the data quantitatively or categorically and various statistical elementary tools. L2, L3							
CO2	Develop skills in designing mathematical models involving probability, random variables and the critical thinking in the theory of probability and its applications in real life problems. L3, L5							
CO3	Apply the theoretical probability distributions like binomial, Poisson, and Normal in the relevant application areas. L3							
CO4	Analyze to test various hypotheses included in theory and types of errors for large samples. L2, L3							
CO5	Apply the different testing tools like t-test, F-test, chi-square test to analyze the relevant real life problems. L3, L5							
UNIT-I	Descriptive statistics							
Statistics Introduction, Population vs Sample, Collection of data, primary and secondary data, Measures of Central tendency, Measures of Variability (spread or variance) Skewness, Kurtosis, correlation, correlation coefficient, rank correlation, regression coefficients, method of least squares, regression lines.								
UNIT-II	Probability							
Probability, probability axioms, addition law and multiplicative law of probability, conditional probability, Baye's theorem, random variables (discrete and continuous), probability density functions, properties, mathematical expectation.								
UNIT-III	Probability distributions							
Probability distributions: Binomial, Poisson and Normal-their properties (Chebyshevs inequality). Approximation of the binomial distribution to normal distribution.								
UNIT-IV	Estimation and Testing of hypothesis, large sample tests							
Estimation-parameters, statistics, sampling distribution, point estimation, Formulation of null hypothesis, alternative hypothesis, the critical and acceptance regions, level of significance, two types of errors and power of the test. Large Sample Tests: Test for single proportion, difference of								

proportions, test for single mean and difference of means. Confidence interval for parameters in one sample and two sample problems

UNIT-V | Small sample tests

Student t-distribution (test for single mean, two means and paired t-test), testing of equality of variances (F-test), χ^2 - test for goodness of fit, χ^2 - test for independence of attributes.

Textbooks:

1. Miller and Friends, Probability and Statistics for Engineers, 7/e, Pearson, 2008.
2. S.C. Gupta and V.K. Kapoor, Fundamentals of Mathematical Statistics, 11/e, Sultan Chand & Sons Publications, 2012.

Reference Books:

1. S. Ross, a First Course in Probability, Pearson Education India, 2002.
2. W. Feller, an Introduction to Probability Theory and its Applications, 1/e, Wiley, 1968.
3. B. V. Ramana, Higher Engineering Mathematics, Mc Graw Hill Education.

Online Learning Resources:

1. https://onlinecourses.nptel.ac.in/noc21_ma74/preview
2. https://onlinecourses.nptel.ac.in/noc22_mg31/preview

Operating Systems

II Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23CS4P C01	Profession al Core	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		3	0	0	3	30	70	100
Sessional Exam Duration : 1Hr 50 Min					End Exam Duration: 3 Hrs			
Course Objectives: The main objectives of the course is to make student <ul style="list-style-type: none"> Understand the basic concepts and principles of operating systems, including process management, memory management, file systems, and Protection Make use of process scheduling algorithms and synchronization techniques to achieve better performance of a computer system. Illustrate different conditions for deadlock and their possible solutions. 								
Course Outcomes: A student after completion of the course will be able to								
CO1	Describe the basics of the operating systems, mechanisms of OS to handle processes, threads, and their communication. (L1)							
CO2	Understand the basic concepts and principles of operating systems, including process management, memory management, file systems, and Protection. (L2)							
CO3	Make use of process scheduling algorithms and synchronization techniques to achieve better performance of a computer system. (L3)							
CO4	Illustrate different conditions for deadlock and their possible solutions. (L2)							
CO5	Analyze the memory management and its allocation policies. (L4)							
UNIT-I	Operating Systems Overview: Introduction, Operating system functions, Operating systems operations, Computing environments, Free and Open-Source Operating Systems System Structures: Operating System Services, User and Operating-System Interface, system calls, Types of System Calls, system programs, Operating system Design and Implementation, Operating system structure, Building and Booting an Operating System, Operating system debugging							
UNIT-II	Processes: Process Concept, Process scheduling, Operations on processes, Inter-process communication. Threads and Concurrency: Multithreading models, Thread libraries, Threading issues. CPU Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms, Multiple processor scheduling.							
UNIT-III	Synchronization Tools: The Critical Section Problem, Peterson's Solution, Mutex Locks, Semaphores, Monitors, Classic problems of Synchronization.							

