



# **First Year Curriculum**

**Admission Year 2026-27**

**Bachelor of Technology  
CSE in Big Data Analytics**

**Faculty of Engineering & Technology**

**Parul University**

**Vadodara, Gujarat, India**

## Semester 1

- a. **Course Name:** Web Design
- b. **Course Code:** 03010501PC01
- c. **Prerequisite:** Basic understanding computer concepts and basic programming.
- d. **Rationale:** Fundamental knowledge of web design and development is essential for designing interactive, accessible, and responsive websites that meet modern digital communication and industry needs.

### e. Course Learning Objective:

<b>CLOBJ 1</b>	To explain the fundamentals of the Internet, web technologies, website structures, and the web development lifecycle.
<b>CLOBJ 2</b>	To design and develop web pages using HTML5 and CSS3 with responsive layouts, multimedia elements, and styling techniques.
<b>CLOBJ 3</b>	To apply UI/UX design principles, accessibility concepts, and Bootstrap components for creating user-friendly web interfaces.
<b>CLOBJ 4</b>	To demonstrate introductory full stack development concepts, including JavaScript, databases, hosting, and frontend-backend interaction.

### f. Course Learning Outcomes:

<b>CLO 1</b>	Explain the basics of the Internet, World Wide Web, browsers, and how websites work.
<b>CLO 2</b>	Construct simple web pages using HTML with text, images, links, lists, tables, and forms.
<b>CLO 3</b>	Apply CSS to style web pages using colors, fonts, layouts, and responsive design techniques.
<b>CLO 4</b>	Identify key UI/UX principles and create simple wireframes for structured webpage design.
<b>CLO 5</b>	Describe the basics of frontend and backend development, JavaScript, databases, and full stack technologies.

### g. Teaching & Examination Scheme:

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
3	0	2	4	20	20	20	60	30	150

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation, **CE-** Continuous Evaluation, **ESE-** End Semester Examination

## h. Course Content:

Sr. No.	Content	Weightage	Teaching Hours
1	<b>Introduction to Internet and Web</b> Internet Introduction, History of the World Wide Web, How Websites Work (Client-Server Model), Web Browser, Types of Websites (Static vs. Dynamic), Basics of URLs and Domain Names, Web Development Process lifecycle.	10%	8
2	<b>HTML - Building Web Pages</b> HTML, Basic Structure of an HTML Page, Headings and Paragraphs, Links tag, Image tag, Video tag, Lists, Tables, HTML Forms, Introduction to Semantic HTML. Features of HTML5.	20%	6
3	<b>CSS - Styling Web Pages</b> CSS, Types of CSS (Inline, Internal, External), Basic Selectors, Colors and Backgrounds, Font Styling and Text Formatting, Box Model (Margin, Padding, Border), Basic Layouts (Centering, Aligning Elements), Introduction to Responsive Design (Media Queries). Hover Effects. Features of CSS3.	30%	10
4	<b>UI/UX Basics UI/UX Basics</b> What is UI & UX? Importance of Good Design, Simple Layout Techniques, Basic Color and Font Choices, Introduction to Wireframing (Paper Sketching), Common Website Structures (Header, Content, Footer), Introduction to Accessibility in Web Design. <b>Bootstrap Basics</b> UI, Bootstrap Introduction, Bootstrap Layouts, Forms, Buttons, Cards, Carousel.	25%	11
5	<b>Introduction to Full Stack Development</b> Role of frontend, Role of backend, Basics of JavaScript (Alerts, Variables, Buttons, Events), How Websites Handle Data, Introduction to Databases (Basic Idea of Storing Data), Overview of Full Stack Technologies (MERN, Java Spring, .NET), How Websites are Hosted (Basic Idea of Domains and Hosting).	15%	6

## i. List of Practicals:

List of Practical	
1.	<b>Creating Your First Webpage</b> Create a basic HTML page that displays your name, enrollment number, and a brief introduction. Add a heading, a paragraph, and a horizontal line. Save and open the file in a web browser.
2.	<b>Enhancing the Webpage with Links and Images</b> Extend the previous webpage by adding at least three hyperlinks, including internal and external links. Insert an image and align it properly. Add a "Back to Top" button using an anchor link for smooth navigation.
3.	<b>Building a Personal Information Table</b> Create a 4x4 table that contains your educational details, including Subject, Year, Percentage, and Grade. Use proper table headers, borders, and cell spacing. Apply row span and column span where

	needed to improve table readability.
<b>4.</b>	<p><b>Creating an Interactive Form</b></p> <p>Design a registration form using HTML forms. Include text fields, radio buttons, checkboxes, dropdowns, and a submit button. Add proper labels and placeholders for each input field to enhance user experience.</p>
<b>5.</b>	<p><b>Applying Basic CSS Styling</b></p> <p>Link an external CSS file to your existing webpage. Apply background colors, font styles, and borders. Style the table and form for a better appearance, ensuring a consistent and visually appealing design.</p>
<b>6.</b>	<p><b>CSS Box Model and Layout Design</b></p> <p>Implement margins, padding, and borders using the CSS Box Model. Create a two-column layout using either the float property or flexbox. Ensure that the layout is properly centered and aligned for a structured webpage design.</p>
<b>7.</b>	<p><b>Making a Responsive Webpage</b></p> <p>Modify your existing webpage to be mobile-friendly. Use media queries to adjust font size and layout for different screen sizes. Add a navigation bar that adapts to different devices, ensuring a smooth user experience on both desktop and mobile.</p>
<b>8.</b>	<p><b>Introduction to UI/UX - Wireframing &amp; Color Theory</b></p> <p>Design a wireframe for a basic portfolio website either hand-drawn or using a tool like Figma. Choose appropriate color schemes and fonts for the website and justify your design choices based on UI/UX principles.</p>
<b>9.</b>	<p><b>Adding JavaScript for Basic Interactivity</b></p> <p>Write JavaScript to display an alert message when the page loads. Create a button that changes the background color of the webpage when clicked. Implement a simple form validation that checks if required fields are filled before submission.</p>
<b>10.</b>	<p><b>Final Mini Project - Building a Simple Portfolio Website</b></p> <p>Combine all skills learned so far to create a personal portfolio website. Include multiple pages such as Home, About, and Contact, linked together. Apply HTML, CSS, and basic JavaScript for styling and interactivity, ensuring that the website is fully responsive and visually appealing.</p>

**j. Text Book and Reference Book:**

1. **Head First HTML and CSS: A Learner's Guide to Creating Standards-Based Web Pages (TextBook)**  
By Elisabeth Robson and Eric Freeman
2. **Don't Make Me Think: A Common-Sense Approach to Web Usability**  
By Steve Krug | New Riders
3. **Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics (TextBook)**  
By Jennifer Niederst Robbins
4. **Developing web design latest edition**  
By Jennifer Niederst Robbins

(2)

- a. **Course Name:** Calculus
- b. **Course Code:** 03019101BS01
- c. **Prerequisite:** Basic concepts of Calculus
- d. **Rationale:** Calculus is the branch of mathematics that deals with continuous change. Calculus is also called infinitesimal calculus or “the calculus of infinitesimals”. The meaning of classical calculus is the study of continuous change of functions. Most of these quantities are the functions of time such as velocity is equal to change in distance with respect to time.
- e. **Course Learning Objectives:**

<b>CLOBJ 1</b>	Understand the consequences of various mean value theorems for differentiable functions.
<b>CLOBJ 2</b>	Apply the concept of Multivariate calculus in optimization problems.
<b>CLOBJ 3</b>	Apply the concept of multiple integration in problems.
<b>CLOBJ 4</b>	Analyze Inter-relationship amongst the line integral, double and triple integral formulations.

