DEPARTMENT OF INFORMATION TECHNOLOGY

ACADEMIC REGULATIONS & SYLLABI (R-25)

B. Tech I Year

(Information Technology)

w.e.f. the Academic Year 2025-2026

Vidya Jyothi Institute of Technology

(An Autonomous Institution)

(Accredited by NAAC A+ Approved by AICTE New Delhi & Permanently Affiliated to JNTUH)

Aziz Nagar Gate, C.B. Post, Hyderabad-500 075

DEPARTMENT OF INFORMATION TECHNOLOGY

B. TECH I YEAR COURSE STRUCTURE 2025-26

Semester – I

S. No	Course Code	Course Title	L	T	P	Credits
1	A251002	Mathematics-I(Matrices & Calculus)	3	1	0	4.0
2	A251003	Advanced Engineering Physics	3	0	0	3.0
3	A251302	Engineering Drawing and computer aided drafting	2	0	2	3.0
4	A251201	Basic Electrical Engineering	3	0	0	3.0
5	A251503	Programming for Problem Solving	3	0	0	3.0
6	A251082	Advanced Engineering Physics Lab	0	0	2	1.0
7	A251583	Programming for Problem Solving Lab	0	0	2	1.0
8	A251281	Basic Electrical Engineering Lab	0	0	2	1.0
9		Induction Program				
Total			**14**	**1**	**8**	**19**

Semester – II

S. No	Course Code	Course Title	L	T	P	Credits
1	A252002	Mathematics – II (Ordinary Differential Equations & Vector Calculus)	3	0	0	3.0
2	A252004	Engineering Chemistry	3	0	0	3.0
3	A252001	English for Skill Enhancement	3	0	0	3.0
4	A252201	Basic Electronics	3	0	0	3.0
5	A252501	Data Structures	3	0	0	3.0
6	A252083	Engineering Chemistry Lab	0	0	2	1.0
7	A252581	Data Structures Lab	0	0	2	1.0
8	A252081	English Language & Communication Skills Lab	0	0	2	1.0
9	A252381	Engineering Workshop	0	0	2	1.0
10	A252583	Python Programming Lab	0	0	2	1.0
11	A252585	IT Workshop	0	0	2	1.0
Total			**15**	**0**	**12**	**21**

DEPARTMENT OF INFORMATION TECHNOLOGY

B. TECH II YEAR COURSE STRUCTURE 2025-26

Semester – I

S. No	Course Code	Course Title	L	T	P	Credits
1	A253001	Probability and Statistics	3	0	0	3.0
2	A2531201	Computer Organization and Microprocessor	3	0	0	3.0
3	A253502	Object Oriented Programming Through Java	3	0	0	3.0
4	A253503	Operating Systems	3	0	0	3.0
5	A2531202	Introduction to IoT	3	0	0	3.0
6	A253081	Computational Mathematics Lab	0	0	2	1.0
7	A253582	Object Oriented Programming Through Java Lab	0	0	2	1.0
8	A253583	Operating Systems Lab	0	0	2	1.0
9	A2531281	Internet of Things Lab	0	0	2	1.0
10	A2531282	Data Visualization-Power BI/Tableau	0	0	2	1.0
Total			**15**	**0**	**10**	**20**

Semester – II

S. No	Course Code	Course Title	L	T	P	Credits
1	A254004	Discrete Mathematics	3	0	0	3.0
2	A2541201	Data Communications and Computer Networks	3	0	0	3.0
3	A2541202	Formal Languages and Automata Theory	3	0	0	3.0
4	A2541203	Database Management Systems	3	0	0	3.0
5	A254501	Full Stack Development	3	0	0	3.0
6	A254005	Innovation and Entrepreneurship	2	0	0	1.0
7	A2541281	Computer Networks and Simulation Lab	0	0	2	1.0
8	A254586	Database Management Systems Lab	0	0	2	1.0
9	A254582	Full Stack Development Lab	0	0	2	1.0
10	A254587	UI Design- Flutter	0	0	2	1.0
11	A254001	Indian Knowledge System	1	0	0	1.0
Total			**18**	**0**	**08**	**22**

MATHEMATICS-I (MATRICES AND CALCULUS)

Department of Information Technology					I B. Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A251002	L	T	P	C	CIE	SEE	Total
	3	1	0	4	40	60	100

Pre-requisites: Mathematical Knowledge at pre-university level

Objectives: To learn

1. Applying basic operations on matrices and their properties.
2. Concept of a rank of the matrix and applying this concept to know the consistency and solving the system of linear equations.
3. Concept of eigenvalues and eigenvectors and to reduce the quadratic form to canonical form
4. Geometrical approach to the mean value theorems and their application to the mathematical problems
5. Finding maxima and minima of functions of two and three variables.
6. Evaluation of multiple integrals and their applications.

Course outcomes: After learning the contents of this paper, the student must be able to

1. Formulate the matrix representation of a given set of linear equations and analyze the solution of the system using appropriate methods.
2. Compute the eigenvalues and determine the corresponding eigenvectors of a given matrix.
3. Transform a quadratic form into its canonical form using orthogonal transformations.
4. Apply the mean value theorems to solve real-world and mathematical problems.
5. Determine the extreme values of functions of two variables with and without constraints.
6. Evaluate multiple integrals and apply them to calculate areas and volumes.

UNIT-I: Matrices 8 L

Rank of a matrix by Echelon form and Normal form — Inverse of Non-singular matrices by Gauss-Jordan method. System of linear equations: Solving system of Homogeneous and Non-Homogeneous equations.

UNIT-II: Eigen values and Eigen vectors 10 L

Linear Transformation and Orthogonal Transformation: Eigen values — Eigen vectors and their properties – Diagonalization of a matrix – Cayley-Hamilton Theorem (without proof) – Finding inverse and power of a matrix by Cayley-Hamilton Theorem. Quadratic forms and Nature of the Quadratic Forms – Reduction of Quadratic form to canonical form by Orthogonal Transformation.

UNIT-III: Single Variable Calculus 10 L

Limit and Continuous of functions and its properties. Mean value theorems: Rolle's theorem — Lagrange's Mean value theorem with their Geometrical Interpretation and applications – Cauchy's Mean value Theorem — Taylor's Series (All the theorems without proof).
Curve Tracing: Curve tracing in Cartesian coordinates.

UNIT-IV: Multivariable Calculus (Partial Differentiation and applications) 10 L

Definitions of Limit and continuity – Partial Differentiation: Total derivative – Jacobian — Functional dependence & independence. Applications: Maxima and minima of functions of two variables and three variables using method of Lagrange multipliers.

UNIT-V: Multivariable Calculus (Integration) 10 L

Evaluation of Double Integrals (Cartesian and polar coordinates) – change of order of integration (only Cartesian form) — Change of variables for double integrals (Cartesian to polar). Evaluation of Triple Integrals — Change of variables for triple integrals (Cartesian to Spherical and Cylindrical polar coordinates).

TEXT BOOKS:

1. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 36th Edition, 2010.
2. R.K. Jain and S.R.K. Iyengar, Advanced Engineering Mathematics, Narosa Publications, 5th Editon, 2016.

REFERENCE BOOKS:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
2. G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, 9thEdition, Pearson, Reprint, 2002.
3. N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2008.
4. H. K. Dass and Er. Rajnish Verma, Higher Engineering Mathematics, S Chand and Company Limited, New Delhi.

ADVANCED ENGINEERING PHYSICS

Department of Information Technology					I B. Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A251003	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Course Objectives:

1. To study crystal structures and material characterization techniques like XRD.
2. To learn the various wave optics techniques, phenomena and applications.
3. To understand fundamental concepts of quantum mechanics and their applications in solids.
4. To introduce quantum computing principles, quantum gates, and basic quantum measurements.
5. To explore the working and applications of lasers and fibre optics in modern technology.

Course Outcomes:

1. Analyze crystal structures and apply XRD technique or material characterization.
2. Classify various wave optics techniques and explain their phenomena and applications.
3. Apply quantum mechanical principles to explain particle behaviour and discrete energy levels in materials.
4. Understand quantum computing concepts, use quantum gates, and explain basic quantum measurements.
5. Explain the principles of lasers and fibre optics and their applications in various communications.

UNIT-I: Crystallography & Materials Characterization [09Hrs]

Introduction: Unit cell, Space lattice, Basis, Lattice parameters; Crystal structures, Bravais lattices, Atomic radius, Number of atoms per unit cell, Coordination number, Distance of nearest neighbourhood, packing factor of Simple cubic (SC), Body Centered Cubic (BCC), Face Centered Cubic (FCC) structure; Miller indices, Inter-planar distance. Concept of Nanomaterials: Surface area to Volume ratio, X -ray diffraction: Bragg's law, Powder method, Calculation of average crystallite size using Debye Scherrer formula.

UNIT-II: Wave Optics [09 Hrs]

Principle of superposition, Coherence, Interference - Interference in thin films by reflection, Newton's rings. Diffraction – Fresnel and Fraunhofer diffraction, Fraunhofer diffraction due to single slit, Plane diffraction grating, Resolving power of grating (qualitative treatment), Polarization – Polarization of light waves, Plane of vibration, Plane of polarization, Double refraction, Nicol's Prism, Applications of polarization.

UNIT-III: Quantum Mechanics [09Hrs]

Introduction: de-Broglie hypothesis, Heisenberg uncertainty principle, Postulates of quantum mechanics: Operators in quantum mechanics, Eigen values and Eigen functions, Expectation value, Physical significance of wave function, Schrödinger's time independent wave equation, Application of Schrödinger's time independent wave equation for a particle in a1-D box: Expression for solution of wave function and derivation for Eigen energy states, Band Theory of solids, Classification of solids based on Band theory.

UNIT-IV: Quantum Computing [09Hrs]

Introduction to Quantum Computing, Mathematical Tools: Linear Algebra Basics, Tensor Product, Dirac Notation (Bra-Ket) and Properties, Hilbert Space, Bloch Sphere, Superposition, Classical Bits vs Quantum Bits (Qubits), Multiple Qubit Systems and Entanglement, Quantum Evolution and Measurements. Quantum Gates: Hadamard Gate, CNOT, Pauli-X gate, Swap Gate and Quantum Circuits. Future of Quantum Computing.

UNIT-V: Laser and Fiber Optics [09Hrs]

Introduction to interaction of radiation with matter: Absorption, Spontaneous emission and Stimulated emission, Einstein coefficients and their relations, Characteristics of a laser, Population inversion, Components of a laser: active medium, pumping source, optical resonator, Construction and working of Ruby laser, He-Ne laser, Applications of lasers.

Introduction to Optical Fibers: Total internal reflection, Construction of optical fiber, Acceptance angle and Numerical aperture, Classification of optical fibers (step and graded index fibers), Block diagram of optical fiber communication system, Applications of optical fibers.

TEXT BOOKS:

1. Engineering Physics, Rajendran; McGraw Hill Education.
2. Engineering Physics by B K Pandey and S Chaturvedi; CENGAGE Learning.
3. Introduction to Classical and Quantum Computing, by Thomas G. Wong; Rooted Grove.

REFERENCE BOOKS:

1. Quantum computing: A Gentle Introduction by Rieffel and Polak; The MIT Press.
2. Introduction to Solid State Physics, Charles Kittel, John Wiley & Sons, Inc.
3. Optics, Ajoy K. Ghatak, McGraw Hill Education India.

ENGINEERING DRAWING AND COMPUTER AIDED DRAFTING

Department of Information Technology					I B. Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A251302	L	T	P	C	CIE	SEE	Total
	2	0	2	3	40	60	100

Course Out comes: At the end of the course, the student will be able to:

1. Understand and apply the fundamental principles of engineering graphics.
2. Apply orthographic projections to points, lines, and planes using conventional and computer-aided methods.
3. Analyze projections and sectional views of regular solids using conventional and computer-aided methods.
4. Develop surfaces of basic solids like prism, cylinder, pyramid, and cone.
5. Create isometric projections and convert between isometric and orthographic views.

UNIT–I: Introduction to Engineering Graphics (Conventional)

Principles of Engineering Graphics and their Significance, Geometrical Constructions, Scales, Plain and Diagonal, Conic Sections including the Rectangular Hyperbola, General method only. Cycloid, Epicycloid and Hypocycloid.

UNIT-II: Orthographic Projections (Conventional and Computer Aided)

Principles of Orthographic Projections, Conventions, Projections of Points and Lines, Projections of Plane regular geometric figures. Auxiliary Planes. Computer aided orthographic projections, points, lines and planes. Introduction to Computer aided drafting, views, commands and conics.

UNIT–III: Projections of Regular Solids (Conventional and Computer Aided)

Auxiliary Views, Sections or Sectional views of Right Regular Solids, Prism, Cylinder, Pyramid, Cone, Auxiliary views, Computer aided projections of solids, sectional views.

UNIT–IV: Development of Surfaces (Conventional)

Prism, Cylinder, Pyramid and Cone.

UNIT–V: Isometric Projections (Conventional and Computer Aided)

Principles of Isometric Projection, Isometric Scale, Isometric Views, Conventions, Isometric Views of Lines, Plane Figures, Simple and Compound Solids, Isometric Projection of objects having non, isometric lines. Isometric Projection of Spherical Parts. Conversion of Isometric Views to Orthographic Views and Vice- versa, Conventions. Conversion of orthographic projection into isometric view.

TEXTBOOKS:

1. Engineering Drawing, N.D. Bhatt, Charotar, 54th Edition,2023.
2. Engineering Drawing and graphics Using AutoCAD, T. Jeyapoovan and Vikas, S. Chand and company Ltd., 3rdEdition,2010.

REFERENCE BOOKS:

1. Engineering Drawing, Basant Agrawal and C.M. Agrawal, McGraw Hill,3rd Edition,2019.

BASIC ELECTRICAL ENGINEERING

Department of Information Technology					I B. Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A251201	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Prerequisites: Mathematics

Course Objectives:

- To understand DC and Single & Three phase AC circuits
- To study and understand the different types of DC, AC machines and Transformers.
- To import the knowledge of various electrical installations and the concept of power, power factor and its improvement.

Course Outcomes: At the end of the course, students will be the able to

CO1. Explain basic principles of electrical elements.
CO2. Apply the concepts of AC circuits to various elements and combinations.
CO3. Examine principle and tests of transformer.
CO4. Contrast the working of DC machines and induction motors.
CO5. Assess working principle of AC generator and electrical installations.

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	1	0	1	0	0	2	0	2	2
CO2	3	2	1	0	3	1	1	1	2	1	2
CO3	3	2	1	1	3	2	0	1	0	2	2
CO4	3	2	1	0	1	0	0	2	0	2	2
CO5	3	2	1	0	3	1	1	1	2	1	2

UNIT–I:

D.C. Circuits: Electrical circuit elements (R, L and C), voltage and current sources, KVL&KCL, analysis of simple circuits with dc excitation. Superposition, Thevenin's and Norton's Theorems – simple problems.

UNIT-II:

A.C. Circuits: Representation of sinusoidal waveforms, peak and rms values, phasor representation, real power, reactive power, apparent power, power factor, Analysis of single-phase series ac circuits consisting of R, L, C, RL, RC, RLC combinations – simple problems, resonance in series R-L-C circuit. Three-phase balanced circuits, voltage and current relations in star and delta connections.