	Deadlocks: system Model, Deadlock characterization, Methods for handling Deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from Deadlock.
UNIT-IV	
	Memory-Management Strategies: Introduction, Contiguous memory allocation, Paging, Structure of the Page Table, Swapping. Virtual Memory Management: Introduction, Demand paging, Copy-on-write, Page replacement, Allocation of frames, Thrashing Storage Management: Overview of Mass Storage Structure, HDD Scheduling.
UNIT-V	
	File System: File System Interface: File concept, Access methods, Directory Structure; File system Implementation: File-system structure, File-system Operations, Directory implementation, Allocation method, Free space management; File-System Internals: File-System Mounting, Partitions and Mounting, File Sharing. Protection: Goals of protection, Principles of protection, Protection Rings, Domain of protection, Access matrix.
	<p>TextBooks:</p> <ol style="list-style-type: none"> 1. Operating System Concepts, Silberschatz A, Galvin P B, Gagne G, 10th Edition, Wiley, 2018. 2. Modern Operating Systems, Tanenbaum A S, 4th Edition, Pearson , 2016 <p>Reference Books:</p> <ol style="list-style-type: none"> 1. Operating Systems -Internals and Design Principles, Stallings W, 9th edition, Pearson, 2018 2. Operating Systems: A Concept Based Approach, D.M Dhamdhare, 3rd Edition, McGraw- Hill, 2013 <p>Online Learning Resources:</p> <ol style="list-style-type: none"> 1. https://nptel.ac.in/courses/106/106/106106144/ 2. http://peterindia.net/OperatingSystems.html

Database Management Systems

II Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23CS4P C02	Professional Core	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		3	0	0	3	30	70	100
Sessional Exam Duration : 1Hr 50 Min					End Exam Duration: 3 Hrs			
Course Objectives: The main objectives of the course is to <ul style="list-style-type: none"> • Introduce database management systems and to give a good formal foundation on the relational model of data and usage of Relational Algebra • Introduce the concepts of basic SQL as a universal Database language • Demonstrate the principles behind systematic database design approaches by covering conceptual design, logical design through normalization • Provide an overview of physical design of a database system, by discussing Database indexing techniques and storage techniques. 								
Course Outcomes: A student after completion of the course will be able to								
CO1	Understand the basic concepts of database management systems (L2)							
CO2	Analyze a given database application scenario to use ER model for conceptual design of the database (L4)							
CO3	Utilize SQL proficiently to address diverse query challenges (L3).							
CO4	Employ normalization methods to enhance database structure (L3)							
CO5	Assess and implement transaction processing, concurrency control and database recovery protocols in databases.(L4)							
UNIT-I	Introduction							
Database system, Characteristics (Database Vs File System), Database Users, Advantages of Database systems, Database applications. Brief introduction of different Data Models; Concepts of Schema, Instance and data independence; Three tier schema architecture for data independence; Database system structure, environment, Centralized and Client Server architecture for the database. Entity Relationship Model: Introduction, Representation of entities, attributes, entity set, relationship, relationship set, constraints, sub classes, super class, inheritance, specialization, generalization using ER Diagrams.								
UNIT-II	Relational Model							
Introduction to relational model, concepts of domain, attribute, tuple, relation, importance of null values, constraints (Domain, Key constraints, integrity constraints) and their importance, Relational Algebra, Relational Calculus. BASIC SQL: Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update).								
UNIT-III	SQL							
Basic SQL querying (select and project) using where clause, arithmetic & logical operations, SQL functions(Date and Time, Numeric, String conversion).Creating tables with relationship, implementation of key and integrity constraints, nested queries, sub queries, grouping, aggregation, ordering, implementation of different types of joins, view(updatable and non-updatable), relational set operations.								

UNIT-IV	Schema Refinement (Normalization)
Purpose of Normalization or schema refinement, concept of functional dependency, normal forms based on functional dependency Lossless join and dependency preserving decomposition, (1NF, 2NF and 3 NF), concept of surrogate key, Boyce-Codd normal form(BCNF), MVD, Fourth normal form(4NF), Fifth Normal Form (5NF).	
UNIT-V	Transaction Concept
Transaction State, ACID properties, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability, lock based, time stamp based, optimistic, concurrency protocols, Deadlocks, Failure Classification, Storage, Recovery and Atomicity, Recovery algorithm.	
Introduction to Indexing Techniques: B+ Trees, operations on B+Trees, Hash Based Indexing:	
Text Books:	
<ol style="list-style-type: none"> 1. Database Management Systems, 3rd edition, Raghurama Krishnan, Johannes Gehrke, TMH (For Chapters 2, 3, 4) 2. Database System Concepts, 5th edition, Silberschatz, Korth, Sudarsan, TMH (For Chapter 1 and Chapter 5) 	
Reference Books:	
<ol style="list-style-type: none"> 1. Introduction to Database Systems, 8th edition, C J Date, Pearson. 2. Database Management System, 6th edition, Ramez Elmasri, Shamkant B. Navathe, Pearson 3. Database Principles Fundamentals of Design Implementation and Management, Corlos Coronel, Steven Morris, Peter Robb, Cengage Learning. 	
Web-Resources:	
<ol style="list-style-type: none"> 1. https://nptel.ac.in/courses/106/105/106105175/ 2. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01275806667282022456_shared/overview 	