- f. **Course Learning Outcomes:**

<b>CLO 1</b>	Explain the consequences of various mean value theorems for differentiable functions.
<b>CLO 2</b>	Apply the concept of Multivariate calculus in optimization problems.
<b>CLO 3</b>	Apply the concept of multiple integration in problems.
<b>CLO 4</b>	Analyze Inter-relationship amongst the line integral, double and triple integral formulations.
<b>CLO 5</b>	Realize importance of Green, Gauss and Stokes’ theorems in other branches of mathematics.

g. **Teaching & Examination Scheme:**

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
4	-	-	4	20	20	-	60	-	100

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

h. **Course Content:**

Sr. No.	Content	Weightage	Teaching Hours
1	<b>Application of Functions of One Variable</b> Review of the function of one variable, limit, continuity, and differentiability, mean value theorems and applications, Riemann integration, fundamental theorem of calculus, application to length, area, volume, and surface area of revolution.	25%	15
2	<b>Multivariate Calculus</b> Functions of several variables, limit and continuity, partial derivatives and differentiability, composite functions, chain rule, implicit functions, Euler's theorem, Taylor's expansion, Jacobian, maxima and minima, method of Lagrange multipliers.	25%	15
3	<b>Multiple Integration</b> Double integration, change of order, change into polar form, application to area and volume.	20%	12
4	<b>Vector Calculus</b> Vector-valued function, velocity and acceleration, the gradient of a scalar function, directional derivatives, divergence and curl of a vector-valued function. Parameterization of curves and surfaces, vector fields, line integrals, Green's theorem, surface integrals, Gauss divergence theorem and Stokes' theorems with applications.	30%	18

i. **Text Book and Reference Book:**

- a. Calculus and Analytic Geometry, (Textbook), By G.B. Thomas and R.L. Finney, Addison Wesley
- b. Calculus, By Ron Larson, Bruce Edwards, | Cengage Learning.
- c. Calculus with early transcendental functions, By James Stewart, Cengage Learning.
- d. Calculus, By Tom M Apostol, | Wiley Publication.

(3)

- a. **Course Name:** Programming for Problem Solving
- b. **Course Code:** 03010501ES01
- c. **Prerequisite:** Requires Basic Knowledge of Computer
- d. **Rationale:** This course is design to provide basic ideas of computer programming. This course also makes help to understand programming language. It will help to develop their logical abilities.

e. **Course Learning Objective:**

<b>CLOBJ 1</b>	Recognize and recall fundamental principles and organizations of computers, demonstrating a foundational understanding of computer architecture and design.
<b>CLOBJ 2</b>	Comprehend the concepts of computer programming languages, illustrating a grasp of syntax, semantics, and the essential components of programming languages.
<b>CLOBJ 3</b>	Develop algorithms for solving basic engineering problems, demonstrating the ability to apply theoretical knowledge to practical problem-solving scenarios.
<b>CLOBJ 4</b>	Demonstrate proficiency in the practical application of C programming by writing, compiling, and debugging programs, showcasing the ability to implement and troubleshoot code effectively.
<b>CLOBJ 5</b>	Evaluate and analyse complex computational programs written in C, demonstrating the capacity to assess and understand intricate solutions to computational challenges.
<b>CLOBJ 6</b>	Develop simple projects using the C programming language, showcasing creativity and application of learned principles to produce functional and practical software solutions.

f. **Course Learning Outcomes:**

<b>CLO 1</b>	Recognize the computer's basic principles and organizations.
<b>CLO 2</b>	Understand Concepts of Computer Programming Language.
<b>CLO 3</b>	Develop the algorithm for solving basic Engineering Problems.
<b>CLO 4</b>	Write, Compile and debug program with C Programming.
<b>CLO 5</b>	Analyze the Solved, Complex Computational Program written in C.
<b>CLO 6</b>	Develop simple projects using C Language.

g. **Teaching & Examination Scheme:**

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
3	-	2	4	20	20	20	60	30	150

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

## h. Course Content:

Sr. No.	Topics	Weightage	Teaching Hours
1	<b>Introduction to 'C' Programming:</b> What is C language, History of C language, Application areas of C, Features of C language, structure of C Program, execution flow of program, reading a character, writing a character, formatted input, formatted output functions.	10%	8
2	<b>Constants, Variables, Data Types, Operators and Expressions</b> Constants, Variables, Data Types: Character Set, C tokens, Keywords and Identifiers, Constants, Variables, Data types, Declaration of Variables, Assigning values to variables, typedef, and Defining symbolic constants. Operators and Expression: Introduction to Operators and its types, Evaluation of expressions, Precedence of arithmetic operators, Type conversions in expressions, Operator precedence and associativity.	15%	5
3	<b>Control structures in C</b> Decision Making & branching: Decision making with If & If .. Else statements, If .. Else statements (Nested .... Ladder), The Switch, The break statement & goto statements, The ternary (?:) Operator Looping: The while statement, & The Do.. While loop, The FOR loop, Jump within loops – Programs	15%	4
4	<b>Arrays and Strings</b> Arrays: Introduction, One-dimensional array, Two-dimensional array, Concept of Multidimensional arrays.  Strings: String declaration, storage, Built-in-string functions.	20%	7
5	<b>User-Defined Functions, Structure and Unions</b> User-Defined Functions: Concepts of user defined functions, prototypes, definition of function, parameters, parameter passing, calling a function, recursive function, Macros, Pre-processing. Structure and Unions: Introduction, Structure definition, declaring and initializing Structure variables, Accessing Structure members, Copying & Comparison of structures, Arrays of structures, Arrays within structures, Structures within Structures, Structures and functions, Unions	20%	10
6	<b>Pointers, Dynamic memory allocation and File Management in C</b> Pointers: Basics of pointers, pointer to pointer, pointer and array, Pointer to array, array of pointers, functions returning pointers Dynamic memory allocation: Introduction to Dynamic memory allocation, malloc(), calloc(), free(), realloc(). File Management in C: Introduction and standard file handling functions.	20%	8

## i. Text Book and Reference Book:

1. "Programming in ANSI C", (Textbook), By E. Balaguruswamy, Tata McGraw-Hill

2. "C Programming: Test Your Skills", By Ashok Kamthane
3. "Computer Fundamentals", By P.K. Sinha and Priti Sinha, BPB Publications, 4<sup>th</sup> Edition
4. "Star C Programming", STAR Certification, C Certification Exam
5. "Programming with C", By Byron Gottfried, Tata McGraw Hill Education.
6. "C The Complete Reference", By Herbert Schildt
7. "Let Us C", By Yeshavant Kanetkar | BPB Publications

## j. Experiment List

List of Practical											
<b>1.</b>	<p><b>Practical Set 1 (Basics)</b></p> <ol style="list-style-type: none"> <li>1. Write a program to print HELLO FRIENDS!</li> <li>2. Write a program that reads two nos. from key board and gives their addition, subtraction, multiplication, division and modulo.</li> <li>3. Write a program to calculate area of circle, use <math>\Omega</math> as symbolic constants.</li> <li>4. Write a program to convert days into months and days.</li> <li>5. Write a program which calculates the summation of three digits from the given 3 digit number.</li> <li>6. Write a program to demonstrate enumerates data type.</li> <li>7. Write a program to compute Fahrenheit from centigrade.</li> <li>8. Write a program to calculate simple interest.</li> <li>9. Read the price of item in decimal form e.g. 12.50 and separate Rs and Paise from the given value e.g. 12 rupees and 50 paise.</li> </ol>										
<b>2.</b>	<p><b>Practical Set 2 (Control Structures)</b></p> <ol style="list-style-type: none"> <li>1. Write a program to find the largest of the three nos. using Nested-If-Else statement.</li> <li>2. Write a C program to enter a character and to check whether it is a small letter or it is a capital letter or it is a digit or it is a special symbol.</li> <li>3. Write a program to read marks from keyboard and your program should display equivalent grade according to following table.</li> </ol> <table style="margin-left: 20px;"> <tr><td>Mark</td></tr> <tr><td>s</td></tr> <tr><td>Grad</td></tr> <tr><td>e</td></tr> <tr><td>100-</td></tr> <tr><td>80</td></tr> <tr><td>Dist</td></tr> <tr><td>60-79 First Class</td></tr> <tr><td>35-59 Second Class</td></tr> <tr><td>0-34 Fail</td></tr> </table> <ol style="list-style-type: none"> <li>1. Write a program to read marks of a student from keyboard whether the student id pass (if).</li> <li>2. Write a program to find the sum of first N odd numbers.</li> <li>3. Write a program using while loop construct which finds the factorial of a given integer number.</li> <li>4. Write a C program using do...while and for loop constructs to reverse the digits of the number.</li> <li>5. Write a program to demonstrate use of Switch- Break Statement.</li> <li>6. Write a program to find out all the numbers divisible by 5 and 7 between 1 to 100. Check for Armstrong number. A number is Armstrong if sum of cube of every digit is same as the original number. E.g. <math>153=1^3+5^3+3^3=153</math></li> <li>7. Write a program to print the output of bellow series. <math>1!+2!+3!+4!+. \quad n!</math></li> <li>8. Write a program to print the following outputs using for Loop.</li> </ol>	Mark	s	Grad	e	100-	80	Dist	60-79 First Class	35-59 Second Class	0-34 Fail
Mark											
s											
Grad											
e											
100-											
80											
Dist											
60-79 First Class											
35-59 Second Class											
0-34 Fail											