UNIT-III:

Transformers: Construction and working of Single-Phase Transformer, Ideal and practical transformer, equivalent circuit, losses in transformers, regulation and efficiency. Working Principle and applications of Auto-transformer.

UNIT-IV:

Electrical Machines: Construction of DC Machine, working principle of DC Motor, performance characteristics of DC Shunt motor. Construction and working of a Three-phase induction motor – Torque – Speed Characteristics, Construction and working of Single-phase induction motor, Construction and working of Three Phase Alternator – No Load Characteristics.

UNIT-V:

Electrical Installations: Components of LT Switchgear: Switch Fuse Unit (SFU), MCB, Types of Wires and Cables, Earthing. Types of Batteries, Important Characteristics for Batteries. Elementary calculations for energy consumption, battery backup.

TEXT BOOKS:

1. D.P. Kothari and I. J. Nagrath, “Basic Electrical Engineering”, Tata McGraw Hill, 4th Edition, 2019.
2. MS Naidu and S Kamakshaiah, “Basic Electrical Engineering”, Tata McGraw Hill, 2nd Edition, 2008.

REFERENCE BOOKS:

1. P. Ramana, M. Suryakalavathi, G.T. Chandrasheker, “Basic Electrical Engineering”, S. Chand, 2nd Edition, 2019.
2. D. C. Kulshreshtha, “Basic Electrical Engineering”, McGraw Hill, 2009
3. M. S. Sukhija, T. K. Nagsarkar, “Basic Electrical and Electronics Engineering”, Oxford, 1st Edition, 2012.

PROGRAMMING FOR PROBLEM SOLVING

Department of Information Technology					I B. Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A251503	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Course Outcomes: At the end of the course, student will be able to:

1. Write algorithms, draw flowcharts and convert them into modular C programs using variables, control structures, and loops.
2. Apply arrays and functions to design structured and reusable programs.
3. Use pointers and string manipulation techniques to develop modular C programs
4. Implement recursive solutions and create programs using structures, unions, and enumerated types.
5. Perform file handling operations and implement basic searching and sorting algorithms.

UNIT-I

Overview of C: C Language Elements, Variable Declarations and Data Types, Executable Statements, General Form of a C Program, Arithmetic Expressions, Formatting Numbers in Program Output.
Selection Structures: Control Structures, Conditions, if Statement, if Statements with Compound Statements, Decision Steps in Algorithms.
Loop Statements: Repetition in Programs, Counting Loops and the while Statement, Computing a Sum or Product in a Loop, for Statement, Conditional Loops, Loop Design, Nested Loops, do-while Statement.

UNIT-II

Arrays: Declaring and Referencing Arrays, Array Subscripts, Using for Loops for Sequential Access, Multi-dimensional Arrays.
Top-Down Design with Functions: Building Programs from Existing Information, Library Functions, Top-Down Design and Structure Charts, Functions without Arguments, Functions with Arguments, Using Array Elements as Function Arguments.

UNIT-III

Pointers and Modular Programming: Pointers and the Indirection Operator, Functions with Output Parameters, Multiple Calls to a Function with Input/ Output Parameters,
Scope of Names, Formal Output Parameters as Actual Arguments.
Strings: String Basics, String Library Functions: Assignment and Substrings, Longer Strings-Concatenation and Whole-Line Input, String Comparison.

UNIT-IV

Recursion: The Nature of Recursion, Tracing a Recursive Function,
Structure and Union Types: User-Defined Structure Types, Structure Type Data as Input and Output Parameters, Union Types, Enumerated Types.

UNIT-V

Text and Binary File Pointers: Input/ Output Files, Binary Files.
Sorting and Searching: Basic searching in an array of elements (linear and binary search techniques), Basic algorithms to sort array of elements (Bubble and Insertion algorithms).

TEXT BOOKS:

1. Jeri R. Hanly and Elliot B. Koffman, Problem solving and Program Design in C 7th Edition, Pearson.
2. B.A. Forouzan and R.F. Gilberg C Programming and Data Structures, Cengage Learning, (3rd Edition).

REFERENCE BOOKS:

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice-Hall of India.
2. E. Balagurusamy, Computer Fundamentals and C, 2nd Edition, McGraw-Hill.
3. Yashawant Kanetkar, Let Us C, 18th Edition, BPB.
4. R.G. Dromey, How to solve it by Computer, Pearson (16th Impression).
5. Programming in C, Stephen G. Kochan, Fourth Edition, Pearson Education.
6. Herbert Schildt, C: The Complete Reference, McGraw-Hill, 4th Edition.
7. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill.

ADVANCED ENGINEERING PHYSICS LAB

Department of Information Technology					I B. Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A251082	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Objectives:

1. To enable students to determine fundamental physical constants and material parameters using modern physics experiments.
2. To provide practical exposure to optical phenomena such as interference, diffraction, and dispersion using precision instruments.
3. To develop skills in the characterization of semiconductor devices including LEDs, photodiodes, and solar cells.
4. To impart knowledge and hands-on experience with laser-based measurements.
5. To familiarize students with optical fiber parameters and loss measurements, and strengthen skills in data analysis, interpretation, and reporting.

Course Outcomes: By the end of the course, the student will be able to:

1. **CO1:** Determine fundamental constants such as Planck's constant and the energy gap of a semiconductor.
2. **CO2:** Analyze the optical phenomena including interference, diffraction, and dispersion through experimental study.
3. **CO3:** Evaluate the electrical and optical characteristics of semiconductor devices such as LEDs, photodiodes, and solar cells.
4. **CO4:** Demonstrate working knowledge of lasers through wavelength determination using single-slit and diffraction grating methods.
5. **CO5:** Determine the numerical aperture and bending losses of optical fibers, and apply scientific methods for accurate data collection and reporting.

List of Experiments:

1. Determination of work function and Planck's constant using the photoelectric effect.
2. Determination of the energy gap of a semiconductor.
3. Determination of the radius of curvature of a plano-convex lens by forming Newton's rings.
4. Determination of the wavelength of a given monochromatic light source using a plane diffraction grating.
5. Determination of the dispersive power of a given prism.
6. Determination of the operating point for maximum efficiency of a given LED.
7. Determination of the operating point of a photodiode under varying illumination levels
8. Determination of the optimum load for maximum power output of a given solar cell.
9. (a)Determination of the wavelength of a laser using a single slit.
   (b) Determination of the wavelength of a laser using a diffraction grating (N-slits).
10. (a)Determination of the numerical aperture of a given optical fiber.
    (b) Determination of the bending losses in a given optical fiber.

Note: Any 8 experiments are to be performed.

TEXTBOOKS:

Laboratory Manual of Engineering Physics, Aparna and K. Venkateswara Rao, V.G.S. Publishers.

PROGRAMMING FOR PROBLEM SOLVING LAB

Department of Information Technology					I B. Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A251583	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Outcomes:

At the end of the course, student will be able to:

1. Apply algorithms, flowcharts, and control structures to solve numeric and expression-based problems in C.
2. Develop programs using arrays, pointers, and functions to perform computations and implement modular solutions.
3. Implement string manipulation operations and apply file handling techniques for data processing in C.
4. Design and test programs using user-defined functions and apply structured programming principles for problem solving.
5. Implement sorting & searching techniques and analyze their efficiency through programming.

PRACTICE SESSIONS:

Simple numeric problems:

a) Write a program for finding the max and min from the three numbers.
b) Write the program for the simple, compound interest.
c) Write a program that prints a multiplication table for a given number and the number of rows in the table. For example, for a number 5 and rows = 3, the output should be:

5x1=5
5x2=10
5x3=15

d) Write a program that shows the binary equivalent of a given positive number between 0 to 255.

Expression Evaluation:

a) Write a C program, which takes two integer operands and one operator from the user, performs the operation and then prints the result. (Consider the operators +,-,*, /, % and use Switch Statement).
b) Write a program that finds if a given number is a prime number.
c) Write a C program to find the sum of individual digits of a positive integer and test given number is palindrome.
d) A Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C program to generate the first n terms of the sequence.

Arrays, Pointers, and Functions:

a) Write a C program to find the minimum, maximum, and average in an array of integers.
b) Write a C program that uses functions to perform the following:
i) Addition of Two Matrices ii) Multiplication of Two Matrices.

c) Write a program for reading elements using a pointer into an array and displaying the values using the array.
d) Write a program to display values in reverse order from an array using a pointer.

Files:

a) Write a C program that copies one file to another, replacing all lowercase characters with their uppercase equivalents.
b) Write a C program to merge two files into a third file (i.e., the contents of the first file followed by those of the second are put in the third file).

Strings:

a) Write a C program that uses functions to perform the following operations:
   I. To insert a sub-string into a given main string from a given position.
   II. To delete characters from a given position in a given string.
b) Write a C program to determine if the given string is a palindrome or not (Spelled the same in both directions, with or without a meaning, like madam, civic, noon, abcba, etc.).
c) Write a C program that displays the position of a character 'ch' in the given string 'S' or -1 if 'S' doesn't contain 'ch'.
d) Write a C program to count the lines, words, and characters in a given text.

Sorting and Searching:

a) Write a C program that uses a non-recursive function to search for a key value in a given list of integers using the linear search method.
b) Write a C program that uses a non-recursive function to search for a key value in a given sorted list of integers using the binary search method.
c) Write a C program that implements the Bubble sort method to sort a given list of integers in ascending order.
d) Write a C program that sorts the given array of integers using the selection sort in descending order.
e) Write a C program that sorts the given array of integers using the insertion sort in ascending order.
f) Write a C program that sorts a given array of names.

TEXTBOOKS:

1. Jeri R. Hanly and Elliot B. Koffman, Problem Solving and Program Design in C, 7th Edition, Pearson.
2. B.A. Forouzan and R.F. Gilberg, C Programming and Data Structures, Cengage Learning, (3rd Edition).

REFERENCE BOOKS:

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice-Hall of India.
2. E. Balagurusamy, Computer Fundamentals and C, 2nd Edition, McGraw-Hill.
3. Yashavant Kanetkar, Let Us C, 18th Edition, BPB.
4. R.G. Dromey, How to solve it by Computer, Pearson (16th Impression).
5. Programming in C, Stephen G. Kochan, Fourth Edition, Pearson Education.
6. Herbert Schildt, C: The Complete Reference, McGraw-Hill, 4th Edition.
7. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill.

BASIC ELECTRICAL ENGINEERING LAB

Department of Information Technology					I B. Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A251281	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Objectives:

1. To measure the electrical parameters for different types of DC and AC circuits using conventional and theorems approach.
2. To study the transient response of various R, L and C circuits using different excitations.
3. To determine the performance of different types of DC, AC machines and Transformers.

Course Outcomes: At the end of the course, students will be the able to

CO1. Explain basic principles of electrical elements.
CO2. Apply the concepts of AC circuits to various elements and combinations.
CO3. Examine principle and tests of transformer.
CO4. Contrast the working of DC machines and induction motors.
CO5. Assess working principle of AC generator and electrical installations.

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	1	0	1	0	0	2	0	2	2
CO2	3	2	1	0	3	1	1	1	2	1	2
CO3	3	2	1	1	3	2	0	1	0	2	2
CO4	3	2	1	0	1	0	0	2	0	2	2
CO5	3	2	1	0	3	1	1	1	2	1	2

Any Ten experiments should be conducted

1. Verification of Ohm's Law
2. Verification of KVL and KCL
3. Verification of Superposition theorem.
4. Verification of Thevenin's theorem
5. Verification of Norton's theorem
6. Calculations and Verification of Impedance and Current of RL and RC series circuits
7. Transient Response of Series RL and RC Circuits for DC Excitation.
8. Measurement of Voltage, Current and Real Power in primary and Secondary Circuits of a Single-Phase Transformer
9. Load Test on Single Phase Transformer (Calculate Efficiency and Regulation)
10. Performance Characteristics of a DC Shunt Motor
11. Torque-Speed Characteristics of a Three-phase Induction Motor.
12. No-Load Characteristics of a Three-phase Alternator

TEXT BOOKS:

1. D.P. Kothari and I. J. Nagrath, “Basic Electrical Engineering”, Tata McGraw Hill, 4th Edition, 2019.
2. MS Naidu and S Kamakshaiah, “Basic Electrical Engineering”, Tata McGraw Hill, 2nd Edition, 2008.

REFERENCE BOOKS:

1. P. Ramana, M. Suryakalavathi, G.T.Chandrasheker,”Basic Electrical Engineering”, S. Chand, 2nd Edition, 2019.
2. D. C. Kulshreshtha, “Basic Electrical Engineering”, McGraw Hill, 2009
3. M. S. Sukhija, T. K. Nagsarkar, “Basic Electrical and Electronics Engineering”, Oxford, 1st Edition, 2012.

MATHEMATICS II (ORDINARY DIFFERENTIAL EQUATIONS AND VECTOR CALCULUS)

Department of Information Technology					I B. Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A252002	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Course Objectives: To learn

1. Methods of solving the differential equations of first and higher order.
2. Concept, properties of Laplace transforms.
3. Solving ordinary differential equations using Laplace transforms techniques.
4. The physical quantities involved in engineering field related to vector valued functions
5. The basic properties of vector valued functions and their applications to line, surface and volume integrals

Course outcomes: After learning the contents of this paper, the student must be able to

1. Apply the concept of first order differential equations to solve real world problems
2. Solve higher differential equation and apply the concept of differential equation to real world problems.
3. Use the Laplace Transforms techniques for solving Ordinary Differential Equations.
4. Apply the concepts of vector differentiation to compute gradient, divergence, and curl, and interpret their physical significance in engineering problems.
5. Evaluate line, surface, and volume integrals, and apply vector integration theorems such as Green's, Gauss's, and Stokes's to solve engineering problems.

UNIT-I: First Order Ordinary Differential Equations

Exact differential equations – Equations reducible to exact differential equations – linear and Bernoulli's equations – Orthogonal Trajectories (only in Cartesian Coordinates). Applications: Newton's law of cooling — Law of natural growth and decay.

UNIT-II: Ordinary Differential Equations of Higher Order

Higher order linear differential equations with constant coefficients: Non-Homogeneous terms of the type e^{ax}, sin , cos ax, polynomials in x, $e^{ax}V(x)$ and $x\,V(x)$ – Method of variation of parameters.

UNIT-III: Laplace Transforms

Laplace Transforms: Laplace Transform of standard functions — First and second shifting theorems — Laplace transforms of functions multiplied by 't' and divided by 't' — Laplace transforms of derivatives and integrals of function — Evaluation of integrals by Laplace transforms — Laplace transform of periodic functions — Inverse Laplace transform by different methods, convolution theorem (without proof). Applications: solving Initial value problems by Laplace Transform method.

UNIT-IV: Vector Differentiation

Vector point functions and scalar point functions — Gradient — Directional derivatives — Divergence — Solenoidal Vector — Curl — Irrotational vectors – Vector Identities.

UNIT-V: Vector Integration

Line, Surface and Volume Integrals. Theorems of Green, Gauss and Stokes (without proofs) and their applications

TEXT BOOKS:

1. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 36th Edition, 2010.
2. R.K. Jain and S.R.K. Iyengar, Advanced Engineering Mathematics, Narosa Publications, 5th Edition, 2016.