Software Engineering

II Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23CS4P C03	Professional Core	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		1	1	0	3	30	70	100
Sessional Exam Duration :1Hr 50 Min					End Exam Duration: 3 Hrs			
Course Objectives: The objectives of this course are to introduce <ul style="list-style-type: none"> • Software life cycle models, Software requirements and SRS document. • Project Planning, quality control and ensuring good quality software. • Software Testing strategies, use of CASE tools, Implementation issues, validation & verification procedures. 								
Course Outcomes: A student after completion of the course will be able to								
CO1	Perform various life cycle activities like Analysis, Design, Implementation, Testing and Maintenance (L3)							
CO2	Analyse various software engineering models and apply methods for design and development of software projects. (L4)							
CO3	Develop system designs using appropriate techniques. (L3)							
CO4	Understand various testing techniques for a software project. (L2)							
CO5	Apply standards, CASE tools and techniques for engineering software projects (L3)							
UNIT-I								
Introduction: Evolution, Software development projects, Exploratory style of software developments, Emergence of software engineering, Notable changes in software development practices, Computer system engineering. Software Life Cycle Models: Basic concepts, Waterfall model and its extensions, Rapid application development, Agile development model, Spiral model.								
UNIT-II								
Software Project Management: Software project management complexities, Responsibilities of a software project manager, Metrics for project size estimation, Project estimation techniques, Empirical Estimation techniques, COCOMO, Halstead's software science, risk management. Requirements Analysis And Specification: Requirements gathering and analysis, Software Requirements Specification (SRS), Formal system specification, Axiomatic specification, Algebraic specification, Executable specification and 4GL.								
UNIT-III								
Software Design: Overview of the design process, How to characterize a good software design? Layered arrangement of modules, Cohesion and Coupling. approaches to software design. Agility: Agility and the Cost of Change, Agile Process, Extreme Programming (XP), Other Agile Process Models, Tool Set for the Agile Process (Text Book 2) Function-Oriented Software Design: Overview of SA/SD methodology, Structured analysis, Developing the DFD model of a system, Structured design, Detailed design, and Design Review. User Interface Design: Characteristics of a good user interface, Basic concepts, Types of user interfaces, Fundamentals of component-based GUI development, and user interface design methodology.								

UNIT-IV	
<p>Coding And Testing: Coding, Code review, Software documentation, Testing, Black-box testing, White-Box testing, Debugging, Program analysis tools, Integration testing, Testing object-oriented programs, Smoke testing, and Some general issues associated with testing.</p> <p>Software Reliability And Quality Management: Software reliability. Statistical testing, Software quality, Software quality management system, ISO 9000. SEI Capability maturity model. Few other important quality standards, and Six Sigma.</p>	
UNIT-V	
<p>Computer-Aided Software Engineering (Case): CASE and its scope, CASE environment, CASE support in the software life cycle, other characteristics of CASE tools, Towards second generation CASE Tool, and Architecture of a CASE Environment.</p> <p>Software Maintenance: Characteristics of software maintenance, Software reverse engineering, Software maintenance process models and Estimation of maintenance cost.</p> <p>Software Reuse: reuse- definition, introduction, reason behind no reuse so far, Basic issues in any reuse program, A reuse approach, and Reuse at organization level.</p>	
<p>Text Books:</p> <ol style="list-style-type: none"> 1. Fundamentals of Software Engineering, Rajib Mall, 5th Edition, PHI. 2. Software Engineering A practitioner's Approach, Roger S. Pressman, 9th Edition, Mc-Graw Hill International Edition. <p>Reference Books:</p> <ol style="list-style-type: none"> 1. Software Engineering, Ian Sommerville, 10th Edition, Pearson. 2. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press. <p>e-Resources:</p> <ol style="list-style-type: none"> 1) https://nptel.ac.in/courses/106/105/106105182/ 2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01260589506387148827_shared/overview <p>https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013382690411003904735_shared/overview</p>	

Operating Systems Lab

II Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23CS4P C04	Professional Core	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		0	0	3	1.5	30	70	100
Sessional Exam Duration : 1Hr 50 Min					End Exam Duration: 3 Hrs			
<p>Course Objectives: The main objectives of the course are to</p> <ul style="list-style-type: none"> Provide insights into system calls, file systems, semaphores, Develop and debug CPU Scheduling algorithms, page replacement algorithms, thread implementation Implement Bankers Algorithms to Avoid the Dead Lock 								
<p>Course Outcomes: A student after completion of the course will be able to</p>								
CO1	Trace different CPU Scheduling algorithms (L2).							
CO2	Implement Bankers Algorithms to Avoid the Dead Lock (L3).							
CO3	Evaluate Page replacement algorithms (L5).							
CO4	Illustrate the file organization techniques (L4).							
CO5	Illustrate Inter process Communication and concurrent execution of threads (L4)							
<p>Experiments covering the Topics:</p> <ul style="list-style-type: none"> UNIX fundamentals, commands & system calls CPU Scheduling algorithms, thread processing IPC, semaphores, monitors, deadlocks Page replacement algorithms, file allocation strategies Memory allocation strategies 								
<p>Sample Experiments:</p> <ol style="list-style-type: none"> 1. Practicing of Basic UNIX Commands. 2. Write programs using the following UNIX operating system calls fork, exec, getpid, exit, wait, close, stat, opendir and readdir 3. Simulate UNIX commands like cp, ls, grep, etc., 4. Simulate the following CPU scheduling algorithms a) FCFS b) SJF c) Priority d) Round Robin 5. Control the number of ports opened by the operating system with a) Semaphore b) Monitors. 6. Write a program to illustrate concurrent execution of threads using pthreads library. 7. Write a program to solve producer-consumer problem using Semaphores. 8. Implement the following memory allocation methods for fixed partition a) First fit b) Worst fit c) Best fit 9. Simulate the following page replacement algorithms a) FIFO b) LRU c) LFU 10. Simulate Paging Technique of memory management. 11. Implement Bankers Algorithm for Dead Lock avoidance and prevention 12. Simulate the following file allocation strategies a) Sequential b) Indexed c) Linked 								