	<p>(a) 1 (b) * 12 ** 123 ***</p> <p>9. Write a program to print the following outputs using for Loop. (a) 1 (b) 321 21 21 321 1</p>
3.	<p><b>Practical Set 3 (Array &amp; Strings)</b></p> <ol style="list-style-type: none"> <li>Write a program which sorts 10 numbers into ascending order.</li> <li>Write a program to find maximum element from 1-D array.</li> <li>Write a program to find number of odd and even elements from the 1-D array.</li> <li>Write a program add two 2x2 matrices.</li> <li>Write a program to count number of positive, negative and zero elements from 3x3 matrix.</li> <li>Write a function for the following operations on string: Copy one string to another</li> </ol> <p><b>Comparing two strings</b> <b>Adding a string to the end of another.</b></p> <ol style="list-style-type: none"> <li>Write a program to count vowels from a entered String.</li> <li>Write a program which finds whether a string is a palindrome or not.</li> </ol>
4.	<p><b>Practical Set 4 (Functions)</b></p> <ol style="list-style-type: none"> <li>Write a program to find factorial of a number using recursion.</li> <li>Write a program that used user defined function Swap ( ) and interchange the value of two variable.</li> <li>Write a function to return 1 if the number is prime otherwise return 0.</li> </ol>
5.	<p><b>Practical Set 5 (Structures)</b></p> <ol style="list-style-type: none"> <li>Define a structure type, personal that would contain person name, date of joining and salary.</li> <li>Define a structure called cricket that will describe the following information: Player name Team name Batting average</li> </ol>
6.	<p><b>Practical Set 6 (Pointers)</b></p> <ol style="list-style-type: none"> <li>Write a program to add two numbers using pointers.</li> <li>Write a program to swap two numbers using pointer</li> </ol>
7.	<p><b>Practical Set 7 (File Management)</b></p> <ol style="list-style-type: none"> <li>Write a program to illustrate reading files contents.</li> <li>Write a program to illustrate the use of fgets( ).</li> </ol>

(4)

- a. **Course Name:** Electrical and Electronics Engineering
- b. **Course Code:** 03010601E S02
- c. **Prerequisite:** Knowledge of Physics and Mathematics up to 12<sup>th</sup> science level
- d. **Rationale:** The main objective of this subject is introductory treatment of the field of Electrical and Electronics Engineering to the students of various branches of engineering.

**e. Course Learning Objective:**

<b>CLOBJ 1</b>	Gain familiarity with electrical current, potential difference, power and energy, sources of electrical energy and elements of electrical circuit.
<b>CLOBJ 2</b>	Solve problems related to Alternating current, alternating voltage, etc, Demonstrate a clear understanding of Pure R, L C circuit and combination of RLC, Series and Parallel combination of R, L and C, etc
<b>CLOBJ 3</b>	Acquire knowledge of the resistor, capacitor, and inductor and their performance characteristics for series and parallel connections.
<b>CLOBJ 4</b>	Understand different single phase and three phase circuits.
<b>CLOBJ 5</b>	Demonstrate a clear understanding of the basic concepts, working principles and applications of transformer, DC machines and AC machines.
<b>CLOBJ 6</b>	Study the use of LT Switch Gear, Fuse, MCB, ELCB etc.

**f. Course Learning Outcomes:**

<b>CLO 1</b>	Apply fundamental electrical laws and circuit theorems to electrical circuits.
<b>CLO 2</b>	Analyze single-phase and three phase AC circuits
<b>CLO 3</b>	Evaluate the operating regions of BJTs through DC analysis and Q-points using load line analysis
<b>CLO 4</b>	Design DC-regulated power supplies using voltage regulators and introduce various transducers with operation.

**g. Teaching & Examination Scheme:**

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
3	-	2	4	20	20	20	60	30	150

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

**h. Course Content:**

Sr. No.	Content	Weightage	Teaching Hours
1	<p><b>DC Circuits</b>            Concepts of Electrical Current, Voltage, Power and Energy. Electrical circuit elements (R, L and C), voltage and current sources, Kirchoff current and voltage laws, Mesh and Node analysis, Simplifications of networks using series and parallel combinations and star-delta conversions. Superposition, Thevenin and Norton Theorems.</p>	20%	10
2	<p><b>AC Circuits</b>            Sinusoidal voltages and currents, their mathematical and graphical representation, Concept of instantaneous, peak (maximum), average and R.M.S. values, frequency, cycle, period, peak factor and form factor, phase difference, lagging, leading and in phase quantities and phasor representation. Rectangular and polar representation of phasors, examples based on theory. Analysis with phasor diagrams of R, L, C, R-L, R-C and R-L-C series circuits and it's example. study of parallel R-L, R-C and R-L-C circuit, concept of impedance, admittance, conductance and susceptance in case of above combinations and relevant voltage-current phasor diagrams, Resonance in series and parallel circuits. Q-factor. concept of active, reactive and apparent power and power factor. Concept of three-phase supply and phase sequence. Voltages, currents and power relations in three-phase balanced star-connected loads and delta-connected loads along with phasor diagrams with examples.</p>	30%	13
3	<p><b>Diode and Transistors</b>            Introduction to Ideal Diode, Effect of temperature Ideal diodes, unbiased diode and Forward and reverse bias of Diode. PIV, surge current, Diode as Uncontrolled switch. Rectifiers: Half wave, Full wave, and bridge wave. Ripple factor, PIV rating. Choke and Capacitor input filter rectifiers, Clipper and Clamper circuits, Construction and working of BJT, Characteristics &amp; specifications of BJT (PNP &amp; NPN transistors), Biased and unbiased BJT, Configuration of the transistor, the concept of gain &amp; BW, Operation of BJT in the cut-off, saturation &amp; active regions (DC analysis), BJT as a switch, Transistor as an amplifier, Voltage divider bias and analysis, VDB load line and Q point.</p>	30%	10
4	<p><b>Voltage Regulators and Transducers</b>            Basic series and shunt regulator, Types of voltage regulator IC: Fixed and adjustable positive and negative linear voltage regulator, IC linear fixed voltage regulator (78XX, 79XX,) Linear Adjustable Regulator, DC Regulated Power supply. Introduction of</p>	20%	10

	Transducers. Types of Transducers and its applications		
--	--	--	--

**i. Text Book and Reference Book:**

1. A textbook of Electrical Technology Volume 1&2, By B. L. Theraja, S. Chand Publication
2. Electrical Engineering Fundamentals, By V. D. Toro, Prentice Hall India, Pub. Year 1989
3. Electrical and Electronics Technology, By E. Hughes Pearson, Pub. Year 2010
4. Basic Electrical Engineering, By D. P. Kothari and I. J. Nagrath, Tata McGraw Hill, Pub. Year, 2010
5. Basic Electrical Engineering, By D. C. Kulshreshtha, McGraw Hill
6. Introduction to Electrical Engineering by M.S Sarma, Oxford university press 2001  
By M.S Sarma | Oxford university press 2001