REFERENCE BOOKS:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
2. G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, 9th Edition, Pearson, Reprint, 2002.
3. N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2008.
4. H. K. Dass and Er. Rajnish Verma, Higher Engineering Mathematics, S Chand and Company Limited, New Delhi.

ENGINEERING CHEMISTRY

Department of Information Technology					I B. Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A252004	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Course Objectives:

1. To understand the industrial significance of water treatment, fundamental principles of battery chemistry, and the impact of corrosion along with its control methods for structural protection.
2. To impart foundational knowledge of various energy sources and their practical applications in engineering.
3. To equip students with an understanding of biosensors, and analytical techniques applicable in engineering, industrial, environmental, and biomedical fields.

Course Outcomes:

1. **Identify** and **explain** the fundamental properties of water and its applications in domestic and industrial systems.
2. **Describe** the principles of electrochemical processes and **demonstrate** basic methods of corrosion prevention.
3. **Explain** the working of batteries and energy sources and **analyze** their relevance in engineering and entrepreneurial applications.
4. **Define** and **classify** polymers and other engineering materials based on their structure, properties, and applications.
5. **Apply** the principles of material properties and **analyze** analytical techniques for engineering and environmental problem-solving.

UNIT-I: Water and its treatment: [8]

Introduction- Hardness, types, degree of hardness and units. Estimation of hardness of water by complexometric method - Numerical problems. Potable water and its specifications (WHO) - Steps involved in the treatment of potable water - Disinfection of potable water by chlorination and breakpoint chlorination. Defluoridation - Nalgonda technique. Boiler troubles: Scales, Sludges and Caustic embrittlement. Internal treatment of boiler feed water - Calgon conditioning, Phosphate conditioning, Colloidal conditioning. External treatment methods - Softening of water by ion- exchange processes. Desalination of brackish water – Reverse osmosis.

UNIT-II: Electrochemistry and Corrosion: [8]

Introduction- Electrode potential, standard electrode potential, Nernst equation (no derivation), electrochemical cell - Galvanic cell, cell representation, EMF of cell - Numerical problems. Types of electrodes, reference electrodes - Primary reference electrode - Standard Hydrogen Electrode (SHE), Secondary reference electrode - Calomel electrode. Construction, working and determination of pH of unknown solution using SHE, Quinhydrone electrode and Calomel electrode.

Corrosion: Introduction- Definition, causes and effects of corrosion – Theories of corrosion, chemical and electrochemical theories of corrosion, Types of corrosion: galvanic, water-line and pitting corrosion. Factors

affecting rate of corrosion - Nature of the metal (Position of metal in galvanic series, Purity of metal, Nature of corrosion product), Nature of the corroding environment (Temperature, Humidity, and pH). Corrosion control methods - Cathodic protection Methods - Sacrificial anode and impressed current methods.

UNIT–III: Energy sources: [8]

Batteries: Introduction – Classification of batteries - Primary, secondary and reserve batteries with examples. Construction, working and applications of Zn-air and Lithium-ion battery. Fuel Cells – Differences between a battery and a fuel cell, Construction and applications of Direct Methanol Fuel Cell (DMFC).
Fuels: Introduction, classification and characteristics of a good fuel, Calorific value – Units - HCV, LCV- Dulongs formula - Numerical problems.
Fossil fuels: Introduction, Petroleum - Refining of Crude oil, Cracking - Types of cracking - Moving bed catalytic cracking. LPG and CNG composition and uses.
Synthetic Fuels: Fischer-Tropsch process, Introduction and applications of Hythane and Green Hydrogen.

UNIT - IV: Polymers: [8]

Definition - Classification of polymers: Based on origin and tacticity with examples – Types of polymerization - Addition (free radical addition mechanism) and condensation polymerization. Plastics, Elastomers and Fibers: Definition, preparation, properties and applications of PVC, Buna-S, Nylon-6,6. Differences between thermoplastics and thermo setting plastics, Fiber reinforced plastics (FRP).
Conducting polymers: Definition and Classification with examples - Mechanism of conduction in transpoly-acetylene and applications of conducting polymers.
Biodegradable polymers: Polylactic acid and its applications.

UNIT-V-Engineering Materials: [8]

Lubricants: Definition and Characteristics of a good lubricant- thin film mechanism of lubrication, properties of lubricants- viscosity, cloud and pour point, flash and fire point.
Biosensor - Definition, Classification, Amperometric Glucose monitor sensor.
Interpretative spectroscopic applications of UV-Visible spectroscopy for Analysis of pollutants in dye industry, IR spectroscopy in night vision-security, Pollution Under Control- CO sensor (Passive Infrared detection),

TEXT BOOKS:

1. Engineering Chemistry by P.C. Jain and M. Jain, Dhanpatrai Publishing Company, 2010.
2. Engineering Chemistry by Rama Devi, Dr. P. Aparna and Rath, Cengage learning, 2025.

REFERENCE TEXT BOOKS:

1. Engineering Chemistry: by Thirumala Chary Laxminarayana & Shashikala, Pearson Publications (2020)
2. Engineering Chemistry by Shashi Chawla, Dhanpatrai and Company (P) Ltd. Delhi 2011.
3. Engineering Chemistry by Shikha Agarwal, Cambridge University Press, Delhi 2015.
4. Challenges and Opportunities in Green Hydrogen by Editors: Paramvir Singh, Avinash Kumar Agarwal, Anupma Thakur, R.K Sinha.
5. E-Content- https://doi.org/10.1142/13094 | October 2023
6. E-books: https://archive.org/details/EngineeringChemistryByShashiChawla/page/n11/mode/2u

ENGLISH FOR SKILL ENHANCEMENT

Department of Information Technology					I B. Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A252001	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Course Objectives: This course will enable the students to:

1. Improve their vocabulary.
2. Use appropriate sentence structures in their oral and written communication.
3. Develop their reading and study skills.
4. Equip students to write paragraphs, essays, précis and draft letters.
5. Acquire skills for Technical report writing.

Course Outcomes: Students will be able to:

1. Choose appropriate vocabulary in their oral and written communication.
2. Demonstrate their understanding of the rules of functional grammar and sentence structures.
3. Develop comprehension skills from known and unknown passages.
4. Write paragraphs, essays, précis and draft letters.
5. Write abstracts and reports in various contexts.

UNIT –I

Theme : Perspectives

Lesson on 'The Generation Gap' by Benjamin M. Spock from the prescribed textbook titled ***English for the Young in the Digital World*** published by Orient Black Swan Pvt. Ltd

Vocabulary : The Concept of Word Formation-The Use of Prefixes and Suffixes-Words Often Misspelt - Synonyms and Antonyms

Grammar : Identifying Common Errors in Writing with Reference to Parts of Speech particularly Articles and Prepositions—Degrees of Comparison

Reading : Reading and Its Importance-Sub Skills of Reading–Skimming and Scanning.

Writing : Sentence Structures and Types -Use of Phrases and Clauses in Sentences- Importance of Proper Punctuation-Techniques for Writing Precisely–Nature and Style of Formal Writing.

UNIT–II

Theme : Digital Transformation

Lesson on '*Emerging Technologies*' from the prescribed textbook titled ***English for the Young in the Digital World*** **published by** Orient Black Swan Pvt. Ltd.

Vocabulary : Homophones, Homonyms and Homographs

Grammar : Identifying Common Errors in Writing with Reference to Noun-pronoun Agreement and Subject-verb Agreement.

Reading : Reading Strategies-Guessing Meaning from Context Identifying Main Ideas – Exercises for Practice

Writing : Paragraph Writing—Types, Structures and Features of a Paragraph - Creating Coherence—Linkers and Connectives - Organizing Principles in a Paragraph—Defining- Describing People, Objects, Places and Events—Classifying- Providing Examples or Evidence - Essay Writing - Writing Introduction and Conclusion.

UNIT–III

Theme : Attitude and Gratitude

Poems on' *Leisure'* by William Henry Davies and '*Be Thankful'* Unknown Author from the prescribed textbook titled ***English for the Young in the Digital World*** **published** by Orient Black Swan Pvt. Ltd.

Vocabulary : Words Often Confused-Words from Foreign Languages and their Use in English.

Grammar : Identifying Common Errors in Writing with Reference to Misplaced Modifiers and Tenses.

Reading : Sub-Skills of Reading – Identifying Topic Sentence and Providing Supporting Ideas - Exercises for Practice.

Writing : Format of a Formal Letter-Writing Formal Letters E.g. Letter of Complaint, Letter of Requisition, Job Application with CV/ Resume–Difference between Writing a Letter and an Email - Email Etiquette.

UNIT–IV

Theme : Entrepreneurship

Lesson on '*Why a Start-Up Needs to Find its Customers First* 'by Pranav Jain from the prescribed text book titled ***English for the Young in the Digital World*** published by Orient Black Swan Pvt. Ltd.

Vocabulary : Standard Abbreviations in English–Inferring Meanings of Words through Context– Phrasal Verbs—Idioms.

Grammar : Redundancies and Clichés in Written Communication – Converting Passive to Active Voice and Vice-Versa.

Reading : Prompt Engineering Techniques–Comprehending and Generating Appropriate Prompts - Exercises for Practice

Writing : Writing Practices-Note Making-Précis Writing.

UNIT–V

Theme : Integrity and Professionalism

Lesson on '*Professional Ethics*' from the prescribed textbook titled ***English for the Young in the Digital World*** **published by Orient Black Swan Pvt. Ltd**.

Vocabulary: Technical Vocabulary and their Usage–One Word Substitutes–Collocations.

Grammar : Direct and Indirect Speech-Common Errors in English (Covering all the other aspects of grammar which were not covered in the previous units)

Reading : Survey, Question, Read, Recite and Review (SQ3R Method) – Inferring the Meaning and Evaluating a Text-Exercises for Practice

Writing : Report writing-Technical Reports-Introduction–Characteristics of Report – Categories of Reports Formats- Structure of Reports (Manuscript Format)- Types of Reports- Writing a Technical Report.

TEXTBOOKS:

1. Board of Editors.*2025. English for the Young in the Digital World.* Orient Black Swan Pvt.Ltd.

REFERENCE BOOKS:

1. Swan, Michael.(2016).Practical English Usage .Oxford University Press. New Edition.
2. Karal,Rajeevan.2023 .English Grammar Just for You .Oxford University Press .New Delhi
3. 2024. Empowering with Language: Communicative English for Undergraduates. Cengage Learning India Pvt. Ltd. New Delhi
4. Sanjay Kumar & Pushp Lata. 2022. Communication Skills – A Workbook. Oxford University Press. New Delhi
5. Wood, F.T.(2007).Remedial English Grammar. Macmillan.
6. Vishwamohan, Aysha.(2013).English for Technical Communication for Engineering Students. McGraw- Hill Education India Pvt.

BASIC ELECTRONICS

Department of Information Technology					I B. Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A252201	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Course Outcomes: By the end of this course, students will be able to

CO1: Analyze the electrical characteristics and models of semiconductor diodes and apply them in rectifier and clipping circuits.
CO2: Evaluate the operation and configurations of Bipolar Junction Transistors (BJTs) and analyze their input and output characteristics.
CO3: Design appropriate biasing networks for BJTs and determine the operating point for amplifier applications.
CO4: Analyze transistor amplifier circuits using h-parameter models and assess performance for various configurations.
CO5: Analyze the structure, working, and characteristics of JFETs, MOSFETs

UNIT-I

Diode Characteristics and Applications: PN junction diode – I-V characteristics, Diode resistance and capacitance, Diode models (Ideal, Simplified, Piecewise Linear), Rectifiers – Half-wave, Full- wave (Center-tap and bridge), Capacitor filter for rectifiers, Clippers and clampers, Zener diode – I-V characteristics and voltage regulation.

UNIT - II

Bipolar Junction Transistor (BJT): Structure and working principle of BJT, Current components and transistor action, Configurations: Common Base (CB), Common Emitter (CE), Common Collector (CC), Input and output characteristics, Determination of h-parameters from transistor characteristics.

UNIT - III

BJT Biasing: Need for biasing and stabilization, Load line and operating point, Biasing techniques: Fixed bias, Collector-to-base bias, Voltage divider bias, Stability factors and thermal runaway

UNIT - IV

Field Effect Transistor (FET): JFET- Construction, Principle of Operation, Pinch-Off Voltage, Volt-Ampere Characteristic, Comparison of BJT and FET, FET as Voltage Variable Resistor, MOSFET, MOSFET as a capacitor.

UNIT – V

Special Purpose Devices: Zener Diode - Characteristics, Zener diode as Voltage Regulator, Principle of Operation - SCR, Tunnel diode, UJT, Varactor Diode, Photo diode, Solar cell, LED, Schottky diode.

TEXT BOOKS:

1. Boylestad, Robert L., and Louis Nashelsky. Electronic Devices and Circuit Theory. Pearson, 11th edition, 2013.
2. Millman, Jacob, and Christos C. Halkias. Electronic Devices and Circuits. Tata McGraw-Hill, 1991.
3. Bell, David A. Electronic Devices and Circuits. Oxford University Press, 5th ed., 2008

REFERENCE BOOKS:

1. Neamen, Donald A. Electronic Circuit Analysis and Design. McGraw-Hill, 2nd ed., 2001.
2. Salivahanan, S., and N. Suresh Kumar. Electronic Devices and Circuits. McGraw-Hill Education, 4th ed., 2017.
3. Razavi, Behzad. Fundamentals of Microelectronics. Wiley, 2nd ed., 2013.
4. Taur, Yuan, and Tak H. Ning. Fundamentals of Modern VLSI Devices. Cambridge University Press, 2nd ed., 2009.

DATA STRUCTURES

Department of Information Technology					I B. Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A252501	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Course Outcomes: At the end of the course, student will be able to:

1. Ability to select the data structures that efficiently model the information in a problem.
2. Understand the concepts of trees and implementing the trees with real time applications
3. Implement and know the application of algorithms for Heap and searching
4. Ability to have knowledge of graph concepts and summarize sorting techniques
5. Outline the concepts of hashing, collision and its resolution methods using hash functions.

UNIT-I

Introduction to Data Structures: Basic Terminology, Classification of Data Structures, Operation on Data Structures, abstract data types, Linear list -Introduction, singly linked list, Circular Linked Lists, Doubly Linked List, Stacks- Operations, Stack algorithm, Stack ADT, Stack applications, Queues- operations, Queue Algorithm, Queue ADT, Queue Applications.

UNIT-II

Trees: Introduction, Types of Trees, Representation of Trees, traversing a Binary Tree.
Binary Search Trees (BST): BST Operations - Searching, Insertion and Deletion, BST ADT, BST Applications, Threaded Binary Trees.

UNIT-III

Advanced Trees: AVL Trees, Red-Black Trees
Multi-way Search Trees: Introduction, B Trees, B^+ Trees – properties, B Trees Vs B^+ Trees.
Heaps: Binary Heaps, (Min and Max heap).
Searching: Introduction, Interpolation Search, Jump search.

UNIT-IV

Graphs: Introduction, Directed Graphs, Representation of Graphs, Graph Traversal Algorithms, Graph ADT, Applications of Graphs.
Sorting: Radix Sort, Heap Sort.