Reference Books:

1. Operating System Concepts, Silberschatz A, Galvin P B, Gagne G, 10th Edition, Wiley, 2018.
2. Modern Operating Systems, Tanenbaum A S, 4th Edition, Pearson, 2016
3. Operating Systems -Internals and Design Principles, Stallings W, 9th edition, Pearson, 2018
4. Operating Systems: A Concept Based Approach, D.M Dhamdhare, 3rd Edition, McGraw- Hill, 2013

Online Learning Resources:

1. <https://www.cse.iitb.ac.in/~mythili/os/>
2. <http://peterindia.net/OperatingSystems.html>

SRIT (SR)

Database Management Systems Lab

II Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23CS4P C05	Professional Core	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		0	0	3	1.5	30	70	100
						End Exam Duration: 3 Hrs		

Course Objectives:

This Course will enable students to

- Populate and query a database using SQL DDL/DML Commands
- Declare and enforce integrity constraints on a database
- Writing Queries using advanced concepts of SQL
- Programming PL/SQL including procedures, functions, cursors and triggers,

Course Outcomes: The students will be able to

CO1	Utilizing Data Definition Language (DDL), Data Manipulation Language (DML), and Data Control Language (DCL) commands effectively within a database environment (L3)
CO2	Constructing and execute queries to manipulate and retrieve data from databases. (L3)
CO3	Develop application programs using PL/SQL. (L3)
CO4	Analyze requirements and design custom Procedures, Functions, Cursors, and Triggers, leveraging their capabilities to automate tasks and optimize database functionality (L4)
CO5	Establish database connectivity through JDBC (Java Database Connectivity) (L3)

List of topics:

- DDL, DML, DCL commands
- Queries, nested queries, built-in functions,
- PL/SQL programming- control structures
- Procedures, Functions, Cursors, Triggers,
- Database connectivity- ODBC/JDBC

List of Experiments:

1. Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.
2. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOT EXISTS, UNION, INTERSET, Constraints. Example:- Select the roll number and name of the student who secured fourth rank in the class.
3. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
4. Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)
5.
 - i. Create a simple PL/SQL program which includes declaration section, executable section and exception –Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)

- ii. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.
6. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.
7. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE-APPLICATION ERROR.
8. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.
9. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
10. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.
11. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers
12. Create a table and perform the search operation on table using indexing and non-indexing techniques.
13. Write a Java program that connects to a database using JDBC
14. Write a Java program to connect to a database using JDBC and insert values into it
15. Write a Java program to connect to a database using JDBC and delete values from it

Text Books/Suggested Reading:

1. Oracle: The Complete Reference by Oracle Press
2. Nilesh Shah, "Database Systems Using Oracle", PHI, 2007
3. Rick F Vander Lans, "Introduction to SQL", Fourth Edition, Pearson Education, 2007

Full Stack Development – 1

II Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23CS4S C01	Skill Enhance ment Course	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		0	1	2	2	30	70	100
						End Exam Duration: 3 Hrs		
Course Objectives:								
The main objectives of the course are to								
<ol style="list-style-type: none"> 1. Make use of HTML elements and their attributes for designing static web pages 2. Build a web page by applying appropriate CSS styles to HTML elements 3. Experiment with JavaScript to develop dynamic web pages and validate forms 								
Course Outcomes: The students will be able to								
CO1	Design Websites. (L6)							
CO2	Apply Styling to web pages. (L3)							
CO3	Make Web pages interactive. (L3)							
CO4	Design Forms for applications. (L6)							
CO5	Choose Control Structure based on the logic to be implemented. (L4)							
Experiments covering the Topics:								
<ul style="list-style-type: none"> • Lists, Links and Images • HTML Tables, Forms and Frames • HTML 5 and Cascading Style Sheets, Types of CSS • Selector forms • CSS with Color, Background, Font, Text and CSS Box Model • Applying JavaScript - internal and external, I/O, Type Conversion • JavaScript Conditional Statements and Loops, Pre-defined and User-defined Objects • JavaScript Functions and Events • Node.js 								
Sample Experiments:								
<ol style="list-style-type: none"> 1. Lists, Links and Images <ol style="list-style-type: none"> a. Write a HTML program, to explain the working of lists. Note: It should have an ordered list, unordered list, nested lists and ordered list in an unordered list and definition lists. b. Write a HTML program, to explain the working of hyperlinks using <a> tag and href, target Attributes. c. Create a HTML document that has your image and your friend's image with a specific height and width. Also when clicked on the images it should navigate to their respective profiles. d. Write a HTML program, in such a way that, rather than placing large images on a page, the preferred technique is to use thumbnails by setting the height and width parameters to something like to 100*100 pixels. Each thumbnail image is also a link to a full sized version of the image. Create an image gallery using this technique 2. HTML Tables, Forms and Frames <ul style="list-style-type: none"> • Write a HTML program, to explain the working of tables. (use tags: <table>, <tr>, <th>, <td> 								