**j. Experiment List:**

Sr. NO.	Experiment List
1	<b>To study about various electrical and electronics symbols and demonstrate various measuring instruments used in electrical &amp; electronics laboratories.</b>
2	<b>Verification of superposition theorem with DC source.</b>
3	<b>Verification of Thevenin's theorem with DC source.</b>
4	<b>To experimentally investigate the behaviour of a series RLC circuit under AC excitation.</b>
5	<b>Verification of current and voltage relations in three-phase balanced star and delta-connected loads</b>
6	<b>To plot V-I characteristics of the P-N junction diode.</b>
7	<b>To perform half wave rectifier with and without filter.</b>
8	<b>To perform full wave rectifier with and without filter</b>
9	<b>To plot and study input-output characteristics of the Common Emitter (CE) configuration of the transistor</b>
10	<b>To perform and observe the response of voltage regulator IC 7805</b>

(5)

- a. **Course Name:** Prompt Engineering
- b. **Course Code:** 03010501ES05
- c. **Prerequisite:** Basic computer literacy; familiarity with any programming language
- d. **Rationale:** This course introduces students to the principles and practices of Prompt Engineering for Large Language Models (LLMs) and Generative AI systems. Students will learn how to design, refine, and evaluate prompts to achieve desired outputs across various domains including code generation, education, data analysis, and content creation. The course combines theoretical understanding with extensive hands-on practice through laboratory sessions.
- e. **Course Learning Objective:**

<b>CLOBJ 1</b>	To explain the fundamentals of artificial intelligence, large language models, generative AI, and the principles of prompt engineering.
<b>CLOBJ 2</b>	To apply core prompting techniques and prompt design strategies for generating accurate, structured, and context-aware AI responses.
<b>CLOBJ 3</b>	To develop advanced prompting workflows using reasoning, chaining, meta-prompting, and retrieval-augmented generation techniques.
<b>CLOBJ 4</b>	To demonstrate domain-specific applications of prompt engineering while evaluating ethical, responsible, and effective use of AI systems.

f. **Course Learning Outcomes:**

<b>CLO 1</b>	Demonstrate proficiency in designing effective prompts for various LLM applications
<b>CLO 2</b>	Apply appropriate prompting techniques based on task requirements
<b>CLO 3</b>	Evaluate and iterate on prompt designs to improve output quality
<b>CLO 4</b>	Understand the ethical implications and limitations of generative AI
<b>CLO 5</b>	Build practical applications leveraging prompt engineering principles

g. **Teaching & Examination Scheme:**

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
2	-	2	3	20	20	20	60	30	150

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation, **CE-** Continuous Evaluation, **ESE-** End Semester Examination

h. **Course Content:**

Sr. No.	Content	Weightage	Teaching Hours
1	<p><b>Foundations of AI &amp; Prompt Engineering</b></p> <p>Introduction to AI, LLMs, and Generative AI - Overview of artificial intelligence, introduction to large language models, and understanding generative AI systems at a high level.</p> <p>Evolution of Language Models - Historical progression from rule-based systems to modern transformer-based models like GPT.</p> <p>What is Prompt Engineering - Definition, importance, and real-world applications of prompt engineering across industries.</p> <p>Anatomy of a Prompt - Understanding prompt components: instruction, context, input data, and output format specifications.</p> <p>Types of Prompts - Classification of prompts: open-ended, instructional, conversational, and role-based prompts.</p> <p>How LLMs Generate Responses - Technical overview of tokens, temperature, top-p sampling, and other generation parameters.</p>	20	6
2	<p><b>Core Prompting Techniques</b></p> <p>Zero-Shot Prompting - Techniques for asking questions without providing examples.</p> <p>One-Shot and Few-Shot Prompting - Guiding model behavior through strategic use of examples. Chain-of-Thought (CoT) Prompting - Eliciting step-by-step reasoning from language models.</p> <p>Instruction-Based Prompting - Imperative framing techniques and achieving clarity in instructions. Role Prompting - Assigning personas and roles to models for context-appropriate responses.</p> <p>Contextual Prompting - Providing background information for enhanced output quality. Negative Prompting - Specifying what the model should avoid in its responses.</p> <p>Iterative Prompting - Refining outputs through systematic follow-up instructions.</p>	27	8
3	<p><b>Prompt Design Principles</b></p> <p>Clarity and Specificity - Understanding why vague prompts fail and techniques for precise prompt construction.</p> <p>Constraints and Format Control - Requesting specific output formats: JSON, bullet points, tables, word limits.</p> <p>Decomposing Complex Tasks - Breaking large requests into manageable sub-prompts for better results. Prompt Chaining - Sequential prompting where output from one prompt feeds into the next.</p> <p>Managing Hallucinations - Detection strategies and mitigation techniques for factual inaccuracies. Bias in LLM Outputs - Recognizing and reducing unintended bias in model responses.</p>	20	6

4	<p><b>Advanced Prompting Strategies</b></p> <p>Tree of Thoughts (ToT) Prompting - Exploring multiple reasoning paths for complex problem-solving. ReAct Prompting - Combining reasoning and action steps in prompt design.</p> <p>Self-Consistency Prompting - Sampling multiple outputs and selecting optimal responses. System Prompts and Meta-Prompting - Configuring model behavior at a higher architectural level.</p> <p>Retrieval-Augmented Generation (RAG) Concepts - Introduction to combining prompts with external knowledge sources.</p>	17	5
5	<p><b>Domain-Specific Applications</b></p> <p>Prompt Engineering for Code Generation - GitHub Copilot-style prompting techniques and AI-assisted debugging.</p> <p>Prompt Engineering for Education - Quiz generation, lesson planning, and educational explanation prompts.</p> <p>Prompt Engineering for Data Analysis - Using LLMs for data interpretation, SQL generation, and chart creation.</p> <p>Prompt Engineering for Content Creation - Professional writing: blogs, emails, summarization, and translation tasks.</p>	13	4
6	<p><b>Evaluation, Ethics &amp; Capstone</b></p> <ul style="list-style-type: none"> <li>• Evaluation, Ethics, and Responsible AI - Metrics for prompt quality assessment (accuracy, relevance, coherence); prompt versioning and documentation; ethical considerations including prompt injection risks.</li> </ul>	3	1

**i. Text Book and Reference Book:**

1. Alto, V. (2024). *Building LLM Apps*. Packt Publishing.  
Useful for advanced modules on RAG and chaining techniques.
2. Foster, D. (2023). *Generative Deep Learning* (2nd ed.). O'Reilly Media. Provides strong theoretical background on how LLMs work.
3. Taulli, T. (2024). *AI-Assisted Programming*. O'Reilly Media. Excellent companion for code generation practical labs.
4. Hunter, N. (2023). *The Art of Prompt Engineering with ChatGPT*.  
Clear, accessible introduction suitable for both technical and non-technical learners.
5. Phoenix, J., & Taylor, M. (2024). *Prompt Engineering for Generative AI*. O'Reilly Media.  
Best beginner-friendly textbook covering all core techniques with practical examples.

j. **Experiment List:**

Sr. NO.	Experiment List
1	<b>Interface Exploration Lab</b> Hands-on exploration of ChatGPT/Gemini interfaces; experimentation with different prompt types and observation of output variations by changing temperature and phrasing.
2	<b>Zero-Shot vs Few-Shot Comparison Lab</b> Comparative analysis of same tasks using different prompting strategies; evaluation of output quality across approaches.
3	<b>Chain-of-Thought Lab</b> Application of CoT prompting to mathematical problems, logical puzzles, and step-by-step reasoning tasks.
4	<b>Role Prompting Lab</b> Development of specialized agents: customer support bot, educational tutor, and code reviewer using role-based prompts.
5	<b>Format Control Lab</b> Generation of outputs in multiple structured formats: JSON, Markdown, HTML, and tabular representations.
6	<b>Prompt Chaining Lab</b> Building a 3-step pipeline: outline generation → draft creation → summarization for a research topic.
7	<b>Hallucination Detection Lab</b> Identification of factual errors in LLM responses and crafting prompts that minimize hallucinations.
8	<b>ToT and Self-Consistency Lab</b> Complex decision-making scenarios with multiple solution paths and consistency verification
9	<b>System Prompt Design Lab</b> Building a domain-specific assistant (e.g., coding tutor) using system-level instructions.
10	<b>Code Generation Lab</b> Writing Python functions, debugging code, and generating unit tests using prompt engineering.
11	<b>Education Content Lab</b> Generation of multiple-choice questions, lesson plans, and concept explanations for computer science topics.
12	<b>Data Analysis Lab</b> Using LLMs to write SQL queries and interpret dataset descriptions.
13	<b>Content Creation Lab</b> Drafting professional emails, blog posts, and summaries using prompt templates.
14	<b>Prompt Evaluation Lab</b> Creation of scoring rubrics and comparison of five different prompting strategies on identical tasks.
15	<b>Capstone Project</b> End-to-end project where students design a prompt-based application (e.g., interview prep bot, course assistant, code reviewer), document design decisions, and present results.