UNIT-V

Hashing and Collision: Introduction, Hash Tables, Hash Functions, Different Hash Functions - Division Method, Multiplication Method, Mid-square Method, Folding Method; **Collisions:** Collision Resolution by Open Addressing, Collision Resolution by Chaining.

TEXT BOOKS:

1. Data Structures: A Pseudocode Approach with C, 2nd Edition, R.F.Gilberg and B.A.Forouzan, Cengage Learning
2. Data Structure using C– Reema Thareja, 3rd Edition, Oxford University Press.

REFERENCE BOOKS:

1. Data Structures using C – A. S.Tanenbaum, Y. Langsam, and M.J. Augenstein, PHI/Pearson Education.

ENGINEERING CHEMISTRY LAB

Department of Information Technology					I B. Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A252083	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Objectives:

1. Students will understand and perform experiments based on core chemical principles relevant to engineering applications.
2. Students will learn to estimate the hardness of water to assess its suitability for drinking purposes.
3. Students will acquire the ability to perform acid-base titrations using instrumental methods such as conductometry and potentiometry.
4. Students will gain hands-on experience in synthesizing polymers like Thiokol rubber in the laboratory.

Course Outcomes:

1. Determine water quality parameters and lubricant properties through volumetric and viscometric methods, and analyze their relevance in industrial applications.
2. Apply electrochemical techniques such as conductometry and potentiometry to estimate concentrations of acids and assess chemical reactions.
3. Examine the corrosion behavior of metals under various conditions and evaluate the effect of inhibitors on corrosion rate.
4. Illustrate the synthesis and applications of advanced materials like Thiokol rubber, fuel cells, and smart biomedical materials using real and virtual lab tools.

List of Experiments:

1. Volumetric Analysis: Estimation of Hardness of water by EDTA Complexometry method.
2. Determination of rate of corrosion of mild steel in the presence and absence of inhibitor.
3. Estimation of the concentration of strong acid by Conductometry.
4. Estimation of the concentration of strong and weak acid in an acid mixture by Conductometry.
5. Estimation of concentration of strong acid with strong base by Potentiometry using quinhydrone
6. Preparation of Thiokol rubber.
7. Estimation of acid value of given lubricant oil.
8. Estimation of viscosity of lubricant oil using Ostwald's Viscometer.
9. Construction of Fuel cell and it's working. (Virtual lab)
10. Smart materials for Biomedical applications (Virtual lab)

REFERENCE BOOKS:

1. Lab manual for Engineering chemistry by B. Ramadevi and P. Aparna, S Chand Publications, New Delhi (2022)
2. Vogel's text book of practical organic chemistry 5th edition
3. Inorganic Quantitative analysis by A.I. Vogel, ELBS Publications.
4. College Practical Chemistry by V.K. Ahluwalia, Narosa Publications Ltd. New Delhi (2007).

DATA STRUCTURES LAB

Department of Information Technology					I B. Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A252581	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Outcomes: At the end of the course, student will be able to:

1. Develop C programs to perform various operations on linked list.
2. Implement stacks and queues using static and dynamic representation.
3. Implement different sorting algorithms.
4. Design binary trees and its operations.
5. Develop programs on graph traversal and hashing techniques

List of Experiments

1. Write a program that uses functions to perform the following operations on a singly linked list:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
2. Write a program that uses functions to perform the following operations on a doubly linked list:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
3. Write a program that uses functions to perform the following operations on a circular linked list:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
4. Write a program that implements a stack (its operations) using:
   i) Arrays ii) Linked Lists
5. Write a program that implements a queue (its operations) using:
   i) Arrays ii) Linked Lists
6. Write a program that implements the following sorting methods to sort a given list of integers in ascending order:
   i) Radix Sort ii) Heap Sort
7. Write a program to implement the tree traversal methods (Recursive and Non-Recursive).
8. Write a program to implement:
   i) Binary Search tree ii) AVL Trees
9. Write a program to implement the graph traversal methods.
10. Write a program to implement the following Hash Functions:
    i) Division Method ii) Multiplication Method

TEXT BOOKS:

1. Fundamentals of Data Structures in C, 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson Freed, Universities Press.
2. Data Structures using C – A. S. Tanenbaum, Y. Langsam, and M. J. Augenstein, PHI/Pearson Education.

REFERENCE BOOKS:

1. Data Structures: A Pseudocode Approach with C, 2nd Edition, R. F. Gilberg and B. A. Forouzan, Cengage Learning.

ENGLISH LANGUAGE AND COMMUNICATION SKILLS LAB

Department of Information Technology					I B. Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A252081	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Objectives:

Listening Skills:

1. To enable students, develop their active listening kills
2. To equip students with necessary training in listening, so that they can comprehend the speech of people from different linguistic backgrounds

Speaking Skills:

1. To improve their pronunciation and neutralize accent
2. To enable students express themselves fluently and appropriately
3. To practice speaking in social and professional contexts

Course Outcomes: Students will be able to:

1. Listen actively and identify important information in spoken texts
2. Interpret the speech and infer the intention of the speaker
3. Improve their accent for intelligibility
4. Speak fluently with clarity and confidence
5. Use the language in real life situations

Exercise-I

CALL Lab:

Instruction: Speech Sounds- Listening Skill-Importance – Purpose- Types-Barriers- Active Listening
Practice: Listening to Distinguish Speech Sounds (Minimal Pairs)-*Testing Exercises*

ICS Lab:

- **Diagnostic Test–Activity titled '*Express Your View*'**

Instruction: Spoken and Written language - Formal and Informal English - Greetings - Introducing Oneself and Others
Practice: Any Ice-Breaking Activity

Exercise–II

CALL Lab:

Instruction: Listening vs. Hearing-Barriers to Listening

Practice: Listening for General Information - Multiple Choice Questions - *Listening*

ICS Lab:

Instruction: Features of Good Conversation–Strategies for Effective Communication
Practice: Role Play Activity-Situational Dialogues –Expressions used in Various Situations – Making Requests and Seeking Permissions—Taking Leave- Telephone Etiquette

Exercise-III

CALL Lab:

Instruction: Errors in Pronunciation– Tips for Neutralizing Mother Tongue Influence(MTI)
Practice: Differences between British and American Pronunciation –*Listening Comprehension Exercises*

ICS Lab:

Instruction: Describing Objects, Situations, Places, People and Events
Practice: Picture Description Activity– Looking at a Picture and Describing Objects, Situations, Places, People and Events

Exercise–IV

CALL Lab:

Instruction: Techniques for *Effective* Listening

Practice: Listening for Specific Details-Listening-Gap Fill Exercises-*Listening Comprehension Exercises*

ICS Lab:

Instruction: How to Tell a Good Story –Story Star-Sequencing-Creativity
Practice: Activity on Telling and Retelling Stories-Collage

Exercise–V

CALL Lab:

Instruction: Identifying the literal and implied meaning
Practice: Listening for Evaluation –Write the Summary–Listening Comprehension Exercises

ICS Lab:
Instruction: Understanding Non-Verbal Communication
Practice: Silent Speech-Dumb Charades Activity

❖ **Post-Assessment Test on 'Express Your View'**

Suggested Software:

- Cambridge Advanced Learners 'English Dictionary with CD.
- Grammar Made Easy by Darling Kindersley.
- Punctuation Made Easy by Darling Kindersley.
- Oxford Advanced Learner's Compass, 10th Edition.
- English in Mind (Series1-4), Herbert Puchta and Jeff Stranks with Meredith Levy, Cambridge.
- English Pronunciation in Use (Elementary, Intermediate, Advanced) Cambridge University Press.
- English Vocabulary in Use (Elementary, Intermediate, Advanced) Cambridge University Press.
- TOEFL & GRE (KAPLAN, AARCO & BARRONS, USA, Cracking GRE by CLIFFS).

REFERENCE BOOKS:

1. Shobha, KN & Rayen, J. Lourdes. (2019). Communicative English – A workbook. Cambridge University Press
2. Board of Editors.(2016). ELCS Lab Manual: A Work book for CALL and ICS Lab Activities. Orient Black Swan Pvt. Ltd.
3. Mishra, Veerendra et al. (2020). English Language Skills: A Practical Approach. Cambridge University Press
4. (2022). English Language Communication Skills– Lab Manual cum Workbook. Cengage Learning India Pvt. Ltd.
5. Ur, Pennyand Wright, Andrew. 2022. Five Minute Activities – *A Resource Book for Language Teachers*. Cambridge University Press.

ENGINEERING WORKSHOP

Department of Information Technology					I B. Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A252381	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Outcomes: At the end of the course, the student will be able to:

1. Understanding the tools and methods of using to fabricate Engineering Components.
2. Applying the methods to read and interpret work shop drawings.
3. Evaluating various methods and trades of workshop in the component building.

.

1. TRADESFOREXERCISES:

At least two exercises from each trade:

i. Carpentry: T-Lap Joint, Dovetail Joint, Mortise and Tenon Joint
ii. Fitting: V-Fit, Dovetail Fit and Semi-circular fit
iii. Tin Smithy: Square Tin, Rectangular Tray and Conical Funnel
iv. Foundry: Preparation of Green S and Mould using Single Piece and Split Pattern
v. Welding Practice: Arc Welding and Gas Welding
vi. House wiring: Parallel and Series, Two-way Switch and Tube Light
Lathe: Facing and Step Turning

2. TRADESFORDEMONSTRATIONANDEXPOSURE:

Plumbing, Machine Shop, Power tools in construction and Wood Working

TEXTBOOKS:

1. Elements of Workshop Technology, Hajra Choudhury S. K., Hajra Choudhury A. K. and Nirjhar Roy S.K., Vol.I 2008andVol.II2010, Media promoters and publishers private limited, Mumbai.
2. Workshop Practice Manual, K. Venkata Reddy, BS Publication,6th Edition, Rpt.2025.

REFERENCE BOOKS:

1. Workshop Manual, K. Venugopal, Anuradha Publications,2012th edition,2012

PYTHON PROGRAMMING LAB

Department of Information Technology					I B. Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A252583	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Outcomes: At the end of the course, student will be able to:

1. Develop application-specific codes using Python.
2. Understand strings, lists, tuples, and dictionaries in Python.
3. Write programs using functions and modules.
4. Illustrate file I/O operations, and Exception Handling.
5. Apply different data visualization using Python standard libraries.

Note: The lab experiments will be like the following experiment examples.

List of Experiments

1. I. Use a web browser to go to the Python website http://python.org. This page contains information about Python and links to Python-related pages, and it gives you the ability to search the Python documentation.

   II. Start the Python interpreter and type help() to start the online help utility.
2. Start a Python interpreter and use it as a calculator.
3. Write a program to calculate compound interest when principal, rate and number of periods are given.
4. Read the name, address, email and phone number of a person through the keyboard and print the details.
5. Print the below triangle using a for loop:
   5
   44
   333
   2222
   11111
6. Write a program to check whether the given input is a digit or a lowercase character or an uppercase character or a special character (use 'if-else-if' ladder).
7. Python program to print all prime numbers in a given interval (use break).
8. Write a program to convert a list and a tuple into arrays.
9. Write a program to find common values between two arrays.
10. Write a function called palindrome that takes a string argument and returns True if it is a palindrome and False otherwise. Remember that you can use the built-in function len() to check the length of a string.
11. Write a function called is _sorted that takes a list as a parameter and returns True if the list is sorted in ascending order and False otherwise.
12. Write a function called has_ duplicates that takes a list and returns True if there is any element that appears more than once. It should not modify the original list.
13. Write a function called remove_ duplicates that takes a list and returns a new list with only the unique elements from the original. Hint: they don't have to be in the same order.

14. The word list provided words.txt doesn't contain single letter words. So you might want to add "I", "a", and the empty string.
15. Write a python code to read dictionary values from the user. Construct a function to invert its content i.e., keys should be values and values should be keys.
16. Add a comma between the characters. If the given word is 'Apple', it should become 'A,p,p,l,e'
17. Remove the given word in all the places in a string ?
18. Write a function that takes a sentence as an input parameter and replaces the first letter of every word with the corresponding upper case letter and the rest of the letters in the word by corresponding letters in lowercase without using a built-in function ?
19. Write a recursive function that generates all binary strings of n-bit length.
20. Write a python program that defines a matrix and prints it.
21. Write a python program to perform multiplication of two square matrices.
22. How do you make a module? Give an example of construction of a module using different geometrical shapes and operations on them as its functions.
23. Use the structure of exception handling in all general-purpose exceptions.
24. Write a python code to read a phone number and an email id from the user and validate it for correctness.
25. Write a Python code to merge two given file contents into a third file.
26. Write a Python code to open a given file and construct a function to check for given words present in it and display if found.
27. Write a Python code to read text from a text file, find the word with the most number of occurrences.
28. Write a function that reads a file and displays the number of words, number of vowels, blank spaces, lowercase letters and uppercase letters.
29. Import numpy, Plotpy and Scipy and explore their functionalities.
30. Install NumPy package with pip and explore it.
31. Write a program to implement Digital Logic Gates-AND, OR, NOT, EX-OR.
32. Write a GUI program to create a window with two text labels, two text fields and two buttons as Submit and Reset.

TEXT BOOKS:

1. Super charged Python: Take your code to the next level, Overland
2. Learning Python, MarkLutz, O'reilly

REFERENCE BOOKS:

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson
2. Python Programming: A Modular Approach with Graphics, Database, Mobile and Web Applications, Sheetal Taneja, Naveen Kumar, Pearson
3. Introduction to Python Programming, Gowrishankar S, Veena A, CRC Press
4. Programming with Python: A User's Book, Michael Dawson, Cengage Learning, India Edition
5. Python for Data Science, Dr. Mohd Abdul Hameed, Wiley publications
6. Core Python Programming, Dr. R. Nageswara Rao, Dreamtech press
7. Introduction to Python, Gowrishankar S, Veena A, CRC Press

IT WORKSHOP

Department of Information Technology				I B. Tech II Semester		
Course Code	Hours/Week		Credits	Marks		
A252585	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Outcomes: At the end of the course, student will be able to:

1. Perform Hardware troubleshooting
2. Understand Hardware components and interdependencies
3. Document/Presentation preparation
4. Perform calculations using spreadsheets
5. Query successful

PC Hardware

Task 1: Identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral and submit it to your instructor.

Task 2: Every student should disassemble and assemble the PC back to working condition. Lab instructors should verify the work and follow it up with a Viva. Also, students need to go through the video which shows the process of assembling a PC. A video would be given as part of the course content.

Task 3: Every student should individually install MS windows on the personal computer. Lab instructor should verify the installation and follow it up with a Viva.

Task 4: Every student should install Linux on the computer. This computer should have windows installed. The system should be configured as a dual-boot with both Windows and Linux. Lab instructors should verify the installation and follow it up with a Viva.

Internet & World Wide Web

Task 1: Orientation & Connectivity Boot Camp: Students should get connected to their Local Area Network and access the Internet. In the process, they configure the TCP/IP setting. Finally, students should demonstrate to the instructor how to access websites and email. If there is no internet connectivity, preparations need to be made by the instructors to simulate the WWW on the LAN.

Task 2: Search Engines & Netiquette: Students should know what search engines are and how to use them. A few topics would be given to the students for which they need to search on Google. This should be demonstrated to the instructors by the student.