- and attributes: border, rowspan, colspan)
- Write a HTML program, to explain the working of tables by preparing a timetable. (Note: Use <caption> tag to set the caption to the table & also use cell spacing, cell padding, border, rowspan, colspan etc.).
 - Write a HTML program, to explain the working of forms by designing Registration form. (Note: Include text field, password field, number field, date of birth field, checkboxes, radio buttons, list boxes using <select>&<option> tags, <text area> and two buttons ie: submit and reset. Use tables to provide a better view).
 - Write a HTML program, to explain the working of frames, such that page is to be divided into 3 parts on either direction. (Note: first frame □ image, second frame □ paragraph, third frame □ hyperlink. And also make sure of using “no frame” attribute such that frames to be fixed).
3. HTML 5 and Cascading Style Sheets, Types of CSS
- a. Write a HTML program, that makes use of <article>, <aside>, <figure>, <figcaption>, <footer>, <header>, <main>, <nav>, <section>, <div>, tags.
 - b. Write a HTML program, to embed audio and video into HTML web page.
 - c. Write a program to apply different types (or levels of styles or style specification formats) - inline, internal, external styles to HTML elements. (identify selector, property and value).
4. Selector forms
- a. Write a program to apply different types of selector forms
 - Simple selector (element, id, class, group, universal)
 - Combinator selector (descendant, child, adjacent sibling, general sibling)
 - Pseudo-class selector
 - Pseudo-element selector
 - Attribute selector
5. CSS with Color, Background, Font, Text and CSS Box Model
- a. Write a program to demonstrate the various ways you can reference a color in CSS.
 - b. Write a CSS rule that places a background image halfway down the page, tilting it horizontally. The image should remain in place when the user scrolls up or down.
 - c. Write a program using the following terms related to CSS font and text:
 - i. font-size
 - ii. font-weight
 - iii. font-style
 - iv. text-decoration
 - v. text-transformation
 - vi. text-alignment
 - d. Write a program, to explain the importance of CSS Box model using
 - i. Content
 - ii. Border
 - iii. Margin
 - iv. padding
6. Applying JavaScript - internal and external, I/O, Type Conversion
- a. Write a program to embed internal and external JavaScript in a web page.
 - b. Write a program to explain the different ways for displaying output.
 - c. Write a program to explain the different ways for taking input.
 - d. Create a webpage which uses prompt dialogue box to ask a voter for his name and age. Display the information in table format along with either the voter can vote or not
7. JavaScript Pre-defined and User-defined Objects
- a. Write a program using document object properties and methods.
 - b. Write a program using window object properties and methods.
 - c. Write a program using array object properties and methods.

- d. Write a program using math object properties and methods.
- e. Write a program using string object properties and methods.
- f. Write a program using regex object properties and methods.
- g. Write a program using date object properties and methods.
- h. Write a program to explain user-defined object by using properties, methods, accessors, constructors and display.

8. JavaScript Conditional Statements and Loops

- a. Write a program which asks the user to enter three integers, obtains the numbers from the user and outputs HTML text that displays the larger number followed by the words “LARGER NUMBER” in an information message dialog. If the numbers are equal, output HTML text as “EQUAL NUMBERS”.
- b. Write a program to display week days using switch case.
- c. Write a program to print 1 to 10 numbers using for, while and do-while loops.
- d. Write a program to print data in object using for-in, for-each and for-of loops
- e. Develop a program to determine whether a given number is an ‘ARMSTRONG NUMBER’ or not. [Eg: 153 is an Armstrong number, since sum of the cube of the digits is equal to the number i.e., $1^3 + 5^3 + 3^3 = 153$]
- f. Write a program to display the denomination of the amount deposited in the bank in terms of 100’s, 50’s, 20’s, 10’s, 5’s, 2’s & 1’s. (Eg: If deposited amount is Rs.163, the output should be 1-100’s, 1-50’s, 1- 10’s, 1-2’s & 1-1’s)

9. Javascript Functions and Events

- a. Design a appropriate function should be called to display
 - Factorial of that number
 - Fibonacci series up to that number
 - Prime numbers up to that number
 - Is it palindrome or not
- b. Design a HTML having a text box and four buttons named Factorial, Fibonacci, Prime, and Palindrome. When a button is pressed an appropriate function should be called to display

11. Factorial of that number

12. Fibonacci series up to that number

13. Prime numbers up to that number

14. Is it palindrome or not

- c. Write a program to validate the following fields in a registration page
 - i. Name (start with alphabet and followed by alphanumeric and the length should not be less than 6 characters)
 - ii. Mobile (only numbers and length 10 digits)
 - iii. E-mail (should contain format like xxxxxxx@xxxxxx.xxx)

Text Books:

1. Programming the World Wide Web, 7th Edition, Robert W Sebesta, Pearson, 2013.
2. Web Programming with HTML5, CSS and JavaScript, John Dean, Jones & Bartlett Learning, 2019 (Chapters 1-11).
3. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasanth Subramanian, 2nd edition, APress, O’Reilly.