0	-	2	2	-	-	2	-	30	50
---	---	---	---	---	---	---	---	----	----

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

#### h. Course Content:

Sr. No.	Content	Weightage	Teaching Hours
1	<b>Parts of speech, Punctuation</b> <ul style="list-style-type: none"> <li>Overview of 8 parts of speech Basic punctuation marks and their usage</li> <li>Importance of grammar in writing</li> </ul>	10%	4
2	<b>Tenses Types:</b> <ul style="list-style-type: none"> <li>Present, Past, Future</li> <li>Forms and correct usage</li> <li>Common tense errors</li> </ul>	8%	2
3	<b>Phrase and Clauses:</b> <ul style="list-style-type: none"> <li>Types of Sentences (Structure Wise)</li> <li>Difference between phrases and clauses Classification of Simple, Compound &amp; Complex Interchange of Simple to Compound, Simple to Complex, Compound to Complex</li> </ul>	10%	4
4	<b>Picture Perception</b> <ul style="list-style-type: none"> <li>Observing and interpreting images Building a short story or description</li> <li>Improving vocabulary and creativity</li> </ul>	15%	4
5	<b>Paragraph Development</b> <ul style="list-style-type: none"> <li>Structure: topic sentence, development, conclusion</li> </ul>	8%	2
6	<b>Meeting and Greeting (Initiating a conversation):</b> <ul style="list-style-type: none"> <li>Basic conversation skills Formal and informal greetings</li> <li>Common phrases for starting interactions</li> </ul>	7%	2
7	<b>Reading Comprehension (Basic Level):</b> <ul style="list-style-type: none"> <li>Types – Skimming &amp; Scanning Reading short passages</li> <li>Identifying main ideas and details Answering basic questions</li> </ul>	12%	4
8	<b>Listening Comprehension (Basic Level):</b> <ul style="list-style-type: none"> <li>Listening to short audios/Conversation Understanding and responding to key information</li> </ul>	8%	2
9	<b>Extempore Speech:</b> <ul style="list-style-type: none"> <li>Speaking on random topics</li> <li>Organizing thoughts quickly</li> <li>Improving fluency and confidence</li> </ul>	15%	4

<b>10</b>	<b>Goal Setting and Tracking:</b> <ul style="list-style-type: none"> <li>● Setting SMART goals</li> <li>● Planning and tracking progress</li> <li>● Self-assessment and reflection</li> </ul>	<b>7%</b>	<b>2</b>
-----------	---	-----------	----------

**i. Text Book and Reference Book:**

1. **English Grammar in Use (TextBook)**  
By Murphy, Raymond | Cambridge University Press, Pub. Year 2019
2. **A Practical English Grammar (TextBook)**  
By Thomson & Martinet | Oxford University Press, Pub. Year 1986
3. **The St. Martin's Guide to Writing**  
By Rise B. Axelrod & Charles R. | Cooper, Bedford/St. Martin's, Pub. Year 2021
4. **Tactics for Listening (Basic)**  
By Jack C. Richards | Oxford University Press, Pub. Year 2011
5. **Active Skills for Reading: Book 1**  
By Neil J. Anderson | Cengage Learning, Pub. Year 2013
6. **Speak with Impact**  
By Allison Shapira | HarperCollins Leadership, Pub. Year 2018

## Semester 2

(1)

a. **Course Name:** Operating Systems

b. **Course Code:** 03010503PC05

c. **Prerequisite:** Fundamentals of Computer Systems

d. **Rationale:** This course is an introduction to the theory and practice behind modern computer operating systems. Topics will include what an operating system does (and doesn't) do, system calls and interfaces, processes, concurrent programming, resource scheduling and management, virtual memory, deadlocks, algorithms, programming, and security. The approach of the subject is from both a theoretical perspective as well as a practical one.

e. **Course Learning Objective:**

<b>CLOBJ 1</b>	To explain operating system fundamentals, architectures, virtualization technologies, and the role of modern desktop, mobile, and cloud operating systems.
<b>CLOBJ 2</b>	To analyze process management, multithreading, CPU scheduling algorithms, synchronization mechanisms, and deadlock handling techniques.
<b>CLOBJ 3</b>	To apply memory management and virtual memory concepts, including paging, segmentation, page replacement, and performance optimization techniques.
<b>CLOBJ 4</b>	To evaluate storage management, file systems, I/O operations, security mechanisms, and containerization technologies in modern operating systems.

f. **Course Learning Outcomes:**

<b>CLO 1</b>	Distinguish different styles of operating system design.
<b>CLO 2</b>	Describe device and I/O management functions in operating systems as part of a uniform device abstraction.
<b>CLO 3</b>	Describe disk organization and file system structure
<b>CLO 4</b>	Give the rationale for virtual memory abstractions in operating systems.
<b>CLO 5</b>	Explain the main principles and techniques used to implement processes and threads as well as the different algorithms for process scheduling.
<b>CLO 6</b>	Understand the main mechanisms used for inter-process communication.

**g. Teaching & Examination Scheme:**

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
3	0	2	4	20	20	20	60	30	150

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation, **CE-** Continuous Evaluation, **ESE-** End Semester Examination

**h. Course Content:**

Sr.	Topics	W	T
1	<b>OS Fundamentals &amp; Architecture</b> OS concepts, generations, types (batch/time-sharing/distributed/real-time). OS services, system calls (process/file/device). OS structures (layered/monolithic/microkernel/hybrid). Virtualization (Type 1/2 hypervisors: VMware/Xen/KVM/VirtualBox). Cloud OS (AWS EC2, Azure VMs, GCE). Mobile OS (Android: Linux kernel/HAL, iOS: Darwin/Core OS).	8	4
2	<b>Processes, Threads &amp; CPU Scheduling</b> Process fundamentals, states (new/ready/running/waiting/terminated), PCB, context switching. Threads (user-level/kernel-level), multithreading models (many-to-one/one-to-one/many-to-many). Scheduling criteria (CPU utilization, throughput, turnaround/waiting/response time). Algorithms: FCFS, SJF/SRTF, Round Robin, Priority (aging). MLQ, MLFQ. Multicore scheduling (load balancing, processor affinity). Real-time (RM, EDF). Linux CFS.	18	8
3	<b>Process Synchronization &amp; Deadlocks</b> Critical section, race conditions, mutual exclusion. Hardware solutions (test-and-set, compare-and-swap). Peterson's solution. Semaphores (counting/binary, P/V operations). Classical problems (Producer-Consumer, Readers-Writers, Dining Philosophers). Monitors, condition variables. Modern primitives (mutexes, spinlocks). Atomic operations, lock-free algorithms. Deadlock: necessary conditions (mutual exclusion, hold-and-wait, no preemption, circular wait). Prevention, avoidance (Banker's algorithm), detection (wait-for graph), recovery.	18	8
4	<b>Memory Management Techniques</b> Logical vs physical address, MMU. Contiguous allocation (fixed/variable partitioning, fragmentation, compaction). Paging (page table, PTBR, protection: valid-invalid bit, sharing). Segmentation (segment table: base/limit). Segmentation with paging (x86). TLB (associative memory, hit ratio). Memory protection (privilege levels: Ring 0 kernel, Ring 3 user).	16	7
5	<b>Virtual Memory &amp; Performance</b> Virtual memory fundamentals, demand paging, page fault handling. Locality (spatial/temporal). Working set model, dirty bit. Copy-on-write. Page replacement: Optimal, FIFO (Belady's anomaly), Second Chance, Clock, NRU, LRU (approximation). Frame allocation (equal/proportional/priority). Global vs local replacement. Thrashing (working set, PFF). Memory-mapped files. Kernel allocation (buddy system, slab allocator). NUMA basics.	18	8