LaTeX and WORD

Task 1: Word Orientation: The mentor needs to give an overview of LaTeX and Microsoft (MS) office or equivalent (FOSS) tools. The importance of LaTeX and MS Office or equivalent (FOSS) tools as word processors, details of the four tasks and features that would be covered in each, using LaTeX and word - accessing, overview of toolbars, saving files, using help and resources, rulers, format painter in word.

Task 2: Using LaTeX and Word to create a project certificate. Features to be covered: - Formatting Fonts in word, Drop Cap in word, Applying Text effects, Using Character Spacing, Borders and Colors, Inserting Header and Footer, Using Date and Time option in both LaTeX and Word.

Task 3: Creating project abstract Features to be covered: - Formatting Styles, Inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.

Task 4: Creating a Newsletter. Features to be covered: - Table of Content, Newspaper columns, Images from files and clip art, Drawing toolbar and Word Art, Formatting Images, Text boxes, Paragraphs and Mail Merge in word.

Excel

Excel Orientation: The mentor needs to tell the importance of MS office or equivalent (FOSS) tool Excel as a Spreadsheet tool, give the details of the four tasks and features that would be covered in each. Using Excel - Accessing, overview of toolbars, saving excel files, Using help and resources.

Task 1: Creating a Scheduler - Features to be covered: Gridlines, Format Cells, Summation, autofill, Formatting Text.

Task 2: Calculating GPA - Features to be covered: - Cell Referencing, Formulae in excel - average, std. deviation, Charts, Renaming and Inserting worksheets, Hyper linking, Count function, LOOKUP/VLOOKUP.

Task 3: Split cells, freeze panes, group and outline, Sorting, Boolean and logical operators, Conditional formatting.

Power Point

Task 1: Students will be working on basic PowerPoint utilities and tools which help them create basic PowerPoint presentations. PPT Orientation, Slide Layouts, Inserting Text, Word Art, Formatting Text, Bullets and Numbering, Auto Shapes, Lines and Arrows in PowerPoint.

Task 2: Interactive presentations - Hyperlinks, Inserting Images, Clip Art, Audio, Video, Objects, Tables and Charts.

Task 3: Master Layouts (slide, template, and notes), Types of views (basic, presentation, slide sorter, notes etc) and Inserting Background textures, Design Templates, Hidden slides.

REFERENCE BOOKS:

1. Comdex Information Technology Course tool kit, Vikas Gupta, WILEY Dreamtech.
2. The Complete Computer upgrade and repair book, 3rd edition, Cheryl A. Schmidt, WILEY Dreamtech.
3. Introduction to Information Technology, ITL Education Solutions limited, Pearson Education.
4. PC Hardware - A Handbook, Kate J. Chase, PHI (Microsoft).
5. LaTeX Companion, Leslie Lamport, PHI/Pearson.
6. IT Essentials PC Hardware and Software Companion Guide, Third Edition by David Anfinson and Ken Quamme - CISCO Press, Pearson Education.
7. IT Essentials PC Hardware and Software Labs and Study Guide, Third Edition by Patrick Regan - CISCO Press, Pearson Education.

PROBABILITY AND STATISTICS

Department of Information Technology					II B.Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A253001	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Pre-requisites: Mathematics courses of first year of study.

Course Objectives:

1. To learn random variables that describe randomness or an uncertainty in certain realistic situation
2. To learn Binomial geometric and normal distributions
3. To learn Sampling distribution of mean, variance, point estimation and interval estimation and the testing of hypothesis and ANOVA
4. By using correlation to identify the strength and direction of a linear relationship between two variables and using regression to predict how much a dependent variable changes based on adjustments to an independent variable, you are empowered to make objective, data-driven decisions regarding your processes.
5. To fit a desired curve by the method of least squares for the given data

Course Outcomes: After learning the contents of this course the students must be able to:

1. To differentiate among random variables involved in the probability models which are useful for all branches of engineering.
2. Derive relationship among variety of performance measures using probability distributions.
3. Acquire elementary knowledge of parametric and non parametric –tests and understand the use of observing state analysis for predicting future conditions.
4. Identify and examine situations that generate using problems and able to solve the tests of ANOVA for classified data.
5. Apply proper measurements, Indicators and techniques of Correlation and regression analysis.

UNIT-I: PROBABILITY AND RANDOM VARIABLES:

PROBABILITY: Introduction- Sample space and events- probability. The axioms of probability- some elementary theorems-conditional probability- Bayee's Theorem-

RANDOM VARIABLES: Random variable, Discrete and continuous. Probability distributions, mass function/ density function of a probability distribution.

UNIT-II: PROBABILITY DISTRIBUTIONS:

Fitting of Binomial, Poisson & Normal distributions and their properties (only Statements) Moment Generating Functions of the above three distributions and hence finding the mean and variance.

UNIT-III: SAMPLING THEORY & TESTING OF HYPOTHESIS I :

Sampling Distribution: Definition of Sample, Population, and Types of Sampling. Estimation - Point estimation, Interval estimation.

Testing of Hypothesis: Null hypothesis – Alternative hypothesis, Type I, & Type II errors – critical region confidence interval for mean, Testing of hypothesis for single mean and difference between the means for large samples. Confidence interval for the proportions, Tests of hypothesis for the proportions- single and difference between the proportions for large samples

UNIT-IV: TESTING OF HYPOTHESIS II:

Small Samples - t-distribution, F-Distribution, χ^2 distribution, ANOVA for one-way classified data

UNIT-V: CORRELATION, REGRESSION & CURVE FITTING:

Correlation and Regression: Coefficient of Correlation-Regression coefficients- The lines of Regression-The Coefficient of Rank Correlation.

Curve Fitting: Fitting a Straight line – Second Degree Polynomials – Exponential and Power Curve by the Method of Least Squares.

TEXT BOOKS :

1. Probability and Statistics for Engineers, by Richard Arnold Johnson, Irvin Miller and John E Freund, New Delhi Prentice Hall.
2. Probability and Statistics, by T. K. V. Iyangar others, S. Chand Publications
3. Introduction to Probability & Statistics for Engineers and Scientists by Sheldon M.Ross

References Books:

1. Fundamentals of Mathematical Statistics, by S C Guptha and V K Kapoor, S Chand.
2. Mathematics for Engineers and Scientists, by Alan Jeffrey, sixth edition, CRC press.

COMPUTER ORGANIZATION AND MICROPROCESSOR

Department of Information Technology					II B.Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A2531201	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Course Outcomes:

1. Able to understand the basic components and the design of CPU, ALU and Control Unit.
2. Understand memory hierarchy and its impact on computer cost/performance.
3. Able to understand the advantage of instruction level parallelism and pipelining for high performance Processor design.
4. Understand the instruction set, instruction formats and addressing modes of8086.
5. Able to write assembly language programs to solve problems.

UNIT - I

Digital Computers: Introduction, Block diagram of Digital Computer, Definition of Computer Organization, Computer Design and Computer Architecture.

Basic Computer Organization and Design: Instruction codes, Computer Registers, Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Input – Output and Interrupt, Complete Computer Description.

Micro Programmed Control: Control memory, Address sequencing, micro program example, design of control unit.

UNIT - II

Central Processing Unit: The 8086 Processor Architecture, Register organization, Physical memory organization, General Bus Operation, I/O Addressing Capability, Special Processor Activities, Minimum and Maximum mode system and timings.

8086 Instruction Set and Assembler Directives-Machine language instruction formats, Addressing modes, Instruction set of 8086, Assembler directives and operators.

UNIT - III

Assembly Language Programming with 8086- Machine level programs, Machine coding the programs, Programming with an assembler, Assembly Language example programs. Stack structure of 8086, Interrupts and Interrupt service routines, Interrupt cycle of 8086, Interrupt programming, Passing parameters to procedures, Macros, Timings and Delays.

UNIT - IV

Computer Arithmetic: Introduction, Addition and Subtraction, Multiplication Algorithms, Division Algorithms, Floating - point Arithmetic operations. Input-Output Organization: Peripheral Devices, Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt, Direct memory Access, Input –Output Processor (IOP), Intel 8089 IOP.

UNIT - V

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory.

Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processors.

TEXT BOOKS:

1. Computer System Architecture, M. Morris Mano, Third Edition, Pearson. (UNITS – I, IV, V)
2. Advanced Microprocessors and Peripherals, K M Bhurchandi, A. K Ray ,3rd edition, McGraw Hill India Education Private Ltd. (UNITS - II, III).

REFERENCES BOOKS:

1. Microprocessors and Interfacing, D V Hall, SSSP Rao, 3 rd edition, McGraw Hill India Education Private Ltd.
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky: Computer Organization, 5th Edition, Tata McGraw Hill.
3. Computer Organization and Architecture, William Stallings, 9th Edition, Pearson.
4. David A. Patterson, John L. Hennessy: Computer Organization and Design – The Hardware / Software Interface ARM Edition, 4th Edition, Elsevier.

OBJECT ORIENTED PROGRAMMING THROUGH JAVA

Department of Information Technology					II B.Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A253502	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Course Outcomes:

At the end of the course, student will be able to:

1. Demonstrate the behavior of programs involving the basic programming constructs like control structures, constructors, string handling and garbage collection.
2. Demonstrate the implementation of inheritance (multilevel, hierarchical and multiple) by using extend and implement keywords
3. Use multithreading concepts to develop inter process communication.
4. Understand the different types of file I/O, Exploring String, usage of Container classes
5. Develop web applications using Servlets and JDBC.

UNIT-I

Object oriented thinking and Java Basics- OOP concepts, History of Java, Java buzzwords, data types, variables, operators, expressions, control statements, arrays, type casting, simple java program, concepts of classes, objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, parameter passing, object Class and wrapper classes.

UNIT-II

Inner classes in Java: Types of inner classes, Creating static / non-static inner classes, Local and anonymous inner classes.

Inheritance, Interfaces and Packages– Types of Inheritance, benefits of inheritance, super, final keyword with inheritance, method overriding, abstract classes. Defining and implementing interfaces.

Packages: Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages.

UNIT-III

Exception handling and Multithreading-- Concepts of exception handling, benefits of exception handling, exception hierarchy, usage of try, catch, throw, throws and finally, built in exceptions, User Defined Exceptions. Differences between multithreading and multitasking, thread life cycle, creating threads, thread priorities, synchronizing threads, inter thread communication, thread groups, daemon threads.

UNIT-IV

Exploring String, String Buffer, StringBuilder classes and StringTokenizer.

Java.io package: File, Byte Streams: Input Stream, Output Streams, File Input Stream, File Output Stream, Character Streams: Reader, Writer, FileReader, FileWriter, BufferedReader, BufferedWriter, InputStreamReader.

java.util package- Collection Interfaces: List, Set, Map. Collection classes: LinkedList, TreeSet, HashMap, Enumerator, Iterator, ListIterator, Calendar, Random, Scanner, Comparator, Comparable.

UNIT-V

Java Servlets: Overview of Java Servlet, Servlet Life cycle, Request and Response methods, Servlet Configuration, Servlet Context, Approaches to Session tracking, Servlet Collaboration.

Database Connections: Introduction to JDBC, JDBC Drivers, Connecting to a Database using JDBC.

TEXT BOOKS:

1. Java the complete reference, 13th edition, Herbert schildt, Dr. Denny Coward, Mc Graw Hill.
2. Understanding OOP with Java, updated edition, T. Budd, Pearson education.

REFERENCE BOOKS:

1. An Introduction to programming and OO design using Java, J.Nino and F.A. Hosch, John Wiley & sons.
2. An Introduction to OOP, third edition, T. Budd, Pearson education.
3. Introduction to Java programming, Y. Daniel Liang, Pearson education.
4. An introduction to Java programming and object-oriented application development, R.A. Johnson Thomson.
5. Core Java 2, Vol 1, Fundamentals, Cay.S. Horstmann and Gary Cornell, eighth Edition, Pearson Education.

OPERATING SYSTEMS

Department of Information Technology					II B.Tech. I Semester		
Course Code	Hours/Week			Credits	Marks		
A253503	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Course Outcomes: At the end of the course, student will be able to:

1. Understand the fundamental concepts, structures of operating systems, and process management techniques.
2. Analyze CPU scheduling, process management, and synchronization mechanisms.
3. Explain memory management techniques and virtual memory concepts with page replacement algorithms. Explain and apply interprocess communication mechanisms and file system interfaces using appropriate system calls
4. Analyze deadlock situations and apply suitable prevention, avoidance, detection, and recovery techniques

UNIT-I

Operating System: Introduction, Structures - Simple Batch, Multiprogrammed, Time-shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, System components, Operating System services, System Calls

Process: Process concepts, Process Control block (PCB), Operations on processes, Cooperating Processes, Threads

UNIT-II

CPU Scheduling: Scheduling Criteria, Scheduling Algorithms, Multiple -Processor Scheduling, System call interface for process management-fork, exit, wait, waitpid, exec

Process Management and Synchronization: The Critical Section Problem, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Critical Regions, Monitors

UNIT-III

Memory Management and Virtual Memory: Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging, Demand Paging, Page Replacement, Page Replacement Algorithms.

UNIT-IV

Interprocess Communication Mechanisms: IPC between processes on a single computer system, IPC between processes on different systems, using pipes, FIFOs, message queues, shared memory.

File System Interface and Operations: Access methods, Directory Structure, Protection, File System Structure, Allocation methods, Free-space Management. Usage of open, create, read, write, close, lseek, stat, ioctl system calls.

UNIT-V

Deadlocks: System Model, Deadlocks Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock

TEXT BOOKS:

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley
2. Advanced programming in the UNIX environment, W.R. Stevens, Pearson education.

REFERENCE BOOKS:

1. Operating Systems- Internals and Design Principles, William Stallings, Fifth Edition–2005, Pearson Education/PHI
2. Operating System A Design Approach- Crowley, TMH.
3. Modern Operating Systems, Andrew S. Tanenbaum 2nd edition, Pearson/PHI
4. UNIX programming environment, Kernighan and Pike, PHI/ Pearson Education
5. UNIX Internals -The New Frontiers, U. Vahalia, Pearson Education.

INTRODUCTION TO IOT

Department of Information Technology					II B.Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A2531202	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Course Outcomes: Upon completing this course, the student will be able to:

1. Explain the architecture, layers, protocols, and sensor networks of IoT and their roles in real-world applications.
2. Differentiate IoT and M2M communication models and implement interoperable systems using Arduino programming.
3. Apply Python programming on Raspberry Pi to interface sensors and develop IoT use-case applications.
4. Design and implement SDN-based IoT solutions using Raspberry Pi for efficient network management.
5. Analyze and select appropriate sensor technologies and cloud storage models for specific IoT applications.

UNIT- I

Introduction to IoT and Sensor Networks

Introduction to Internet of Things (IoT), Characteristics and Applications of IoT, IoT Architecture and Reference Models (IETF, ITU-T)

Physical Design of IoT: Devices, Gateways, and Data Centers

Functional Blocks of IoT: Sensing, Actuation, Communication, Enabling Technologies: RFID, Wireless Sensor Networks, Nanotechnology

Basics of Networking and Communication Protocols: MQTT, CoAP, ZigBee, HTTP Sensor Networks: Types, Topologies, and Protocols Introduction to IoT Security and Privacy Fundamentals.