Web Links:

1. <https://www.w3schools.com/html>
2. <https://www.w3schools.com/css>

3. <https://www.w3schools.com/js/>
4. <https://www.w3schools.com/nodejs>
5. <https://www.w3schools.com/typescript>

SRIT (SR)

DESIGN THINKING & INNOVATION

II Semester						R23		
Course Code	Category	Hours/Week			Credits	Maximum Marks		
23BA4 HM03	BS&H	L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
		1	0	2	2	30		30
Sessional Exam Duration :1Hr 50 Min					End Exam Duration: 3 Hrs			
Course Objectives:								
The objective of this course is to familiarize students with design thinking process as a tool for breakthrough innovation. It aims to equip students with design thinking skills and ignite the minds to create innovative ideas, develop solutions for real-time problems.								
Course Outcomes: A student after completion of the course will be able to								
CO1	Define the concepts related to design thinking. (L1, L2)							
CO2	Explain the fundamentals of Design Thinking and innovation (L1, L2)							
CO3	Apply the design thinking techniques for solving problems in various sectors. (L3)							
CO4	Analyze to work in a multidisciplinary environment (L4)							
CO5	Evaluate the value of creativity (L5)							
CO6	Formulate specific problem statements of real time issues (L3, L6)							
UNIT-I	Introduction to Design Thinking							
Introduction to elements and principles of Design, basics of design-dot, line, shape, form as fundamental design components. Principles of design. Introduction to design thinking, history of Design Thinking, New materials in Industry.								
UNIT-II	Design Thinking Process							
Design thinking process (empathize, analyze, idea & prototype), implementing the process in driving inventions, design thinking in social innovations. Tools of design thinking - person, costumer, journey map, brainstorming, product development Activity: Every student presents their idea in three minutes, Every student can present design process in the form of flow diagram or flow chart etc. Every student should explain about product development.								
UNIT-III	Innovation							
Art of innovation, Difference between innovation and creativity, role of creativity and innovation in organizations- Creativity to Innovation- Teams for innovation- Measuring the impact and value of creativity. Activity: Debate on innovation and creativity, Flow and planning from idea to innovation, Debate on value-based innovation.								
UNIT-IV	Product Design							
Problem formation, introduction to product design, Product strategies, Product value, Product planning, product specifications- Innovation towards product design- Case studies Activity: Importance of modeling, how to set specifications, Explaining their own product design.								
UNIT-V	Design Thinking in Business Processes							
Design Thinking applied in Business & Strategic Innovation, Design Thinking principles that redefine business – Business challenges: Growth, Predictability, Change, Maintaining Relevance, Extreme competition, Standardization. Design thinking to meet corporate needs- Design thinking for Startups-								

Defining and testing Business Models and Business Cases- Developing & testing prototypes.

Activity: How to market our own product, About maintenance, Reliability and plan for startup.

Textbooks:

1. Tim Brown, Change by design, Harper Bollins (2009)
2. Idris Mootee, Design Thinking for Strategic Innovation, 2013, John Wiley & Sons.

Reference Books:

1. David Lee, Design Thinking in the Classroom, Ulysses press
2. Shruti N Shetty, Design the Future, Norton Press
3. William Lidwell, Universal Principles of Design- Kritinaholden, Jill Butter.
4. Chesbrough.H, The Era of Open Innovation – 2013

Online Learning Resources:

<https://nptel.ac.in/courses/110/106/110106124/>

<https://nptel.ac.in/courses/109/104/109104109/>

https://swayam.gov.in/nd1_noc19_mg60/preview

COMMUNITY SERVICE PROJECT
Experiential learning through community engagement

II Semester					R23			
Course Code	Category	Hours/Week			Credits	Maximum Marks		
		L/D	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
						30	70	100
					End Exam Duration: 3 Hrs			

Introduction

- Community Service Project is an experiential learning strategy that integrates meaningful community service with instruction, participation, learning and community development.
- Community Service Project involves students in community development and service activities and applies the experience to personal and academic development.
- Community Service Project is meant to link the community with the college for mutual benefit. The community will benefit with the focused contribution of the college students for the village/ local development. The college finds an opportunity to develop social sensibility and responsibility among students and emerge as a socially responsible institution.