<b>6</b>	<b>Storage, File Systems &amp; Security</b> I/O hardware (device controllers, DMA). I/O software (interrupt handlers, device drivers). File concepts, access methods (sequential/direct/indexed), directory structures. File allocation (contiguous/linked/indexed: FAT/inode). Free-space (bitmap/linked list). Disk structure, scheduling (FCFS, SSTF, SCAN, C-SCAN, LOOK). SSD (wear leveling, garbage collection), NVMe. RAID (0/1/5/6/10). File systems (ext4, NTFS, FAT32), journaling. OS security: authentication (MFA), access control (ACL, DAC/MAC/RBAC), encryption (TLS/SSL). Containers (Docker, namespaces: PID/network/mount, cgroups).	22	10
----------	--	----	----

**i. Text Book and Reference Book:**

1. **Operating System Concepts Essentials (TextBook)**  
By by Avi Silberschatz, Peter Galvin, Greg Gagne | 9th Edition Wiley Asia Student Edition.
2. **Operating System: A Design-oriented Approach**  
By Charles Crowley, | 1st Edition - Irwin Publishing
3. **Operating Systems: A Modern Perspective**  
By by Gary J. Nutt | Addison-Wesley; 2nd Edition | 2nd Edition
4. **Design of the Unix Operating Systems**  
By Maurice Bach, | Prentice-Hall of India | 8th Edition
5. **Understanding the Linux Kernel**  
By Daniel P. Bovet, Marco Cesati, | O'Reilly and Associates | 3rd Edition

**j. Experiment List:**

Exp. No.	Name of the Experiment
1.	<b>Explore basic Linux commands and shell scripting for task automation in Linux-based systems.</b>
2.	<b>Write a shell script to calculate the sum of digits from a given set of numbers.</b>
3.	<b>Create a shell script to validate and format user input for a date (dd-mm-yyyy).</b>
4.	<b>Write a shell script to check whether a given string is a palindrome.</b>
5.	<b>Write a shell script to greet the user based on login time (morning/afternoon/evening).</b>
6.	<b>Write a program to find the largest number from command-line arguments.</b>
7.	<b>Write a C program to create and manage a child process in a multitasking environment.</b>
8.	<b>Implement CPU scheduling algorithms such as Shortest Job First (SJF) and Multilevel Queue Scheduling.</b>
9.	<b>Implement FIFO and LRU page replacement algorithms and analyze their performance.</b>
10.	<b>Implement Banker's Algorithm to simulate safe resource allocation and prevent deadlocks.</b>

(2)

- a. **Course Name:** Linear Algebra
- b. **Course Code:** 03019102B S01
- c. **Prerequisite:** Basic Knowledge of Matrix and Determinants

d. **Rationale:** To attain the knowledge of Matrices and apply to the engineering field to check the consistency of the system. The concept of vector space helps to understand programming in field of computer engineering.

e. **Course Learning Outcomes:**

<b>CLO 1</b>	Understand system of linear algebraic equation and solution techniques using matrix.
<b>CLO 2</b>	Exhibit the understanding of vector space and subspace and solve related problems.
<b>CLO 3</b>	Compute eigenvalues and eigenvectors and diagonalize the matrix
<b>CLO 4</b>	Create orthogonal and orthonormal bases: Gram-Schmidt process and use bases and orthonormal bases to solve application problem.

f. **Teaching & Examination Scheme:**

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
4	-	-	4	40	20	-	60	-	100

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

g. **Course Content:**

Sr.	Topics	W	T
1	<b>Matrices</b> Introduction, System of linear equations (Homogeneous and Nonhomogeneous systems) by Gauss Elimination, and LU Decomposition Method, Rank, Eigenvalues and Eigenvectors, Algebraic and Geometric Multiplicity, Diagonalization, Cayley-Hamilton theorem and Applications-coding-decoding, Principal Component Analysis (PCA).	35	21
2	<b>Vector Space</b> Vector spaces over the field of real numbers, Elementary Properties of Vector Spaces, Subspaces, Spanning set, Linear independence and dependence, Basis and Dimension.	20	12
3	<b>Linear Transformation</b> Definition of Linear Transformation from $R^n$ to $R^m$ Some Standard Linear Transformation, The Null Space & Column Space, The Rank-Nullity Theorem, Matrix of Linear Transformation.	25	15
4	<b>Inner Product Spaces</b> Inner-product spaces, norm and its properties, angle between two vectors, orthogonal vectors, orthonormal vector, Gram-Schmidt process, orthonormal basis.	20	12

h. **Text Book and Reference Book:**

1. **Elementary Linear Algebra with Applications, (TextBook)**  
By H. Anton | John Wiley
2. **Linear Algebra and Its Applications**  
By G. Strang | Brooks/Cole India
3. **Advanced Engineering Mathematics**  
By Erwin Kreyszig | Willey India Education
4. **Linear Algebra,**  
By K. Hoffman and R. Kunze, | Prentice Hall
5. **Calculus**  
By T.M. Apostol | John Wiley Publication | Vol. II
6. **Linear Algebra: A Modern Introduction,**  
By D. Poole, | 2nd Edition, Brooks/Cole

(3)

a. **Course Name:** Physics of Semiconductors

b. **Course Code:** 03019201B S01

c. **Prerequisite:** Knowledge of Physics and some basic concepts in Mathematics like differentiation, integration, limit, differential equation, vector calculus up to 12th science level.

d. **Rationale:** Knowledge of physics is essential for all Engineering branch because physics is the foundation subject of all the branches of engineering and it develops scientific temperament and analytical capability of engineering students. Comprehension of basic physics concepts enables the students to solve engineering problem logically and develop scientific approach.

e. **Course Learning Outcomes:**

<b>CLO 1</b>	To conceptualize semiconducting materials on the basis of band theory.
<b>CLO 2</b>	To get familiar with optical fibres and semiconductor based optoelectronic devices useful in fibre optic system.
<b>CLO 3</b>	To conceptualize the foundational principles of quantum Physics and their role in quantum computing.
<b>CLO 4</b>	Formulate and conceptualize various theoretical aspects and the physical phenomena at Low dimensional level (nano)

f. **Teaching & Examination Scheme:**

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
3	-	2	4	20	20	20	20	20	150

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

g. **Course Content:**

Sr. No.	Content	Weightage	Teaching Hours
1	<b>Band theory &amp; Semiconductors</b> Band structure of solid materials, E-k diagram, Direct and Indirect band gap, Effective mass, Concept of Fermi Energy, Density of state, Fermi Level in Intrinsic and Extrinsic Semiconductors, Ohmic and Schottky Junction	25%	11
2	<b>Optoelectronics</b> Interaction of radiation with Matter, Absorption, Spontaneous and Stimulated emission, Characteristics of Lasers, Diode Laser, LED, Photodiode and their applications. Principle and Structure of Optical Fiber, Numerical Aperture of fiber, Types of Optical Fibers, Attenuation in Optical Fibers, Applications of Optical Fibers.	25%	11

<b>3</b>	<b>Quantum Mechanics &amp; Quantum Computing</b> Quantum postulates, wave function, Schrodinger's equation time dependent, independent (No derivation), One-dimensional potential well, quantum tunneling and its application in soft computing Key Principles of Quantum Computing, Difference between classical and quantum computing, Quantum Computing advantages Challenges and Application.	<b>30%</b>	<b>13</b>
<b>4</b>	<b>Low Dimensional Materials</b> Basic characteristic including synthesis, properties, quantum confinement, classification: Quantum Dot, Quantum well, Quantum Wire and their applications, Novel Materials and their applications	<b>20%</b>	<b>10</b>

**h. Text Book and Reference Book:**

1. **Physics of Semiconductor Devices (TextBook)**  
By S. M. Sze and K. N. Kwok | John Wiley & Sons
2. **Semiconductor Optoelectronic Devices**  
By P. Bhattacharya | Prentice Hall of India, Pub. Year 1997
3. **Engineering Physics (TextBook)**  
By B.K.Pandey S Chaturvedi and M. Venkanna | - Cengage Publication
4. **Semiconductor Optoelectronics (TextBook)**  
By J. Singh | McGraw-Hill Inc, Pub. Year 1995

**i. Experiment List:**

Sr. NO.	Experiment List
1	I-V characteristics of light emitting diode in forward bias.
2	I-V characteristics of Zener diode in reverse bias.
3	Determination of Velocity of ultrasonic waves in water.
4	Determination of Dielectric constants of Dielectric samples.
5	Measurement of Band gap of semiconductor material.
6	Measurement of Hall coefficient and carrier concentration in semiconductor material.
7	Measurement of Planck's constant using LED.
8	Measurement of wavelength of laser light using diffraction grating.
9	Measurement of Numerical aperture of an optical Fiber.
10	Determination of the size of lycopodium powder particles.
11	Measurement of power loss in an optical fibre.
12	Moment of Inertia of a flywheel.
13	Young's Modulus by bending of beam.