UNIT- II

Machine to Machine (M2M) and Embedded Programming for IoT

Machine-to-Machine Communications Overview, Difference between IoT and M2M, Interoperability in IoT: Standards and Protocols.

Introduction to Arduino Programming for IoT, Integration of Sensors and Actuators with Arduino. Hands-on Exercises: Sensor Data Acquisition and Actuator Control, Basic Communication Protocols Implementation on Arduino, IoT Device Interoperability Challenges and Solutions.

UNIT- III

Raspberry Pi with Python Programming for IoT

Introduction to Python Programming Basics

Overview of Raspberry Pi and Its Role in IoT, Interfacing Raspberry Pi with Sensors and Actuators (UART, SPI, I2C).

Data Acquisition and Local Processing on Raspberry Pi, Sending Data to Cloud Platforms. Implementation

of IoT Projects Using Raspberry Pi
Case Studies: Smart Home Automation, Healthcare Monitoring, Environmental Sensing

UNIT- IV

Network Management and Software Defined Networking (SDN) for IoT

Introduction to Software Defined Networking (SDN), SDN Architecture: Controller, Data Plane, Control Plane, Benefits of SDN in IoT: Flexibility, Centralized Control, Security, Traffic Management, SDN Models: Open SDN, API SDN, Overlay, Hybrid Data Handling, Analytics, and Security Considerations in SDN for IoT Network Management Challenges in IoT, Integration of SDN with IoT Networks

UNIT- V

Cloud and Edge Computing Models with IoT Use Cases

Introduction to Cloud Computing and Cloud Storage Models, Edge and Fog Computing Concepts for IoT, Web Servers and Cloud Platforms for IoT (AWS IoT, Azure IoT, etc.), Sensor-Cloud Architecture Data Analytics and Visualization Techniques for IoT Data

IoT Use Cases: Smart Cities, Smart Homes, Connected Vehicles, Smart Grid, Industrial IoT Case Studies: Agriculture, Healthcare, Activity Monitoring.

TEXT BOOKS

1. Pethuru Raj and Anupama C. Raman "The Internet of Things: Enabling Technologies, Platforms, and Use Cases", CRC Press.
2. Vijay Madisetti, Arshdeep Bahga, “Internet of Things: A Hands-On Approach”

REFERENCE BOOKS

1. Terokarvinen, kemo, karvinen and villeyvaltokari, “Make sensors”: 1st edition,2014, Maker media.
2. Waltenegus Dargie, Christian Poellabauer, Fundamentals of Wireless Sensor Networks: Theory and Practice.
3. Charles Bell, Beginning Sensor networks with Arduino and Raspberry Pi, 2013, Apress.
4. Fei Hu, Security and Privacy in Internet of Things (IoTs), CRC Press, Taylor & Francis Group.
5. S. Sahoo, S. Sahoo, S. Mishra, Software-Defined Networking for Future Internet Technology: Concepts and Applications, Routledge.
6. Honbo Zhou, The Internet of Things in the Cloud: A Middleware Perspective. 2012, CRC Press (Taylor & Francis).

COMPUTATIONAL MATHEMATICS LAB

(Using Python/MATLAB software)

Department of Information Technology					II B.Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A253081	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Outcomes: After learning the contents of this paper, the student must be able to

1. **Apply numerical and computational techniques** to develop Python/MATLAB programs for computing eigenvalues and eigenvectors of matrices.
2. **Formulate and solve algebraic and transcendental equations** using suitable numerical methods, and **implement solutions** for systems of linear equations using Python/MATLAB.
3. **Develop computational models** to solve first-order ordinary differential equations using appropriate analytical and numerical methods in Python/MATLAB.
4. **Analyze and solve higher-order linear differential equations with constant coefficients** using Python/MATLAB for engineering and scientific applications.

* **Visualize all solutions Graphically through programmes**

UNIT – I: Eigen values and Eigenvectors**:** **6P**

Programs:

- Finding real and complex Eigen values.
- Finding Eigen vectors.

UNIT-II: Solution of Algebraic and Transcendental Equations **6P**

Bisection method, Newton Raphson Method

Programs:

- Root of a given equation using Bisection method.
- Root of a given equation Newton Raphson Method.

UNIT-III: Linear system of equations: 6P

Jacobi‟s iteration method and Gauss-Seidal iteration method

Programs:

- Solution of given system of linear equations using Jacobi‟s method
- Solution of given system of linear equations using Gauss-Seidal method

UNIT-IV: First-Order ODEs 8P

Exact and non-exact equations, Applications: exponential growth/decay, Newton's law of cooling.

Programs:

- Solving exact and non-exact equations
- Solving exponential growth/decay and Newton's law of cooling problems

UNIT-V: Higher order linear differential equations with constant coefficients **6P**

Programs:

- Solving homogeneous ODEs
- Solving non-homogeneous ODEs

TEXT BOOKS:

1. MATLAB and its Applications in Engineering, Rajkumar Basal, Ashok Kumar Geo, Manoj Kumar Sharma, Pearson publication.
2. Kenneth A. Lambert, The fundamentals of Python: First Programs, 2011, Cengage Learnings.
3. Think Python First Edition, by Allen B. Downey, Orielly publishing.
4. Introduction to Python Programming, William Mitchell, Povel Solin, Martin Novak et al., NCLab Public Computing, 2012.
5. Introduction to Python Programming, ©Jacob Fredslund, 2007.

REFERENCE BOOKS:

1. An Introduction to Python, John C. Lusth, The University of Alabama, 2011.
2. Introduction to Python, ©Dave Kuhlman, 2008.

OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB

Department of Information Technology					II B.Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A253582	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Outcomes: At the end of the course, student will be able to:

1. **Apply object-oriented programming principles** to design and develop Java programs for real-world problem solving.
2. **Implement advanced OOP features** such as abstract classes, interfaces, and inheritance in Java applications.
3. **Develop multithreaded Java programs** and **use the Java Collections Framework** for efficient data handling and concurrent execution.
4. **Design and implement basic web applications** using Java Servlets and JDBC for database connectivity.

List of Experiments:

1. Write a Java Program to accept Student Name and Roll Number as command Line Arguments and Display the output.
2. Write a Java Program that prompts the user for an integer and then prints out all prime numbers up to that integer.
3. Write a Java Program that uses both recursive and non recursive functions to print the nth value in the Fibonacci sequence.
4. Write a Java Program to create an Employee class to read and display the data (emp_id, emp_name, Department, salary, experience) using constructor and method.
5. Write a Java Program to illustrate method and constructor overloading.
6. Write a Java Program to multiply two given matrices by passing objects as parameters.
7. Write a Java Program to illustrate different types of inheritances.
8. Write a java Program to illustrate Method Overriding.
9. Write a Java Program to create an abstract class named Shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.
10. Write a Java Program to illustrate the use of Creating and Importing Packages.
11. Write a Java Program to handle Multiple Exceptions.
12. Write a Java Program to create User Defined Exceptions.
13. Write a Java Program that implements a multi-thread application that has three threads. The first thread generates a random integer every 1 second and if the value is even, the second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of the cube of the number.
14. Write a Java Program that implements the Producer – Consumer Problem using the concept of Inter Thread Communication.

15. Write a Java Program to list all the files in a directory including the files present in all its subdirectories.
16. Write a Java Program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record is separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint: use hash tables).
17. Write a Java Program that reads a line of integers, and then displays each integer, and the sum of all the integers
18. Write a Java Program to make frequency count of words in a given text
19. Write a Java Program to establish Database connection and execute queries using JDBC.
20. Installation and Configuration of Tomcat and deploy a simple "Hello World" Servlet.

TEXTBOOKS:

1. Java for Programmers, P. J. Deitel and H. M. Deitel, 10th Edition Pearson education.
2. Thinking in Java, Bruce Eckel, Pearson Education.

REFERENCE BOOKS:

1. Java Programming, D. S. Malik and P. S. Nair, Cengage Learning.
2. Core Java, Volume 1, 9th edition, Cay S. Horstmann and G Cornell, Pearson.

OPERATING SYSTEMS LAB

Department of Information Technology					II B.Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A253583	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Outcomes: At the end of the course, student will be able to:

1. **Simulate and analyze CPU scheduling algorithms** and **evaluate their performance** using standard scheduling criteria.
2. **Apply UNIX/Linux system calls** to implement file handling, process creation, and inter-process communication.
3. **Implement synchronization and deadlock handling solutions**, including Producer–Consumer using semaphores and Banker's Algorithm for deadlock avoidance.
4. **Simulate and analyze memory management techniques**, including paging and page replacement policies, to assess efficient memory utilization.

LIST OF EXPERIMENTS:

1. Write C programs to simulate the following CPU Scheduling algorithms
   a) FCFS b) SJF c) Round Robin d) priority
2. Write programs using the I/O system calls of UNIX/LINUX operating system (open, read, write, close, lseek, stat, fork, exit)
3. Write a C program to implement the Producer – Consumer problem using semaphores using UNIX/LINUX system calls.
4. Write C programs to simulate the following memory management techniques
   a) Paging b) Segmentation
5. Write C programs to simulate Page replacement policies
   a) FCFS b) LRU c) Optimal
6. Write C programs to illustrate the following IPC mechanisms
   a) Pipes b) FIFOs c) Message Queues d) Shared Memory
7. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance.

TEXT BOOKS:

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley.
2. Advanced programming in the Unix environment, W. R. Stevens, Pearson education.

REFERENCE BOOKS:

1. Operating Systems – Internals and Design Principles, William Stallings, Fifth Edition– 2005, Pearson Education/PHI.
2. Operating System - A Design Approach-Crowley, TMH.
3. Modern Operating Systems, Andrew S Tanenbaum, 2nd edition, Pearson/PHI.
4. UNIX Programming Environment, Kernighan and Pike, PHI/Pearson Education.
5. UNIX Internals: The New Frontiers, U. Vahalia, Pearson Education.

INTERNET OF THINGS LAB

Department of Information Technology					II B.Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A2531281	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Outcomes:

1. Illustrate various IoT devices, sensors, and actuators, including Arduino, Raspberry Pi, and microcontrollers.
2. Identify sensors technologies used in IoT applications and their working principles.
3. Apply programming languages like Python, C/C++ in IoT applications.
4. Develop skills in collecting data from sensors and processing it using microcontrollers or edge computing devices.

List of Programs

1. Installing OS on Raspberry Pi
   a) Installation using Pi Imager
   b) Installation using image file
   - Downloading an Image
   - Writing the image to an SD card using Linux using Windows
   - Booting up Follow the instructions given in the URL https://www.raspberrypi.com/documentation/computers/gettingstarte d.html

2. Accessing GPIO pins using Python
   a. Installing GPIO Zero library.
      First, update your repositories list:
      sudo apt update
      Then install the package for Python 3:
      sudo apt install python3-gpiozero
   b. Blinking an LED connected to one of the GPIO pin
   c. Adjusting the brightness of an LED Adjust the brightness of an LED (0 to 100, where 100 means maximum brightness) using the in-built PWM wavelength.

3. Using Raspberry Pi
   a. Calculate the distance using a distance sensor.
   b. Basic LED functionality using switch.

4. Using Arduino
   a. Calculate the distance using a distance sensor.
   b. Basic LED functionality using switch.
   c. Calculate temperature using a temperature sensor

5. Using Node MCU
   a. Including required libraries of Node MCU
   b. Calculate the distance using a distance sensor.
   c. Basic LED functionality using switch.
   d. Calculate temperature using a temperature sensor

6. Using ESP32
   a. Including required libraries of ESP32.
   b. Calculate the distance using a distance sensor.
   c. Basic LED functionality using switch.
   d. Calculate temperature using a temperature sensor.

7. Collecting Sensor Data
   a. DHT Sensor interface
   - Connect the terminals of DHT GPIO pins of Raspberry Pi.
   - Import the DHT library using import Adafruit, DHT.
   - Read sensor data and display it on screen.
   - Read sensor data and display it on the Cloud platform using Thing Speak (API).
   - Read sensor data and display it through the Blynk App.

8. Capstone Project: End-to-End IoT Solution Integrating Sensors, Network, and Cloud.

TEXT BOOKS:

1. Arshdeep Bahga and Vijay Madisetti, Internet of Things - A Hands-on Approach, 2015, Universities Press.
2. Matt Richardson & Shawn Wallace, Getting Started with Raspberry Pi, 2014, O‟Reilly.

REFERENCE BOOKS:

1. Bernd Scholz-Reiter, Florian Michahelles, “Architecting the Internet of Things”, ISBN 978-3- 642-19156-5 e-ISBN 978-3-642-19157-2, Springer, 2016.
2. N. Ida, Sensors, Actuators and Their Interfaces, Scitech Publishers, 2014.

DATA VISUALIZATION - POWER BI/ TABLEAU

Department of Information Technology					II B.Tech I Semester		
Course Code	Hours/Week			Credits	Marks		
A2531282	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Outcomes:

1. Understand how to import data into Tableau.
2. Understand concepts of Dimensions and Measures.
3. Develop Programs and understand how to map Visual Layouts and Graphical Properties
4. Create Dashboard, custom charts, and, publish to tableau online for any real time dataset

List of Programs

1. Understanding Data, what is data, where to find data, Foundations for building Data Visualizations, Creating Your First visualization.
2. Getting started with Tableau Software using Data file formats, connecting your Data to Tableau, creating basic charts (line, bar charts, Tree maps), Using the Show me panel.
3. Tableau Calculations, Overview of SUM, AVR, and Aggregate features, Creating custom calculations and fields.
4. Applying new data calculations to your visualizations, Formatting Visualizations, Formatting Tools and Menus, Formatting specific parts of the view.
5. Editing and Formatting Axes, Manipulating Data in Tableau data, Pivoting Tableau data
6. Structuring your data, Sorting and filtering Tableau data, Pivoting Tableau data.
7. Advanced Visualization Tools: Using Filters, Using the Detail panel, using the Size panels, customizing filters, Using and Customizing tooltips, Formatting your data with colors.
8. Creating Dashboards & Storytelling, creating your first dashboard and Story, Design for different displays, adding interactivity to your Dashboard, Distributing & Publishing your Visualization.
9. Tableau file types, publishing to Tableau Online, sharing your visualizations, printing, and exporting.
10. Creating custom charts, cyclical data and circular area charts, Dual Axis charts.
11. Visualize various data patterns taking any dataset from Kaggle.
12. Visualize data patterns using Google Charts by creating interactive Line, Bar, and Pie charts along with Dashboards using HTML and JavaScript.

REFERENCE BOOKS:

1. Brett Powell, Microsoft Power BI cookbook, 2nd edition.
2. Roger D. Peng, R Programming for Data Science
3. Norman Matloff Cengage Learning India, The Art of R Programming.

WEB RESOURCE:

1. https://developers.google.com/chart/interactive/docs.

DISCRETE MATHEMATICS

Department of Information Technology					II B.Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A254004	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Course Outcomes: At the end of the course, student will be able to:

1. Understand and construct precise mathematical proofs.
2. Apply logic and set theory to formulate precise statements.
3. Analyze and solve counting problems on finite and discrete structures.
4. Describe and manipulate sequences.
5. Apply graph theory in solving computing problems

UNIT-I

Mathematical logic: Introduction, Statements and Notation, Connectives & Truth Tables, Normal Forms (DNF, CNF, PDNF, PCNF).