Objective

Community Service Project should be an integral part of the curriculum, as an alternative to the 2 months of Summer Internships / Apprenticeships / On the Job Training, whenever there is an exigency when students cannot pursue their summer internships. The specific objectives are;

- To sensitize the students to the living conditions of the people who are around them,
- To help students to realize the stark realities of society.
- To bring about an attitudinal change in the students and help them to develop societal consciousness, sensibility, responsibility and accountability
- To make students aware of their inner strength and help them to find new /out of box solutions to social problems.
- To make students socially responsible citizens who are sensitive to the needs of the disadvantaged sections.
- To help students to initiate developmental activities in the community in coordination with public and government authorities.
- To develop a holistic life perspective among the students by making them study culture, traditions, habits, lifestyles, resource utilization, wastages and its management, social problems, public administration system and the roles and responsibilities of different persons across different social systems.

Implementation of Community Service Project

- Every student should put in 6 weeks for the Community Service Project during the summer vacation.
- Each class/section should be assigned with a mentor.
- Specific Departments could concentrate on their major areas of concern. For example, Dept. of Computer Science can take up activities related to Computer Literacy to different sections of people like - youth, women, housewives, etc
- A logbook must be maintained by each of the students, where the activities

undertaken/involved to be recorded.

- The logbook has to be countersigned by the concerned mentor/faculty in charge.
- An evaluation to be done based on the active participation of the student and grade could be awarded by the mentor/faculty member.
- The final evaluation to be reflected in the grade memo of the student.
- The Community Service Project should be different from the regular programs of NSS/NCC/Green Corps/Red Ribbon Club, etc.
- Minor project reports should be submitted by each student. An internal Viva shall also be conducted by a committee constituted by the principal of the college.
- Award of marks shall be made as per the guidelines of Internship/apprentice/ on the job training.

Procedure

- A group of students or even a single student could be assigned for a particular habitation or village or municipal ward, as far as possible, in the near vicinity of their place of stay, to enable them to commute from their residence and return back by evening or so.
- The Community Service Project is a twofold one –
 - First, the student/s could conduct a survey of the habitation, if necessary, in terms of their own domain or subject area. Or it can even be a general survey, incorporating all the different areas. A common survey format could be designed. This should not be viewed as a duplication of work by the Village or Ward volunteers, rather, it could be another primary source of data.
 - Secondly, the student/s could take up a social activity, concerning their domain or subject area. The different areas, could be like –
 - Agriculture
 - Health
 - Marketing and Cooperation
 - Animal Husbandry
 - Horticulture
 - Fisheries
 - Sericulture
 - Revenue and Survey
 - Natural Disaster Management
 - Irrigation
 - Law & Order
 - Excise and Prohibition
 - Mines and Geology
 - Energy
 - Internet
 - Free Electricity
 - Drinking Water

EXPECTED OUTCOMES

BENEFITS OF COMMUNITY SERVICE PROJECT TO STUDENTS

Learning Outcomes

- Positive impact on students' academic learning
- Improves students' ability to apply what they have learned in "the real world"
- Positive impact on academic outcomes such as demonstrated complexity of understanding, problem analysis, problem-solving, critical thinking, and cognitive development.
- Improved ability to understand complexity and ambiguity

Personal Outcomes

- Greater sense of personal efficacy, personal identity, spiritual growth, and moral development
- Greater interpersonal development, particularly the ability to work well with others, and build leadership and communication skills.

Social Outcomes

- Reduced stereotypes and greater inter-cultural understanding
- Improved social responsibility and citizenship skills
- Greater involvement in community service after graduation

Career Development

- Connections with professionals and community members for learning and career opportunities
- Greater academic learning, leadership skills, and personal efficacy can lead to greater opportunity.

Relationship with the Institution

- Stronger relationships with faculty
- Greater satisfaction with college
- Improved graduation rates

BENEFITS OF COMMUNITY SERVICE PROJECT TO FACULTY MEMBERS

- Satisfaction with the quality of student learning
- New avenues for research and publication via new relationships between faculty and community
- Providing networking opportunities with engaged faculty in other disciplines or institutions
- A stronger commitment to one's research.

BENEFITS OF COMMUNITY SERVICE PROJECT TO COLLEGES AND UNIVERSITIES

- Improved institutional commitment.
- Improved student retention
- Enhanced community relations

BENEFITS OF COMMUNITY SERVICE PROJECT TO COMMUNITY

- Satisfaction with student participation
- Valuable human resources needed to achieve community goals.
- New energy, enthusiasm and perspectives applied to community work.
- Enhanced community-university relations.

SUGGESTIVE LIST OF PROGRAMMES UNDER COMMUNITY SERVICE PROJECT

The following the recommended list of projects for Engineering students. The lists are not exhaustive and open for additions, deletions, and modifications. Colleges are expected to focus on specific local issues for this kind of project. The students are expected to carry out these projects with involvement, commitment, responsibility, and accountability. The mentors of a group of students should take the responsibility of motivating, facilitating, and guiding the students. They have to interact with local leadership and people and appraise the objectives and benefits of this kind of project. The project reports shall be placed in the college website for reference. Systematic, Factual, methodical and honest reporting should be ensured.