**(4)**

- a. **Course Name:** ICT Workshop
- b. **Course Code:** 03010702E S01
- c. **Prerequisite:** Basic Computer Knowledge and Physics

**d. Rationale:** This course is design to provide basic knowledge of Electronics components and computer components. This course helps in learning problem solving process of Electronics circuits and Computer.

**e. Course Learning Outcomes:**

<b>CLO 1</b>	Gain ability to understand the working of Electronics Components
<b>CLO 2</b>	Ability to understand the operating of various testing and measurement instrumentation. CO3:
<b>CLO 3</b>	Ability to use Power Supply
<b>CLO 4</b>	Ability to learn working and use of different IoT sensors
<b>CLO 5</b>	Ability to design electronic circuit for the specific applications.

**f. Teaching & Examination Scheme:**

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
	-	2	1	-	-	20	-	30	50

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

**g. List of Practical:**

Sr. NO.	Experiment List
1	<b>Introduction to Breadboard and Digital Multi Meter, Verify the circuit analysis (voltage and current)</b>
2	<b>Identification, symbolic representation and testing of various electronics components.</b>
3	<b>To study CRO and Function generator with specifications.</b>
4	<b>To verify Ideal Value and practical value of voltage regulator ICs.</b>
5	<b>Introduction to Arduino IDE with LED Blinking</b>
6	<b>Interfacing Button-Controlled LED with Arduino board.</b>
7	<b>Buzzer Alarm System Using Arduino.</b>
8	<b>Demonstrate the function of LCD Display Interfacing with Arduino.</b>
9	<b>Relay-Based ON/OFF Control System with Arduino</b>
10	<b>Demonstrate the working of Temperature Sensor-DHT11</b>

11	<b>Demonstrate the working of Ultrasonic Distance Measurement sensor.</b>
12	<b>Verify the functionality of water flow sensor</b>
13	<b>Demonstrate the working of PIR Sensor using Arduino.</b>
14	<b>Project based on electronics components and sensors</b>

(5)

- a. **Course Name:** Object Oriented Programming
- b. **Course Code:** 03010502ES01
- c. **Prerequisite:** Basic knowledge of programming
- d. **Rationale:** The practical component reinforces Core Java concepts through hands-on programming. It helps students apply OOP, collections, exception handling, file I/O, multithreading, and Java 8 features to solve real-world problems, preparing them for advanced development and industry roles.
- e. **Course Learning Objective:**

<b>CLOBJ 1</b>	To explain Java fundamentals, JVM architecture, program execution flow, and basic programming constructs for developing console-based applications.
<b>CLOBJ 2</b>	To apply object-oriented programming principles such as encapsulation, inheritance, polymorphism, abstraction, and package organization in Java.
<b>CLOBJ 3</b>	To develop programs using arrays, strings, file handling, exception handling, and Java collection frameworks for efficient data processing.
<b>CLOBJ 4</b>	To demonstrate multithreading and synchronization concepts for developing concurrent and responsive Java applications.

f. **Course Learning Outcomes:**

<b>CLO 1</b>	Solve higher-order ordinary differential equations with constant coefficients and Euler's ODE with variable coefficient
<b>CLO 2</b>	Apply series solution methods and special functions like Bessel's' functions to solve the differential equations and analyze complex physical phenomena
<b>CLO 3</b>	Use the Laplace transform as a tool to solve differential equations and Fourier integral representation
<b>CLO 4</b>	Apply Fourier integral to analyse the representation of functions in terms of frequency component.
<b>CLO 5</b>	Apply mathematical tools needed in evaluating vector calculus and their usage like Work, Circulation and Flux.
<b>CLO 6</b>	Apply Multiple Integration to solve physical problems such as Area, Volume etc.

**g. Teaching & Examination Scheme:**

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
3	-	2	4	20	20	20	60	30	150

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

**h. Course Content:**

Sr. No.	Content	Weightage	Teaching Hours
1	<b>Java Basics, JVM &amp; Syntax</b> History and features of Java, JDK vs JRE vs JVM, compilation and execution flow. JVM basics (class loading, memory areas). Basic syntax: data types, variables, operators, input (Scanner), control flow (if/switch, loops). Simple console programs.	12	5
2	<b>Core OOP Concepts in Java</b> Defining classes and objects, constructors, this and super, encapsulation with getters/setters. Inheritance (single, multilevel), method overloading and overriding, runtime polymorphism, abstract classes and interfaces. Access modifiers and packages basics. UML-style thinking for class design.	20	5
3	<b>Arrays, Strings &amp; Basic I/O</b> Working with 1D and 2D arrays for data storage and algorithms. Strings in Java (String, StringBuilder), immutability, common string methods. Basic file and console I/O use-cases (reading input, printing formatted output). Simple problems using arrays and strings.	16	5
4	<b>Exception Handling &amp; Debugging</b> Types of errors, exceptions vs errors. try-catch-finally, multiple catch blocks. Checked vs unchecked exceptions, throws, creating simple custom exceptions. Exception propagation and basic debugging using stack traces and IDE tools.	16	5
5	<b>Introduction to Collections</b> Need for collections vs arrays. Introduction to List and Set with ArrayList and HashSet. Iteration using for-each and Iterator. Basics of Map with HashMap (key-value access). Generics at a basic level (List<String>, List<Integer>). When to choose which collection.	20	5
6	<b>Java Multithreading</b> Introduction to concurrency and multithreading, Thread creation (Thread class, Runnable interface), Thread lifecycle and states, Thread methods (start, run, sleep, join, interrupt), Thread priorities and daemon threads, Synchronization (synchronized keyword).	16	5

**i. Text Book and Reference Book:**

1. Herbert Schildt, Java: The Complete Reference, McGraw-Hill.
2. Cay S. Horstmann, Core Java – Volume I & II, Pearson.

3. Joshua Bloch, Effective Java, Addison-Wesley.
4. Kathy Sierra & Bert Bates, Head First Java, O'Reilly.

**j. List of Practical:**

Sr. NO.	List of Practical
1	<b>JVM &amp; Basic Program Structure</b> Write a simple Java program to display student details and explain the compilation and execution process of a Java program through comments.
2	<b>Control Statements &amp; Number Logic</b> Develop a Java program that uses conditional and looping statements to implement a menu-driven calculator and generate number-based patterns such as prime numbers and Fibonacci series.
3	<b>Class, Object &amp; Constructors</b> Design a BankAccount class with appropriate data members and methods to perform deposit and withdrawal operations using constructors and access modifiers.
4	<b>Inheritance &amp; Polymorphism</b> Create a base class Employee and derive Manager and Developer classes to demonstrate method overriding and runtime polymorphism for salary calculation.
5	<b>Interfaces &amp; Abstraction Using Packages</b> Implement an interface Payment and create multiple payment mode classes to demonstrate abstraction, interface implementation, and package usage.
6	<b>Arrays &amp; String Manipulation</b> Write a Java program to perform operations on arrays and strings, including sorting an array, reversing a string, and demonstrating string immutability.
7	<b>Exception Handling &amp; Custom Exception</b> Develop a banking application that throws a custom InsufficientBalanceException when withdrawal amount exceeds available balance, using proper exception handling mechanisms.
8	<b>Java Collections Framework</b> Create a Student class and use ArrayList, HashSet, and HashMap to store, sort, and manage student records using Comparable and Comparator interfaces.
9	<b>File Handling &amp; Serialization</b> Write a Java program to store student details in a file, read the data back, and demonstrate object serialization and deserialization.
10	<b>Multithreading &amp; Java 8 Features</b> Develop a multithreaded Java application using Thread and Runnable, synchronize shared resources, and use Lambda expressions and Stream API to process collections.