UNIT-II

Theory of Inference for the Statement Calculus, The Predicate Calculus, Inference Theory of the Predicate Calculus.

Set theory: Introduction, Basic Concepts of Set Theory,

Relations: Special Properties of Binary Relations, Partially ordered relations, Hasse diagrams, Lattices.

UNIT-III

Functions: Composition of functions Inverse Functions

Algebraic Structures: Algebraic Systems-Examples and General Properties, Semi groups and Monoids, groups, sub groups, homomorphism, Isomorphism.

UNIT-IV

Elementary Combinatorics: Basics of Counting, Combinations and Permutations, Enumeration of Combinations and Permutations, Enumerating Combinations and Permutations with Repetitions, Enumerating Permutation with Constrained Repetitions, Binomial Coefficient, The Binomial and Multinomial Theorems, The Principle of Exclusion.

UNIT-V

Graph Theory: Basic Concepts, Isomorphism and Subgraphs, Trees and their Properties, Spanning Trees, Directed Trees, Binary Trees, Planar Graphs, Euler's Formula, Multi-graphs and Euler Circuits, Hamiltonian Graphs, Chromatic Numbers, The Four-Color Problem.

TEXT BOOKS:

1. Discrete Mathematical Structures with Applications to Computer Science: J.P. Tremblay, R. Manohar, McGraw-Hill, 1st ed.
2. Discrete Mathematics for Computer Scientists & Mathematicians: Joe l. Mott, Abraham Kandel, Teodore P. Baker, Prentis Hall of India, 2nd ed.

REFERENCE BOOKS:

1. Discrete and Combinatorial Mathematics - an applied introduction: Ralph. P. Grimald, Pearson education, 5th edition.
2. Discrete Mathematical Structures: Thomas Kosy, Tata McGraw Hill Publishing co.

DATA COMMUNICATIONS AND COMPUTER NETWORKS

Department of Information Technology					II B.Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A2541201	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Course Outcomes:

1. **Understand the fundamentals of data communication and computer networks**, including network types, topologies, and transmission media.
2. **Describe the functions and services of the Data Link Layer**, including framing, error detection/correction, and flow control mechanisms.
3. **Analyze Network Layer concepts** such as routing, addressing, and congestion control in packet-switched networks.
4. **Understand Transport Layer mechanisms** including port addressing, flow control, error control, and congestion management.
5. **Illustrate the working principles of Application Layer protocols** such as HTTP, FTP, SMTP, and DNS in real-world networking scenarios.

UNIT – I

Data Communications: Components – Direction of Data flow – Networks – Components and Categories – Types of Connections – Topologies –Protocols and Standards – ISO / OSI model, Example Networks such as ATM, Frame Relay, ISDN **Physical layer:** Transmission modes, Multiplexing, Transmission Media, Switching, Circuit Switched Networks, Datagram Networks, Virtual Circuit Networks.

UNIT – II

Data link layer: Introduction, Framing, and Error – Detection and Correction – Parity – LRC – CRC Hamming code, Flow and Error Control, Noiseless Channels, Noisy Channels, HDLC, Point to Point Protocols. 111 Medium Access sub layer: ALOHA, CSMA/CD, LAN – Ethernet IEEE 802.3, IEEE 802.5 – IEEE 802.11, Random access, Controlled access, Channelization.

UNIT – III

Network layer: Logical Addressing, Internetworking, Tunneling, Address mapping, ICMP, IGMP, Forwarding, Uni-Cast Routing Protocols, Multicast Routing Protocols.

UNIT – IV

Transport Layer: Process to Process Delivery, UDP and TCP protocols, Data Traffic, Congestion, Congestion Control, QoS, Integrated Services, Differentiated Services, QoS in Switched Networks.

UNIT – V

Application Layer: Domain name space, DNS in Internet, Electronic Mail, SMTP, FTP, WWW, HTTP, SNMP.

TEXT BOOKS:

1. Data Communications and Networking, Behrouz A. Forouzan, Fourth Edition TMH.

REFERENCE BOOKS:

1. Computer Networks, Andrew S Tanenbaum, 6th Edition. Pearson Education.
2. Computer Networking: A Top-Down Approach Featuring the Internet. James F. Kurose & Keith W. Ross, 3 rd Edition, Pearson Education
3. Data communications and Computer Networks, P.C Gupta, PHI.
4. An Engineering Approach to Computer Networks, S. Keshav, 2nd Edition, Pearson Education.

FORMAL LANGUAGES AND AUTOMATA THEORY

Department of Information Technology					II B.Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A2541202	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Course Outcomes

1. **Explain the concept of abstract machines** and their computational power in recognizing formal languages.
2. **Apply finite state machines** (DFA, NFA) and regular expressions to model and solve computing problems.
3. **Analyze and design context-free grammars** and pushdown automata for specifying formal languages.
4. **Differentiate between decidable and undecidable problems** using formal proofs and examples.
5. **Interpret the limits of computation** through Turing machines and undecidability concepts in language recognition.

UNIT - I

Introduction to Finite Automata: Structural Representations, Automata and Complexity, the Central Concepts of Automata Theory – Alphabets, Strings, Languages, Problems.

Nondeterministic Finite Automata: Formal Definition, an application, Text Search, Finite Automata with Epsilon-Transitions.

Deterministic Finite Automata: Definition of DFA, How A DFA Process Strings, The language of DFA, Conversion of NFA with €-transitions to NFA without €-transitions. Conversion of NFA to DFA, Moore and Melay machines

UNIT - II

Regular Expressions: Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions, Conversion of Finite Automata to Regular Expressions.

Pumping Lemma for Regular Languages, Statement of the pumping lemma, Applications of the Pumping Lemma.

Closure Properties of Regular Languages: Closure properties of Regular languages, Decision Properties of Regular Languages, Equivalence and Minimization of Automata.

UNIT - III

Context-Free Grammars: Definition of Context-Free Grammars, Derivations Using a Grammar, Leftmost and Rightmost Derivations, the Language of a Grammar, Sentential Forms, Parse Tress, Applications of Context-Free Grammars, Ambiguity in Grammars and Languages.

Push Down Automata: Definition of the Pushdown Automaton, the Languages of a PDA, Equivalence of PDA's and CFG's, Acceptance by final state, Acceptance by empty stack, Deterministic Pushdown Automata. From CFG to PDA, From PDA to CFG.

UNIT - IV

Normal Forms for Context- Free Grammars: Eliminating useless symbols, Eliminating €-Productions. Chomsky Normal form Greibach Normal form.

Pumping Lemma for Context-Free Languages: Statement of pumping lemma, Applications **Closure Properties of Context-Free Languages:** Closure properties of CFL"s, Decision Properties of CFL's Turing Machines: Introduction to Turing Machine, Formal Description, Instantaneous description, The language of a Turing machine

UNIT - V

Types of Turing machine: Turing machines and halting

Undecidability: Undecidability, A Language that is Not Recursively Enumerable, An Undecidable Problem that is RE, Undecidable Problems about Turing Machines, Recursive languages, Properties of recursive languages, Post's Correspondence Problem, Modified Post Correspondence problem, Other Undecidable Problems, Counter machines.

TEXT BOOKS:

1. Introduction to Automata Theory, Languages, and Computation, 3nd Edition, John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Pearson Education.
2. Theory of Computer Science – Automata languages and computation, Mishra and Chandrasekaran, 2nd edition, PHI.

REFERENCE BOOKS:

1. Introduction to Languages and The Theory of Computation, John C Martin, TMH.
2. Introduction to Computer Theory, Daniel I.A. Cohen, John Wiley.
3. A Text book on Automata Theory, P. K. Srimani, Nasir S. F. B, Cambridge University Press.
4. Introduction to the Theory of Computation, Michael Sipser, 3rd edition, Cengage Learning.
5. Introduction to Formal languages Automata Theory and Computation Kamala Krithivasan, Rama R, Pearson.

DATABASE MANAGEMENT SYSTEMS

Department of Information Technology					II B.Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A2541203	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Course Outcomes: At the end of the course, student will be able to:

1. Gain knowledge of fundamentals of DBMS, database design.
2. Apply basic SQL concepts for effective retrieval and management of data.
3. Analyze various Normal Forms to carry out the concepts of Schema Refinement.
4. Understand the basics of transaction processing and concurrency control.
5. Understand the database storage structures and access techniques

UNIT-I

Database System Applications: A Historical Perspective, File Systems versus a DBMS, the Data Model, Levels of Abstraction in a DBMS, Data Independence, Structure of a DBMS

Introduction to Database Design: Database Design and ER Diagrams, Entities, Attributes, and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design With the ER Model

UNIT-II

Introduction to the Relational Model: Integrity constraint over relations, enforcing integrity constraints, querying relational data, logical database design, introduction to views, destroying/altering tables and views.

Relational Algebra, Tuple relational Calculus, Domain relational calculus.

UNIT-III

SQL: QUERIES, CONSTRAINTS, TRIGGERS: form of basic SQL query, UNION, INTERSECT, and EXCEPT, Nested Queries, aggregation operators, NULL values, complex integrity constraints in SQL, triggers and active databases.

Schema Refinement: Problems caused by redundancy, decompositions, problems related to decomposition, reasoning about functional dependencies, FIRST, SECOND, THIRD normal forms, BCNF, lossless join decomposition, multivalued dependencies, FOURTH normal form, FIFTH normal form.

UNIT-IV

Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for serializability, Lock Based Protocols, Timestamp Based Protocols, Validation- Based Protocols.

UNIT-V

Hashing: Hash Based Indexing, Tree based Indexing, Comparison of File Organizations, Indexes-Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM)

B+ Trees: A Dynamic Index Structure.

TEXT BOOKS:

1. Database System Concepts, Silberschatz, Korth, McGraw hill, V edition.3rd Edition
2. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill

REFERENCE BOOKS:

1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate, Pearson Education
3. Introduction to Database Systems, C. J. Date, Pearson Education
4. Oracle for Professionals, The X Team, S.Shah and V. Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition.

FULL STACK DEVELOPMENT

Department of Information Technology					II B.Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A254501	L	T	P	C	CIE	SEE	Total
	3	0	0	3	40	60	100

Course Outcomes: At the end of the course, student will be able to:

1. Design aesthetically appealing and responsive web pages using HTML and CSS.
2. Develop real-world React web applications and utilize related development tools.
3. Apply agile methodologies for rapid and efficient project development.
4. Build complete end-to-end applications from scratch using Node.js.
5. Understand and implement logical relationships between documents using MongoDB.

UNIT-I

Introduction to Full Stack: MVC pattern, Web Fundamentals. **HTML 5.0:** Basic tags, HTML DOM, Images, Tables, Lists, Forms, Layout, Graphics, span and div tags. **Cascading Style Sheets (CSS):** Types of CSS, CSS Selectors, CSS BOX Model, Text and Font, Color, CSS Positioning and CSS floating, CSS Grid layout Module, CSS Media Queries.

UNIT-II

JavaScript: Data Types & Type Conversion, JSON, Events, String and Date Functions, Local Storage, Object Oriented Programming (OOP) in JS, Arrow Functions, Async/Await, JavaScript Regular Expressions. **Bootstrap:** Introduction of Bootstrap, Container and Container-fluid, Bootstrap Carousel. **Bootstrap Components:** Button, Grid, Table, Form, Alert, Image, Tabs/Pill, Navbar, Modals.

UNIT-III

React JS: Introduction to React, React with JSX, Actual DOM vs React VDOM, Components, Lifecycle, State, Props, Fragments, Events, Router, Forms, Pagination, Tables, Portals, Hook, Signals, React 18 New Features.

Integration of Google MAP API and GPS Location Tracking: Incorporating Google MAP API and GPS Location Tracking for location-based services.

UNIT-IV

Node JS: Modules, Node Package Manager (npm), Creating Web Server, Sending Requests and Handling HTTP requests, Handling User Authentication with NodeJS, File System, Writing a file asynchronously and Other I/O Operations. **Events:** Event Emitter class, Inheriting Events and Returning event emitter. **Express JS:** Introduction to the Express framework – Server-side rendering with Templating Engines, Routing, Middleware, Custom Middleware, JWT Authentication.

UNIT-V

Mongo DB: Introduction, Importance of NoSQL databases, JSON Vs BSON, Data types and examples. CRUD Operations, Data Modelling & Schema Design, Indexing and Aggregation, MongoDB Replication and Sharding.

TEXT BOOKS:

1. Vasan Subramanian, "Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node", Second Edition, Apress Publications, 2019.
2. David Hows, Peter Membrey, Eelco Plugge – "MongoDB Basics", Apress, 2014.

REFERECNE BOOK:

1. Ethan Brown, "Web Development with Node and Express", O'Reilly Publishers, First Edition, 2014.

WEB RESOURCES:

1. https://web.stanford.edu/class/cs142/index.html
2. https://nodejs.org/en/docs/
3. https://www.mongodb.com/
4. https://reactjs.org/
5. https://getbootstrap.com/docs/5.0/utilities/api/
6. https://edu.anarcho-copy.org/Programming%20Languages/Node/

INNOVATION AND ENTREPRENEURSHIP

Department of Information Technology					II B.Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A254005	L	T	P	C	CIE	SEE	Total
	2	0	0	1	40	60	100

Course Objectives:

1. To familiarize on the basic concepts of innovation, entrepreneurship and its importance.
2. To Identify and analyze the process of problem-opportunity identification, market segmentation, and idea generation techniques.
3. To initiate prototype development and understand minimum viable product.
4. To develop initial Business and financial planning and Go-to-Market strategies
5. To impart knowledge on establishing startups, venture pitching and IPR

Course Outcomes: At the end of the course, student will be able to:

CO1. Understand the entrepreneurship and the entrepreneurial process and its significance in economic development.
CO2. Assess the problem from an industry perspective and generate solutions using the design thinking principles.
CO3. Assess market competition, estimate market size, and develop a prototype.
CO4. Analyze Business and financial planning models and Go-to-Market strategies.
CO5. Able to build a start-up, register IP and identify funding opportunities.

UNIT-I

Fundamentals of Innovation and Entrepreneurship Innovation: Introduction, need for innovation, Features, Types of innovations, fostering a culture of innovation, planning for innovation.

Entrepreneurship: Introduction, types of entrepreneurship attributes, mindset of entrepreneurial and intrapreneurial leadership, Role of entrepreneurs in economic development. Woman Entrepreneurship, Importance of on-campus startups, attributes and networks individuals while on campus.

Core Teaching Tool: Simulation, Game, Industry Case Studies (Personalized for students – 16 industries to choose from), Venture Activity.

UNIT-II

Problem and Customer Identification: Identification of gap, problem, analyzing the problem from a industry perspective, market and customer segmentation, validation of customer problem fit, Iterating

problem-customer fit, Porter's Five Force Model. Idea generation, Ideation techniques: Brainstorming, SCAMPER, Design thinking principles.

Core Teaching Tool: Several types of activities including: Class, game, Gen AI, 'Get out of the Building' and Venture Activity.