For Engineering Students

1. Water facilities and drinking water availability
2. Health and hygiene
3. Stress levels and coping mechanisms
4. Health intervention programmes
5. Horticulture
6. Herbal plants
7. Botanical survey
8. Zoological survey
9. Marine products
10. Aqua culture
11. Inland fisheries
12. Animals and species
13. Nutrition
14. Traditional health care methods
15. Food habits
16. Air pollution
17. Water pollution
18. Plantation
19. Soil protection
20. Renewable energy
21. Plant diseases
22. Yoga awareness and practice
23. Health care awareness programmes and their impact
24. Use of chemicals on fruits and vegetables
25. Organic farming
26. Crop rotation
27. Floury culture
28. Access to safe drinking water
29. Geographical survey
30. Geological survey
31. Sericulture
32. Study of species
33. Food adulteration
34. Incidence of Diabetes and other chronic diseases
35. Human genetics
36. Blood groups and blood levels
37. Internet Usage in Villages
38. Android Phone usage by different people
39. Utilisation of free electricity to farmers and related issues
40. Gender ration in schooling level- observation.

Complimenting the community service project the students may be involved to take up some awareness campaigns on social issues/special groups. The suggested list of programs

Programs for School Children

1. Reading Skill Program (Reading Competition)
2. Preparation of Study Materials for the next class.
3. Personality / Leadership Development
4. Career Guidance for X class students

5. Screening Documentary and other educational films
6. Awareness Program on Good Touch and Bad Touch (Sexual abuse)
7. Awareness Program on Socially relevant themes.

Programs for Women Empowerment

1. Government Guidelines and Policy Guidelines
2. Women's Rights
3. Domestic Violence
4. Prevention and Control of Cancer
5. Promotion of Social Entrepreneurship

General Camps

1. General Medical camps
2. Eye Camps
3. Dental Camps
4. Importance of protected drinking water
5. ODF awareness camp
6. Swatch Bharath
7. AIDS awareness camp
8. Anti Plastic Awareness
9. Programs on Environment
10. Health and Hygiene
11. Hand wash programmes
12. Commemoration and Celebration of important days

Programs for Youth Empowerment

1. Leadership
2. Anti-alcoholism and Drug addiction
3. Anti-tobacco
4. Awareness on Competitive Examinations
5. Personality Development

Common Programs

1. Awareness on RTI
2. Health intervention programmes
3. Yoga
4. Tree plantation
5. Programs in consonance with the Govt. Departments like –
 - i. Agriculture
 - ii. Health
 - iii. Marketing and Cooperation
 - iv. Animal Husbandry
 - v. Horticulture
 - vi. Fisheries
 - vii. Sericulture
 - viii. Revenue and Survey
 - ix. Natural Disaster Management
 - x. Irrigation
 - xi. Law & Order
 - xii. Excise and Prohibition
 - xiii. Mines and Geology
 - xiv. Energy

Role of Students:

- Students may not have the expertise to conduct all the programmes on their own. The students then can play a facilitator role.
- For conducting special camps like Health related, they will be coordinating with the Governmental agencies.
- As and when required the College faculty themselves act as Resource Persons.
- Students can work in close association with Non-Governmental Organizations like Lions Club, Rotary Club, etc or with any NGO actively working in that habitation.
- And also, with the Governmental Departments. If the program is rolled out, the District Administration could be roped in for the successful deployment of the program.
- An in-house training and induction program could be arranged for the faculty and participating students, to expose them to the methodology of Service Learning.

Timeline for the Community Service Project Activity**Duration: 8 weeks****1. Preliminary Survey (One Week)**

- A preliminary survey including the socio-economic conditions of the allotted habitation to be conducted.
- A survey form based on the type of habitation to be prepared before visiting the habitation with the help of social sciences faculty. (However, a template could be designed for different habitations, rural/urban.
- The Governmental agencies, like revenue administration, corporation and municipal authorities and village secretariats could be aligned for the survey.

2. Community Awareness Campaigns (One Week)

- Based on the survey and the specific requirements of the habitation, different awareness campaigns and programmes to be conducted, spread over two weeks of time. The list of activities suggested could be taken into consideration.

3. Community Immersion Programme (Three Weeks)

Along with the Community Awareness Programmes, the student batch can also work with any one of the below-listed governmental agencies and work in tandem with them. This community involvement programme will involve the students in exposing themselves to experiential learning about the community and its dynamics. Programs could be in consonance with the Govt. Departments.

4. Community Exit Report (One Week)

- During the last week of the Community Service Project, a detailed report of the outcome of the 8 weeks' works to be drafted and a copy shall be submitted to the local administration. This report will be a basis for the next batch of students visiting that habitation. The same report submitted to the teacher-mentor will be evaluated by the mentor and suitable marks are awarded for onward submission to the University.

Throughout the Community Service Project, a daily logbook need to be maintained by the students batch, which should be countersigned by the governmental agency representative and the teacher-mentor, who is required to periodically visit the students and guide them.