(6)

- a. **Course Name:** Advanced Communication and Interpersonal Skills
- b. **Course Code:** 03010002H M01
- c. **Prerequisite:** Basic Communication Skills are essential for all Engineers
- d. **Rationale:** Strengthen core language and soft skills through applied grammar, communication tasks, and vocabulary building.
- e. **Course Learning Outcomes:**

<b>CLO 1</b>	Remember the essential rules of English grammar and core vocabulary required for accurate communication. Understand spoken and written texts at an intermediate level to grasp meaning, ideas, and context.
<b>CLO 2</b>	Understand spoken and written texts at an intermediate level to grasp meaning, ideas, and context.
<b>CLO 3</b>	Apply grammatical structures, vocabulary, and writing skills in essays, self-introductions, and everyday communication.
<b>CLO 4</b>	Analyze personal strengths and weaknesses through SWOT analysis to enhance grooming and personality development. Analyze communication habits and daily routines to improve time management and professional effectiveness.
<b>CLO 5</b>	Analyze communication habits and daily routines to improve time management and professional effectiveness.

**f. Teaching & Examination Scheme:**

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
1	-	2	2	40	-	20	60	30	150

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

**g. Course Content:**

Sr. No.	Content	Weightage	Teaching Hours
1	<b>Subject Verb Agreement</b> <ul style="list-style-type: none"><li>● Rules of subject-verb matching Singular vs plural forms</li><li>● Common agreement errors</li></ul>	10%	1

2	<b>Reported Speech</b> <ul style="list-style-type: none"> <li>• Direct and indirect speech</li> <li>• Changing tenses, pronouns, and time expressions Reporting statements, questions, and commands</li> </ul>	10%	1
3	<b>Active and Passive Voice</b> <ul style="list-style-type: none"> <li>• Difference between Active and Passive Rules for converting sentences</li> <li>• Usage in real-life contexts</li> </ul>	10%	1
4	<b>Building Vocabulary</b> <ul style="list-style-type: none"> <li>• Word meanings and usage Synonyms and antonyms</li> <li>• Homonyms, Homophones, Homographs, Idioms</li> </ul>	10%	2
5	<b>Grooming and Personality Development</b> <ul style="list-style-type: none"> <li>• Importance of dressing and professional etiquette Building confidence and positive body language</li> </ul>	10%	2
6	<b>SWOT Analysis with Self Introduction</b> <ul style="list-style-type: none"> <li>• Identifying strengths, weaknesses, opportunities, threats</li> <li>• Preparing and delivering a confident self-introduction</li> <li>• Developing self-awareness and a growth mindset</li> </ul>	10%	2
	<b>Reading Comprehension (Intermediate Level)</b> <ul style="list-style-type: none"> <li>• Understanding written texts</li> <li>• Finding main ideas and supporting details Answering questions accurately</li> </ul>	10%	2
	<b>Listening Comprehension (Intermediate Level)</b> <ul style="list-style-type: none"> <li>• Listening for specific information</li> <li>• Identifying tone and purpose Responding appropriately.</li> </ul>	10%	1
	<b>Essay Writing</b> <ul style="list-style-type: none"> <li>• Structure of an essay: introduction, body, conclusion</li> <li>• Organizing ideas logically</li> <li>• Using appropriate language and tone</li> </ul>	10%	2
	<b>Time Management</b> <ul style="list-style-type: none"> <li>• Importance of managing time Prioritization</li> <li>• Creating schedules</li> </ul>	10%	1

#### **h. Text Book and Reference Book:**

1. **Technical Communication: Principles And Practice**  
By Sangeetha Sharma, Meenakshi Raman | Oxford University Press | 2nd Edition
2. **Personality Development and Soft Skills**  
By Barun K Mitra | Oxford, 2011
3. **High School English Grammar and Composition**  
By Wren & Martin | S. Chand Publishing, Pub. Year 2017
4. **English Grammar in Use**  
By Raymond Murphy | Cambridge University Press, Pub. Year 2019
5. **Communication Skills and Soft Skills**  
By Suresh Kumar | Pearson Publication, 2010

(7)

a. **Course Name:** Privacy and Security in Online Social Media

b. **Course Code:** 03M10002U E01

c. **Prerequisite:** Basic knowledge of social media, privacy and security.

d. **Rationale:** The course explores the fundamentals of Online Social Networks (OSNs), focusing on data collection, analysis, trust, security, and privacy.

e. **Course Learning Outcomes:**

<b>CLO 1</b>	Understand Online Social Networks and Data Collection – Explain the fundamentals of online social networks, data collection methods, and the challenges and opportunities associated with them.
<b>CLO 2</b>	Analyze Trust, Credibility, and Reputation in Social Systems – Evaluate the role of trust and reputation in online communities and develop strategies to detect misinformation and enhance credibility.
<b>CLO 3</b>	Examine Privacy and Security Risks in OSNs – Assess the impact of privacy disclosure, phishing, and fraudulent activities in social media and propose countermeasures.
<b>CLO 4</b>	Apply Data Collection and Analysis Techniques – Utilize social media APIs and data processing methods to extract, clean, and analyze information from online social networks.
<b>CLO 5</b>	Conduct Research on Emerging OSN Topics – Engage in discussions and research on social media security, privacy, and law enforcement applications, contributing to the field through case studies and research papers.

**f. Teaching & Examination Scheme:**

Teaching Scheme				Evaluation Scheme					
L	T	P	C	Internal Evaluation			ESE		Total
				MSE	CE	P	Theory	P	
3	-	-	3	40	20	-	60	-	100

**L-** Lectures; **T-** Tutorial; **P-** Practical; **C-** Credit; **MSE-** Mid-Semester Evaluation, **CE-** Continuous Evaluation, **ESE-** End Semester Examination

**g. Course Content:**

Sr. No.	Content	Weightage	Teaching Hours
1	<b>Introduction to Online Social Networks</b> Definition and Evolution of Online Social Networks (OSNs), Data Collection from Social Networks, Challenges, Opportunities, and Pitfalls in OSNs, Social Media APIs for Data Extraction.	20%	8
2	<b>Data Collection and Analysis in OSNs</b> Techniques for Collecting Data from Online Social Media, Ethical Considerations in Social Media Data Collection, Data Processing and Cleaning for Analysis, Case Studies on Social Media Data Collection.	20%	8
3	<b>Trust, Credibility, and Reputation in Social Systems</b> Understanding Trust and Credibility in Online Communities, Reputation Systems and Their Impact on User Behavior, Detecting Misinformation and Fake News in OSNs, Methods for Enhancing Trustworthiness in Social Media.	20%	8
4	<b>Security, Privacy, and Fraud Detection in OSNs</b> Online Social Media and Its Role in Law Enforcement & Policing, Information Privacy: Disclosure, Revelation, and Its Effects, Phishing Attacks in OSNs and Their Countermeasures, Identifying Fraudulent Entities and Fake Profiles in OSNs.	20%	8
5	<b>Research and Practical Applications</b> Review and Discussion of Key Topics (Refresher), Analyzing Real-World OSN Case Studies, Research Paper Discussions on Online Social Networks and Security, Future Trends in OSN Security and privacy.	20%	8

**h. Text Book and Reference Book:**

1. Social Media Mining: An Introduction  
By Reza Zafarani, Mohammad Ali Abbasi, Huan Liu: Provides a comprehensive introduction to social media data mining, including machine learning and network analysis techniques.
2. Mining the Social Web  
By Matthew A. Russell: Covers data collection, analysis, and visualization techniques using Python and APIs from platforms like Twitter, Facebook, and LinkedIn.
3. Security and Privacy-Preserving Techniques in Social Networks"  
By Barbara Carminati & Elena Ferrari Explores security challenges, privacy risks, and trust management in online social networks.