UNIT-III

Opportunity assessment and Prototype development: Identify and map global competitors, review industry trends, and understand market sizing: TAM, SAM, and SOM. Understanding prototyping and Minimum Viable Product (MVP). Developing a prototype: Testing, and validation.
Core Teaching Tool: Venture Activity, Class activity

UNIT-IV

Business & Financial Models: Introduction to Business Model and types, Business planning: components of Business plan- Sales plan, People plan and financial plan, Financial Planning: Types of costs, preparing a financial plan for profitability using a financial template, analyzing financial performance. Go-To-Market (GTM) approach,
Core Teaching Tool: Founder Case Studies – Sama and Securely Share; Class activity and discussions; Venture Activities.

UNIT-V

Startups and IPR: Startup requirements, building founding team members and mentors, pitch preparation, start-up registration process, funding opportunities and schemes, startup lifecycle, documentation, legal aspects in startup, venture pitching readiness, National Innovation Startup Policy (NISP) and its features. Patents, Designs, Patentability, Procedure for grants of patents. Indian Scenario of Patenting, International Scenario: International cooperation on Intellectual Property. Patent Rights, Copyright, trademark, and GI**.** Licensing and transfer of technology.
Core Teaching Tool: Expert talks; Cases; Class activity and discussions; Venture Activities.

TEXT BOOKS:

1. John R Bessant, Joe Tidd, Innovation and Entrepreneurship, 4E, Wiley, Latest Edition.
2. Ajay Batra, The Stratup Launch Book- A Practical Guide for Launching Customer Centric Ventures, Wiley, 2020. (For Core Teaching Tool).

REFERECNE BOOK:

1. Entrepreneurship Development and Small Business Enterprises, Poornima M Charantimath, 3E, Pearson, 2018.
2. D.F. Kuratko and T.V. Rao, Entrepreneurship: A South-Asian Perspective, Cengage Learning, 2013.
3. Robert D. Hisrich, Michael P. Peters, Dean A. Shepherd, Sabyasachi Sinha (2020). Entrepreneurship, McGrawHill, 11th Edition.

COMPUTER NETWORKS AND SIMULATION LAB

Department of Information Technology					II B.Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A2541281	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Outcomes:

1. **Implement and analyze Data Link Layer framing techniques** along with error detection and correction methods.
2. **Analyze and apply routing algorithms and congestion control techniques** in network design.
3. **Implement encoding and decoding techniques** used at the Presentation Layer for data representation and security.
4. **Use standard network tools** to monitor, analyze, and troubleshoot network performance.

List of Experiments

1. Implement the data link layer framing methods such as character, character-stuffing and bit stuffing.
2. Write a program to compute CRC code for the polynomials CRC-12, CRC-16 and CRC CCIP
3. Develop a simple data link layer that performs the flow control using the sliding window protocol, and loss recovery using the Go-Back-N mechanism.
4. Implement Dijsktra‟s algorithm to compute the shortest path through a network
5. Take an example subnet of hosts and obtain a broadcast tree for the subnet.
6. Implement distance vector routing algorithm for obtaining routing tables at each node.
7. Implement data encryption and data decryption
8. Write a program for congestion control using Leaky bucket algorithm.
9. Write a program for frame sorting techniques used in buffers.
10. **Wireshark**
    i. Packet Capture Using Wire shark
    ii. Starting Wire shark
    iii. Viewing Captured Traffic
    iv. Analysis and Statistics & Filters.

1. How to run Nmap scan
2. Operating System Detection using Nmap
3. Do the following using NS2 Simulator

I. NS2 Simulator-Introduction
II. Simulate to Find the Number of Packets Dropped
III. Simulate to Find the Number of Packets Dropped by TCP/UDP
IV. Simulate to Find the Number of Packets Dropped due to Congestion
V. Simulate to Compare Data Rate & Throughput.
VI. Simulate to Plot Congestion for Different Source/Destination

VII. Simulate to Determine the Performance with respect to Transmission of Packets

TEXT BOOK:

1. Computer Networks, Andrew S Tanenbaum, David. j. Wetherall, 5th Edition. Pearson Education/PHI

REFERENCES:

1. An Engineering Approach to Computer Networks, S.Keshav, 2nd Edition, Pearson Education
2. Data Communications and Networking – Behrouz A. Forouzan. 3rd Edition, TMH.

DATABASE MANAGEMENT SYSTEMS LAB

Department of Information Technology					II B.Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A254586	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Outcomes: At the end of the course, student will be able to:

1. **Apply conceptual database design principles** using the Entity–Relationship (E–R) model to design relational schemas.
2. **Use SQL DDL and DML commands** to create, modify, and manipulate data in relational databases.
3. **Execute advanced SQL queries** involving joins, constraints, set operations, and nested/correlated subqueries.
4. **Develop database applications** using programming components such as triggers, procedures, and cursors.

List of Experiments:

1. Concept design with E-R Model
2. Relational Model
3. Normalization
4. Practicing DDL commands
5. Practicing DML commands
6. A) Querying (using ANY, ALL, UNION, INTERSECT, JOIN, Constraints etc.)

   B) Nested, Correlated subqueries
7. Queries using Aggregate functions, GROUP BY, HAVING and Creation and dropping of Views.
8. Triggers (Creation of insert trigger, delete trigger, update trigger)
9. Procedures
10. Usage of Cursors

TEXTBOOKS:

1. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill,3rd Edition
2. Database System Concepts, Silberschatz, Korth, McGraw Hill, V edition.

REFERENCE BOOKS:

1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate, Pearson Education
3. Introduction to Database Systems, C.J. Date, Pearson Education
4. Oracle for Professionals, The X Team, S. Shah and V. Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition.

FULL STACK DEVELOPMENT LAB

Department of Information Technology					II B.Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A254582	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Outcomes: At the end of the course, student will be able to:

1. **Design and develop responsive web pages** using HTML5 and CSS.
2. **Apply JavaScript concepts** to build interactive and dynamic client-side web applications.
3. **Develop single-page applications (SPAs)** using the React framework.
4. **Design and implement full-stack web applications** using Node.js and real-time database technologies.

List of Experiments:

1. Design a Login Page using HTML, CSS (Media Query), and JavaScript.
2. Design a chessboard pattern using HTML and CSS.
3. Design a calculator application using JavaScript.
4. Create a responsive web page of your class timetable using the Bootstrap Grid System.
5. Create a timer component to start, pause, and reset using ReactJS.
6. Create a React component that checks the strength of a password and displays the result to the user.
7. Design authorized endpoints using JWT (JSON Web Token).
8. Develop a backend application with REST API to perform CRUD operations on student data (use Postman tool).
9. Design a replica set of the student database, insert records in the primary node, and display the records in secondary nodes.
10. Implement MongoDB Sharding to distribute and access student records across multiple shards for improved scalability.

TEXT BOOKS:

1. Vasan Subramanian, "Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node", Second Edition, Apress Publications, 2019.
2. David Hows, Peter Membrey, Eelco Plugge – “MongoDB Basics”, Apress, 2014.

SUGGESTED READING:

1. Ethan Brown, “Web Development with Node and Express”, O’Reilly Publishers, First Edition, 2014.

WEB RESOURCES:

1. https://web.stanford.edu/class/cs142/index.html
2. https://nodejs.org/en/docs/
3. https://www.mongodb.com/
4. https://reactjs.org/

UI DESIGN - FLUTTER

Department of Information Technology					II B.Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A254587	L	T	P	C	CIE	SEE	Total
	0	0	2	1	40	60	100

Course Outcomes:

1. Design responsive web pages using HTML, CSS, Bootstrap, and JavaScript.
2. Develop backend applications using Java and Node.js.
3. Build single-page applications with React.js.
4. Create Flutter apps with custom widgets, layouts, and forms.

List of Experiments: Students need to implement the following experiments

1. Build a responsive website with registration, login, catalog, and cart pages using HTML, CSS3, Bootstrap, and JavaScript validation.
2. Create a react application for the student management system having registration, login, contact, about pages and implement routing to navigate through these pages
3. Create a service in react that fetches the weather information from Openwea thermap.org and the display the current and historical weather information using graphical representation using chart.js
4. Develop an express web application that can interact with REST API to perform CRUD operations on student data. (Use Postman)
5. Implement JWT authentication in Node.js to create secure endpoints.
6. Create a TODO application in react with necessary components and deploy it into github.
7. Integrate a React frontend with Node.js backend to perform CRUD operations on a shared dataset (e.g., MongoDB, MySQL.)
8. Build a Django REST API for managing student data, tested using Postman.
9. Install Flutter and Dart SDK; Write Dart programs on data types, control flow, Functions, Class & Objects and collections to understand syntax and features
10. Create a Flutter app showcasing common widgets (Text, Image, Container, Card, ListView).
11. Design a responsive UI using Row, Column, Stack, media queries, and breakpoints.
12. Implement screen navigation using Navigator and named routes.
13. Use stateful & stateless widgets; manage state using Provider or setState.
14. Design a form in Flutter with input fields, validation, and error handling (e.g., student registration form).
15. Fetch and display data from a REST API (e.g., weather or student info) in Flutter.
16. Add basic animations (fade or slide) to a Flutter app for enhancing UI interactions

TEXT BOOK:

1. Marco L. Napoli, Beginning Flutter: A Hands-on Guide to App Development.

REFERENCE BOOKS:

1. Jon Duckett, Beginning HTML, XHTML, CSS, and JavaScript, Wrox Publications, 2010
2. Bryan Basham, Kathy Sierra and Bert Bates, Head First Servlets and JSP, O"Reilly Media, 2nd Edition, 2008.
3. Vasan Subramanian, Pro MERN Stack, Full Stack Web App Development with Mongo, Express, React, and Node, 2nd Edition, A Press.

INDIAN KNOWLEDGE SYSTEM

Department of Information Technology					II B.Tech II Semester		
Course Code	Hours/Week			Credits	Marks		
A254001	L	T	P	C	CIE	SEE	Total
	1	0	0	1	40	60	100

Bharat is considered one of the oldest civilizations of the world. Some of the archaeological evidences proved the existence of Indus Valley Civilization in 7000 B.C. Bhartiya traditions, culture, cultural activities, rituals, sacraments, painting, art of dancing, art of singing etc. is being practised till the modern times without knowing scientific approaches behind that. Eternity of Indian knowledge system proved itself that not only many rituals but also many traditions, many streams of knowledge like astrology, mathematics, physics, chemistry, biology, language studies, yoga and meditation had been following from the starting till now with some changes, in the form of traditions.

This course is for undergraduate students to inculcate Indian values. It will promote advance study and inter disciplinary research on all aspects of the Indian knowledge system.

Course Objectives: This course aims:

1. To provide a tribune of the rich culture and traditions of Indian knowledge system to students of various disciplines.
2. To introduce historical account on the education and scientific literature available in ancient Indian traditions and its connections with ancient Indian Philosophy
3. To give insights about the applications of Bharatiya Jnana Parampara
4. To introduce Indian approach towards health and wellbeing
5. To elaborate vast contribution of ancient Indian researchers, engineers, scientists and architects to the modern world

Course Outcomes: Students will be able to:

1. Understand nature, scope and related fields of Indian knowledge system.
2. Demonstrate the scientific literature available in ancient Indian traditions
3. Understanding the application of Bharatiya Jnana Parampara
4. Understand Indian approach towards Wellbeing
5. Appreciate vast contribution of ancient Indian researchers, engineers, scientists and architects to the modern world

UNIT-1: Introduction to Indian Knowledge Systems

Meaning**,** Nature, Scope and Salient Aspects of Bharatiya Jnana Parampara - Introduction to Vedas, Upanishads, Vidya, Kala, Jnana, Shastra - Practices and Continuity of Tradition

UNIT-2: Overview of History of Indian Education and Scientific Literature

Gurukul System - Role of Sanskrit in Natural Language Processing - Scientific Literature - Vedic Literature - Available Scientific Treatises - Interlinkings

UNIT-3: Introduction to Scientific Theories from Pure Sciences from Ancient Indian Knowledge Systems

Overview of theories from available ancient Indian Literature about Physics, Chemistry and Mathematics - Interlinkings and applications

UNIT-4: Introduction to Ancient Indian Wellness Systems

Concept of Wellness – Yoga System - Ayurveda System - Ancient Indian Aesthetics

UNIT-5: Development of Engineering, Science, Technology & Fine Arts in India

Various Industries - Silk, Cotton and Ship Building - Evolution of Indian Fine Arts — Cave and Temple Architecture, Vastu - Vidya, Sculpture, Forts and Stepwells, Observatories and Paintings - Music and Natyakala - Cultural Traditions & Folk Arts

- **Pedagogy for Teachers: Apart from Class Room Instruction, the following Methods are Suggested.**

1. Project based activities and learning.
2. Presentation and case studies.
3. Film screening and book reviews.
4. Visit to historical places, archives centre, research centre or library nearby.

***Note**: Activities mentioned above are only suggestive. Teacher-educators should encourage students to be innovative.*

Suggested Readings:

1. B. Mahadevan, Bhat Vinayak and Nagendra Pavan R.N., (2022) '*Introduction to Indian Knowledge Systems: Concepts and Applications*' PHI learning PVT, New Delhi ISBN [9789391818203]
2. Dharmapal (1971) '*Indian Science and Technology in the Eighteenth Century*'. Other India Press, Goa.
3. Kapil Kapoor, Singh Avdhesh Kumar, (2005) '*Indian Knowledge Systems*' D.K. Printworld (P) Ltd. ISBN 10: 8124603367 / ISBN 13: 9788124603369
4. Chakradeo, Ujwala, Temples of Bharat, Aayu Publications, New Delhi, 2024.
5. D.N. Bose, S.N. Sen and B. V. Subbarayappa, *A Concise History of Science in India*, Indian National Science Academy, New Delhi, 2009.
6. Datta B. and A. N. Singh, *History of Hindu Mathematics: Parts I and II*, Asia Publishing House, Bombay, 1962.
7. Kapoor, K. (2021), *Indian Knowledge System: Nature, Philosophy, Character in Indian Knowledge System*, vol. 1, Pub. Indian Institute of Advanced Studies, Shimla
8. Mahadevan, B., Bhat, V.R., Pavana, N. (2022), Philosophical Systems, in Introduction to Indian Knowledge System, Pub. PHI Learning, New Delhi.
9. Mahadevan, B., Bhat, V.R., Pavana, N. (2022), Knowledge: Framework and Classification, in Introduction to Indian Knowledge System, Pub. PHI Learning, New Delhi.

Video Resources:

1. Introductory lectures by Prof. Gauri Mahulikar
2. Introductory lectures by Prof. Kapil Kapoor

Websites:

- https://iksin dia.org/index.php
- Official Website of IKS- Indian Knowledge System
- https://www.youtube.com/watch?v=uKcf-hSlcUE
- Address by Prof Kapil Kapoor | Indian Institute of Advanced Study (FDP 2021)
- https://www.youtube.com/watch?v=MDJTXNiH2_A
- Mukul Kanitkar on Bharatiya Knowledge System
- https://www.youtube.com/watch?v=uARMhv97pjk
- https://www.youtube.com/watch?v=oTwgf56GbsA
- Scientific History of India | Mukul Kanitkar Lecture in DTU
- https://youtu.be/gNJNmPJqXJc?si=WFBbuUT65mLZzpOW
- Ancient India's Scientific Achievements & Contribution in Mathematics, Astronomy, Science & Medicine